Available online at www.sciencedirect.com

ScienceDirect

Computer Communications 31 (2008) 1318–1328

computer communications

www.elsevier.com/locate/comcom

Statistical delay budget partitioning in wireless mesh networks

Najah A. Abu-Ali *, Hossam S. Hassanein

School of Computing, Queen's University, 52 Rosemund Cres., Kingston, Ont., Canada K7M 6Z4

Abstract

Wireless Mesh Networks (WMNs) are currently attracting strong attention due to their great potential in supporting multimedia applications with real-time transport with last-mile Internet access. Multimedia end-to-end transmission requires Quality of Service (QoS) guarantees. Mapping end-to-end QoS requirements into link QoS requirements is an important step for providing QoS in WMNs. Despite the importance of this functionality, it is yet to be addressed in WMNs or, more generally, in multihop wireless networks. Such mappings, however, have resulted in several algorithms being proposed for connection-oriented wired networks. The algorithms proposed, nevertheless, are either near-optimal or heuristics, and provide solutions for only one end-to-end requirement.

In this paper, we propose a partitioning algorithm that is capable of partitioning multiple end-to-end QoS requirements simultaneously. We define QoS as the pair of the required end-to-end delay and the violation probability of meeting the required end-to-end delay. Our approach is motivated by experiments decisively showing that the delay probability distribution can be accurately characterized by a gamma or logistic distribution, thus there is not a specific one distribution that can characterize the delay. This conclusion is used to formulate a mathematical linear program that optimally partitions the end-to-end delay and the violation probability into link delays and link violation probabilities without imposing any specific delay distribution. Extensive simulation verified the effectiveness of the algorithm compared to two representative QoS partitioning algorithms. The proposed algorithm outperforms the other algorithms for loose and stringent QoS requirements, and over different path lengths.

Keywords: Wireless mesh networks; End-to-end QoS; Delay budget partitioning; IEEE 802.16j; IEEE 802.11s; Radio resource management

1. Introduction

Wireless Mesh Networks (WMNs) are multihop wireless networks that are integrally different from mobile ad hoc networks as the former are aimed at backhauling service networks. Mesh routers provide gateway/bridge functionality to integrate heterogeneous networks, i.e., comprising wired and wireless connectivity. Accordingly, a mesh router is usually equipped with multiple wired and wireless interfaces, possibly of different technologies, and are characterized by little to know mobility.

Interest in WMNs is currently growing strong as they are expected to support multimedia and mission critical applications with real time transport for last mile Internet access. Multimedia applications require QoS guarantees for end-to-end delay, jitter, packet loss and flow rate. Different multimedia applications also have different QoS requirements. For example, certain real-time applications, e.g., tele-medicine, require hard bounds on end-to-end delay. Thus, a deterministic guarantee is required from the network, where the delay bound of any packet is not allowed to be violated. In other words, the delay violation probability for such applications must be guaranteed by the network to be zero. Other real time applications, like streamed multimedia, require soft bounds on the end-to-end delay. This means that the delay of small number of packets is allowed to violate delay bound. Such bounds are called stochastic delay bounds. Supporting real time applications over backhaul WMNs (for example, IEEE 802.16-2004-mesh mode, IEEE 802.16j, and IEEE 802.11s) is not a trivial task because of the wide variation of QoS requirements of real time applications and the incapability of certain standards, such as

---

* Corresponding author. Tel.: +1 613 544 6917.

E-mail addresses: najah@cs.queensu.ca (N.A. Abu-Ali), hossam@cs.queensu.ca (H.S. Hassanein).

0140-3664/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2008.01.050

IEEE 802.16 and IEEE 802.11s, of supporting end-to-end QoS requirements. However, in the emerging IEEE 802.16j, it is possible each user can have more than one connection or service type with pre-defined QoS requirements and different traffic models. Traffic models support four service types for QoS including UGS, rtPS, nrtPS, and BE. This classification distinguishes requirements for video, audio and data services.

The standards IEEE 802.16j and 802.16-mesh as a backhaul WMN are based on Time Division Multiple Access/Time Division Duplex (TDMA/TDD) technology and operate under distributed control or centralized control, i.e., distributed scheduling or centralized scheduling. Mesh nodes are required to reserve periodic time slots for their data transmissions and according to the QoS requirements. Thus to have a successful scheduling, mesh nodes need to be synchronized for TDMA operation. The reserved time slots are used to transmit packets from a source mesh node to a destination mesh node over a multihop wireless route in TDD mode. The traffic slots reserved by the mesh nodes are done on hop-by-hop basis, with each one-hop link operating independently.

For mesh networks based on IEEE 802.11s, mesh routers follow the IEEE 802.11e standard to provide QoS by allocating more resource to the traffic with higher demands. End-to-end communications in multihop network involve several radio links in tandem, and as discussed in [1], 802.11e can not support QoS services well in multihop networks. Thus, offering QoS guarantees using 802.11e over multiple hops requires a mechanism above traditional MAC in order to make end-to-end flow based assignments and to guarantee allocated service assignments.

To provide end-to-end QoS guarantees over link-based standards, such as IEEE 802.16 and IEEE 802.11s, end-to-end QoS bounds have to be optimally and correctly mapped into link QoS bounds. The mapping process is a very subtle step in QoS provisioning as it impacts routing and link resource allocation. Consequently, mapping plays a major role in balancing the load in a WMN and optimizing the utilization of network resources. In this paper, we address end-to-end statistical delay mapping over backhaul wireless mesh networks, also known as the delay budget partitioning problem, which is an important network optimization problem. We introduce a novel delay partitioning algorithm to solve the delay budget optimization problem for stochastic delay requirement. The algorithm maps the end-to-end QoS requirements denoted by the pair (D_r, P_r) into link QoS requirements denoted by the pairs $\{(d_i^r, p_i^r)\}_{i=1}^{M}$. The value d_i^r is the required link delay over the ith link, p_i^r is the required link violation probability and M is number of links, while D_r is the end to end delay and P_r is the end to end violation probability.

The proposed algorithm solves a linear programming problem to find the optimum link QoS pairs (d_i^r, p_i^r) using (D_r, P_r) and the probability density function (pdf) of the link delays. Section 3 presents our work in studying the delay distribution of homogenous and heterogenous traffic in a testbed wireless mesh network. The results of the experiments for identifying the delay distribution indicate that the empirical distribution of the delay can be fitted to well known distributions. We are able to identify two well known distributions, namely the gamma and logistic distribution. Our algorithm is accordingly designed to be independent of the type of the delay distribution. To the best of our knowledge, this algorithm is the first algorithm capable of optimally partitioning end-to-end stochastic QoS pair simultaneously in wireless mesh networks.

In designing the proposed delay budget partitioning algorithm, we are attempting to meet two contradicting objectives. The first is aimed at user satisfaction by meeting the user application's QoS requirement. The second objective is concerned with the service provider satisfaction by balancing the load over the connection path. The load is balanced over the path by allowing heavy loaded or in deep fading wireless links to have larger delay and violation probability portions than the lightly loaded or good channel quality links. Consequently, the proposed algorithm avoids generating future bottleneck links that results in rendering the whole path unusable.

The remainder of this paper is organized as follows. In the following section we review related work. We proceed by presenting the mesh testbed and characterizing the probability density function (pdf) of the delay in Section 3. The mathematical solution of the delay partitioning problem is formulated in Section 4. We compare the performance of the proposed algorithm with two existing algorithms in Section 5. We finalize the paper with the conclusions in Section 6.

2. Related Work

To the best of our knowledge, proposals addressing QoS provisioning in wireless communication networks focus on satisfying the link delay bound at a node or satisfying the end-to-end delay bound in general as the summation of the link delays along a path. Wang et al. [2] extend the Proportional Delay Differentiation (PDD) service model in wired networks into the Neighborhood PDD (NPDD) in wireless LANs. The objective of NPDD is to assure equal delay for the same class of service at each link along the path. The NPDD algorithm considers end-to-end delay as the summation of link delays. Draves and Padhye [3] proposed a new metric called the Weighted Cumulative Expected Transmission Time (WCETT) to find a route between static mesh routers based on estimating the delay at each node. The delay in [3] is calculated as a function of packet loss rate and available bandwidth. The packet loss is measured using the algorithm described in [4]. In [5], the authors estimate the end-to-end delay in wireless ad hoc networks by using a mobility model based on a discrete time Markov chain. The end-to-end delay is estimated by knowing a priori the random position of every node using the mobility model. Narlikar and Wilfong [6] propose a framework for activating wireless links in a backhaul network such that they can generate interference free routes between a single gateway and the mesh routers. The selection of links along the route is based on calculating an upper bound link delay. Shetiya and Sharma [7,8] discuss the problem of providing end-to-end QoS guarantees in 802.16 WMNs for different types of Internet traffic at the granularity of one connection. For example, for UDP traffic, both CBR and VBR, the authors detail how to assign the minimum number of slots at each node along the path given that the bounds for packet dropping probability are preserved. They assume that the end-to-end packet dropping probability is partitioned into link dropping probability. However, they do not provide any mechanism for partitioning the end-to-end QoS metric.

The above literature addresses the problem of estimating and/or bounding the link's delay or the end-to-end delay. However, non of the proposed algorithms addressed the problem of how to optimally partition the end-to-end delay to balance the load and/or maximize the network resource utilization.

Only in [9] have we found work that addressed delay budget allocation in multihop wireless networks. The authors propose three algorithms for partitioning end-to-end delay into link delays. The first algorithm assigns all the delay to the first node, the second assigns a constant delay at each node and does not discriminate between the flows QoS requirements, and the third algorithm partitions end-to-end delay equally among the links. The proposed algorithms in [9] do not take the wireless links' load into considerations, which results in imbalance of the load along the path. Additionally, the algorithms partition one QoS metric only which is the end-to-end delay.

The existing end-to-end delay partitioning algorithms in wired networks can be categorized into two categories; heuristic algorithms and optimization algorithms. The first category, i.e., the heuristic algorithms, solve the partition delay budget problem heuristically. Accordingly, the resulting link delay set is not necessarily an optimal solution for the problem. The equal [10] and the proportional [11] partitioning algorithms are examples of this category. The second category solves the problem of partitioning an end-to-end QoS requirement into local requirements, such that the path overall cost is minimized [12–16]. The path cost is considered as the summation of the link costs. The link costs are formulated as a general function that may be related to the resources needed for the QoS guarantees, or any proper network parameter, or it may reflect the service price. Thus, the path cost does not explicitly address the problem of balancing the load along the path. The authors of algorithms from the second category realized that the complexities of these algorithms are high, and suggested different simplified implementations to reduce the complexity.

3. WMN Testbed

To study the delay distribution, we deployed an 802.11b\g based wireless mesh network for supporting VoIP, video and FTP traffic. In this implementation, the mesh network is considered as a two tier network as shown in Fig. 1. The first tier provides access functionality for users in the infrastructure mode. The second tier provides backhaul connectivity to a gateway connected to the Internet in an ad hoc mode. Our focus is studying the aggregated traffic over the wireless backhaul mesh links. The mesh network clients are laptops with VoIP, Video, or FTP traffic. Real time traffic clients have connections across the mesh to other wireless devices through the help of a WinSIP server which simulates a high quality Voice and video traffic.

3.1. Hardware and software configuration

The mesh testbed consists of 13 Access Points (AP) WRT54GL, WRT54G, and Netgear WG302 access points distributed over an indoor area within a proximity of each other. Netgear WG302 APs are used to enable the WiFi multimedia that enhances video and voice calls. We install OpenWRT Linux operating system on the access points to be able to install capturing and mounting applications. The experiment is carried out during off-work hours to facilitate turning all department APs off to minimize interference. Since in this work we are not interested in studying routing over the wireless mesh network, pre-computed routes are maintained at each AP to provide for the traffic between the source and destination/s. Unless otherwise specified, the link rates are running at a fixed rate of 5.5 Mbps.

The following software packages are used in the setup.

- *OpenWRT Linux operating system:* An extensible Linux distribution that runs on Linksys WRT54G/GS routers. It is installed in Linksys WRT54G (L) routers in order to install Linux Kismet application which enables capturing packets from the AP.
- *Ethereal:* A protocol analyzer, or "packet sniffer" application, used to capture the packets.
- *WinSIP:* A VoIP SIP call generator and emulator used to generate voice and video simulated calls. It is installed over the users' laptops.
- *Kismet:* A Linux application used for capturing packets. It is installed on routers in order to calculate per-packet delay through capturing packets traversing the routers.
- *NetTime:* A simple time synchronization client used to synchronize the time in all laptops.
- *X-WRT:* A set of packages and patches to simplify the configuration of OpenWRT.
- *Samba/CIFS:* A linux application that is used to mount a shared folder of windows laptops to the linux folder. It is installed on the routers to make a shared folder on laptop and mount that folder on the router, to save the captured packets by Kismet.
- An in-house, Java-coded application that calculates the delays between the sender's packets and the receiver's packets by reading the packets time-stamps.

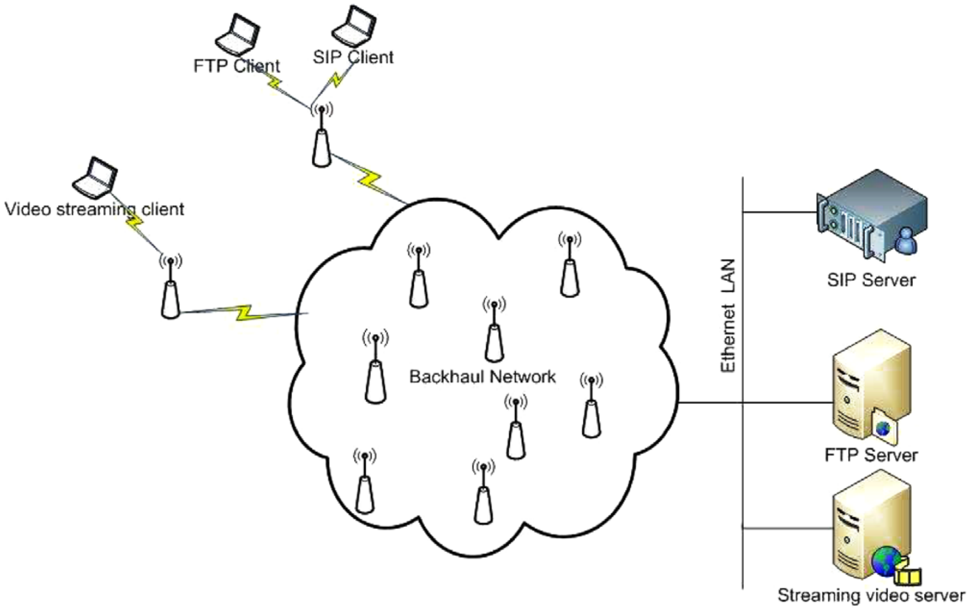


Fig. 1. A sample of the WMNs testbed setup.

3.2. The delay distribution

We designed different experiments to measure the link and end-to-end delays over the WMN testbed. We setup different experiments to investigate different types of homogeneous and heterogeneous flows, such as interactive voice and stream video, in the existence of different types of background traffic. The experiments are repeated for different paths while varying the number of hops. We measured the link delay over the WMN testbed. The measured link delays are used to construct an empirical histogram. Our experiments reveal that irrespective of the number of hops along the paths and type of traffic crossing the link, the empirical histograms almost have the same general shape. The empirical histograms are fitted into different types of standard pdf distributions. We found that the best fits for almost 90% of the empirical distributions are two standard distributions, which are the gamma and the logistic distributions. The gamma distribution is defined by the following.

$$f(\mathbf{x}) = \frac{(\frac{\mathbf{x}}{\theta})^{\gamma-1} \mathrm{e}^{-\frac{\mathbf{x}}{\theta}}}{\theta\Gamma(\gamma)}, \qquad \gamma > 0, \theta > 0, \tag{1}$$

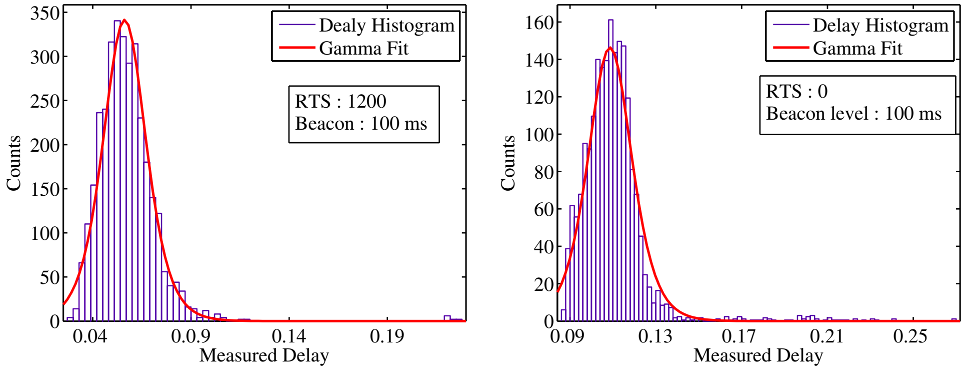
where θ γ are respectively called the scale parameter and the shape parameter, $\Gamma(\gamma)$ is the complete gamma function, if γ is an integer number, $\Gamma(\gamma) = (\gamma - 1)!$. The logistic distribution is defined by

$$f(\mathbf{x}; \boldsymbol{\mu}, \mathbf{s}) = \frac{\mathrm{e}^{-\frac{\mathbf{x}-\boldsymbol{\mu}}{s}}}{s(1 + \mathrm{e}^{-\frac{\mathbf{x}-\boldsymbol{\mu}}{s}})^2}, \qquad s > 0, \tag{2}$$

where s is the scale parameter and μ is the location parameter.

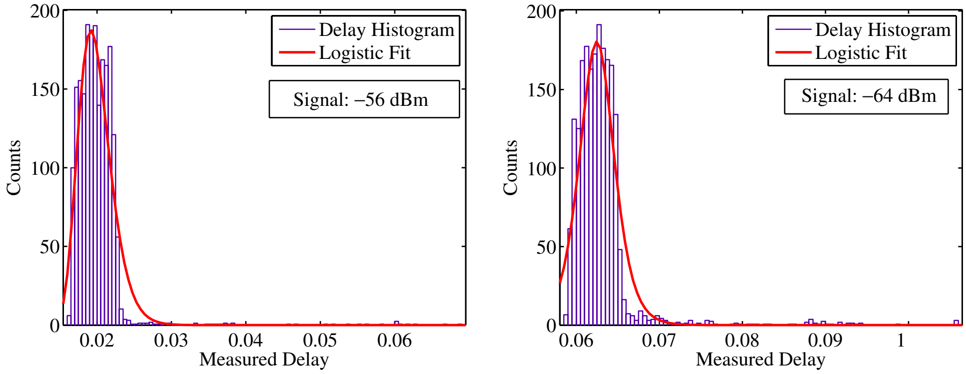
We used the maximum likelihood estimate to calculate the unknown parameters of the standard distributions that have the best fit to the measured delay data. In the following, we present the different experiments and their results.

- *RTS/CTS:* This experiment studied the effect of enabling the RTS/CTS messages over a 100 ms beacon. We noticed that the values of delays are affected. However, the distribution itself in the two scenarios is not affected as shown in Fig. 2(a) and (b). The empirical histogram is fitted into gamma distribution in the two cases. The delay distribution is studied for the VoIP traffic. We generated VoIP calls using the WiNSIP server multiplexed with FTP and video stream traffic generated from a different nodes in the mesh testbed.
- *Beacon size:* Fig. 4(a) and (b) show the delay distribution for two values of the beacon size; 100 and 200 ms. The experiment studied the delay distribution of VoIP traffic multiplexed with FTP and video stream traffic as aforementioned. In this experiment, the link delay with the smaller beacon size is less while the delay distribution stays the same. The delay of the experiment with larger Beacon is larger since packets spend longer time in the queues.
- *Signal strength:* Fig. 3(a) and (b) show the delay distribution under two different values of signal level. The Figures show that the link delay distribution preserves the fit in the two scenarios.
- *QoS support:* Fig. 5(a) and (b) show the effect of heterogenous traffic (FTP, video stream, and VoIP traffic) in two cases with and without enabling QoS support. Since our focus is on the delay distribution, the experiment studied the effect of video and FTP traffic on the delay of VoIP traffic. We noticed that the delay distribution is fitted to gamma distribution in the two cases. However, the measured delay in the QoS supported



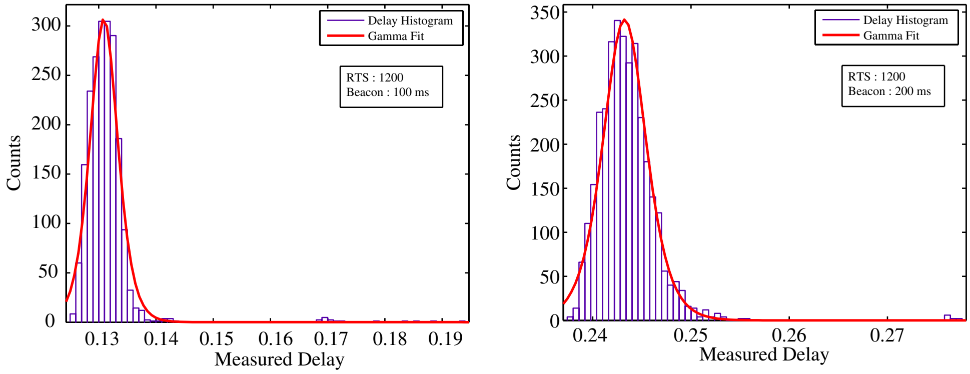
(a) RTS/CTS enabled (b) RTS/CTS disabled

Fig. 2. The delay distribution fit and the measured delay histogram of VoIP Traffic with (a) RTS/CTS Enabled, (b) RTS/CTS Disabled.



(a) higher SNR (b) lower SNR

Fig. 3. The delay distribution fit and the measured delay histogram of VoIP Traffic for (a) high SNR, (b) low SNR.



(a) Beacon 100ms (b) Beacon 200ms

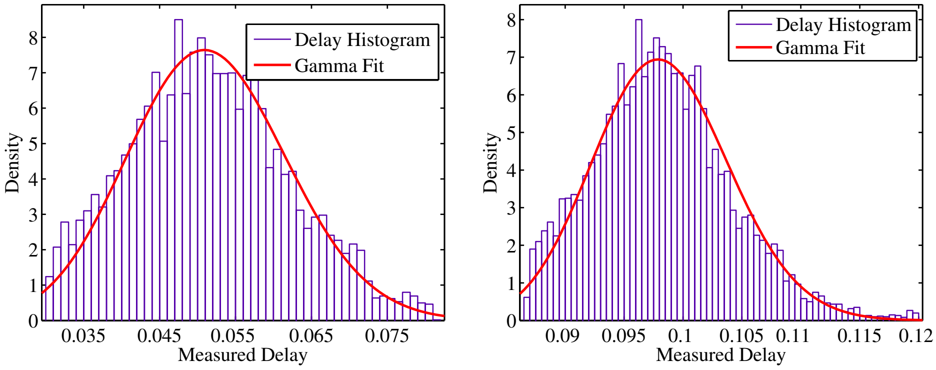
Fig. 4. The delay distribution fit and the measured delay histogram of VoIP Traffic with (a) beacon level = 100 ms, (b) beacon level = 200 ms.

experiment is less than that without support. The experiment was carried out for lightly loaded network and heavy loaded one and over different path lengths (3 to 10 hops).

We emphasize here that the experiments can provide rich information about the performance of real time applications in our Mesh testbed. However, our focus in this paper is on studying the link delay distribution and the end-to-end delay distribution.

3.2.1. Repeatability and reproducibility

We conducted 30 trials for link delay measurements for both link delay and the end-to-end delay. The trial measurements show high consistency and indicate the correctness of the measures. For each trial we collected 30000



(a) with QoS Support

(b) without QoS Support

Fig. 5. The delay distribution fit and the measured delay histogram of VoIP Traffic (a) with QoS support, (b) without QoS support.

delay samples. However, we noticed that 2000 samples are sufficient to capture the delay histogram.

All experiments are reproduced twice. We used the same APs, same routes under the same circumstances; i.e., time of the day and same location. The delay samples are within the same range and the delay distribution fit is of the same type with almost the same parameters.

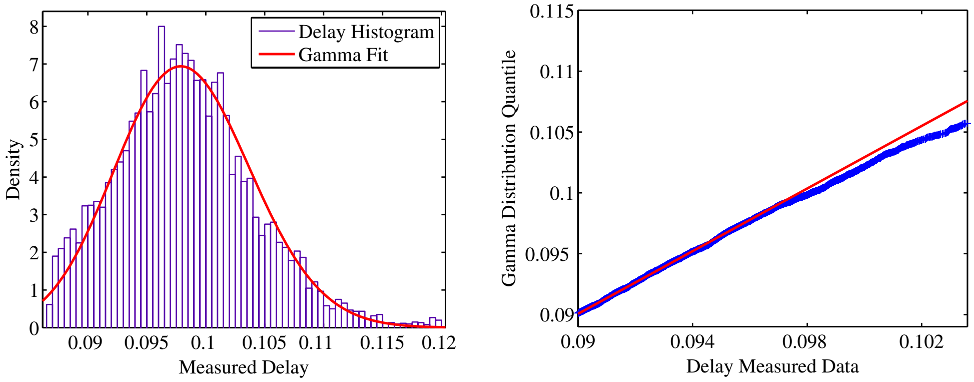
3.2.2. Goodness of the fit

We used the quantile–quantile plot to evaluate the goodness of the fit. Fig. 6(b) shows the goodness of fit as a quantile–quantile plot. A quantile–quantile plot is a standard test to check whether a distribution matches another by plotting the measured quantiles versus the estimated quantiles and checking if their quantile–quantile plot is a straight line. If the measured set resembles the estimated probability distribution under the test, the points should fall approximately along a 45-degrees reference line. The greater the departure from this reference line is, the greater the evidence that the compared data set comes from populations of different distributions. The linear relationship in Fig. 6(b) between the measured delay and the estimated gamma distribution shows that the gamma distribution is a good approximation for the measured delay.

Given the fact that we are interested in partitioning the delay and the violation probability as a QoS metrics, thus, we are interested in the asymptote of the Complementary Cumulative Distribution Function (CCDF); i.e., for a violation probability of 10^{-3} at a delay of 30 ms, we are interested of the 99.999th percentile of a delay of 30 ms. However and based on the experiments' results, 90% of the delay distribution histogram can be fitted into two different standard distributions, namely the gamma and logistic distributions and the other 10% are fitted to different types of distributions. Thus, we designed the delay partitioning algorithm to be independent of the distribution type itself, though we still assume that we have a standard distribution irrespective of its type. More important, the designed algorithm is independent from the technology used for the underlying connectivity, which means a capability of operating over any wireless mesh network. Consequently, and for the sake of generalization to other technologies, we do not assume a specific distribution.

4. End-to-end delay partitioning over backhaul WMNs

In this section, we present a novel delay partitioning algorithm, that is capable of partitioning both deterministic and statistical end-to-end delay bounds. The Optimized Delay Partitioning (ODP) algorithm partitions two QoS metrics simultaneously: an additive QoS metric which is the end-to-end delay, D_r, and a multiplicative QoS metric



(a) Gamma Fit of the Delay Histogram

(b) Goodness of the Fit

Fig. 6. (a) The delay distribution fit and the measured delay histogram of VoIP Traffic, (b) Goodness of the fit using quantile–quantile test.

which is the end-to-end violation probability, P_r. The end-to-end violation probability and end-to-end delay are denoted as the pair (D_r, P_r). The pair implies that packets belong to a WMNs connection traffic shall not violate there delay bound D_r by a probability larger than P_r.

Given (D_r, P_r), where $D_r = \sum_{i=1}^{M} d_i^r$ and $P_r = \prod_{i=1}^{M} p_i^r$, and the distribution parameters at each link, we optimize the links partitions of the link delay and the link violation probability. The end-to-end violation probability, P_r, may be expressed as an additive requirement, where $\ln(P_r)$ is the additive of the vector $\ln(p_i^r)$.

Definition. We define the end-to-end violation probability with respect to the QoS requirement pair (D_r, P_r) as

$$P\left(\sum_{i=1}^{M} \mathbf{x}_i > D_r\right) \overset{\Delta}{\leqslant} P_r \tag{3}$$

where the variable $\mathbf{x}_i$ denotes the random delay at the ith link along the path between the source and each destination and M is number of links along the path. We also define the link violation probability by

$$\begin{aligned} p_i^r &\overset{\Delta}{=} P(\mathbf{x}_i > d_i^r) \\ &= 1 - F_{\mathbf{x}_i}(d_i^r) \end{aligned} \tag{4}$$

Eq. (5) is also defined as the CCDF, $F^c_{\mathbf{x_i}}$. To calculate the link violation probability, we are interested in the asymptotic CCDF. Choudhury and Lucantoni [22,23] provide a generalized form of the asymptotic CCDF with methods to calculate its parameters. The generalized form of the asymptotic CCDF is defined by $F^c_{\mathbf{x}_i} \leqslant \alpha \mathbf{x}_i^{\beta} \mathrm{e}^{-\mathbf{x}_i \eta}$ as $\mathbf{x}_i \to \infty$, where η is the asymptotic decay rate, β is the asymptotic power and α is the asymptotic constant. The parameters α and η are assumed to be strictly positive, while β can be positive, negative or 0, however, in many contexts, β is of a zero value, which boils the CCDF into $F^c_{\mathbf{x}_i} \leqslant \alpha \mathrm{e}^{-\mathbf{x}_i \eta}$. Asymptotic parameters can be calculated using any method available in literature. [1] For example, the following equations relate the asymptotic parameters to the nth order moment using the moment-based algorithm to calculate the asymptotic parameters:

$$m_n = \int_0^{\infty} \mathbf{x}^n \mathrm{d}F(\mathbf{x}) = \int_0^{\infty} n\mathbf{x}^{n-1} F^c(\mathbf{x}) \mathrm{d}\mathbf{x} \tag{5}$$

$$\eta = \frac{n m_{n-1}}{m_n}, \quad \beta = \frac{\eta m_n}{m_{n-1}} - n, \quad \alpha = \frac{\eta^{\beta+n} m_n}{n! n^{\beta}} \tag{6}$$

where the CCDF, $F^c(\mathbf{x}) = 1 - F(\mathbf{x})$ and m_n is the nth moment of the Cumulative Distribution Function. Given the distribution can be any general standard distribution, the following steps is used to calculate the asymptotic parameters:

1. measure the delay at the backhaul routers' links,
2. build an empirical histogram from the measured delays and find the suitable fit,
3. send a message to the gateway with the type of the distribution and the distribution parameters. the message contains the following parameters: (distribution type, parameter-1 value, parameter-2 value, parameter-3 value), and
4. The gateway collects the messages from all the mesh nodes along the path between the gateway and the destination and uses this information to calculate the asymptotic parameters for each link along the path. The link delay distribution is not necessarily be the same distribution for each link.

For example if one of the nodes fitted its empirical distribution into a gamma distribution. The probability density function of a gamma distribution is given by Eq. (1) and the nth order moment of a gamma distribution is given by $m_n = \theta^n \frac{\Gamma(n+\gamma)}{\Gamma(\gamma)}$, where θ and γ are the shape and the scale parameters of the gamma distribution. The asymptotic parameters are calculated using the moment based algorithm as given in Eq. (6). If the distribution is any other distribution, then the nth order moment of the distribution is calculated and Eq. (6) is used to find the asymptotic parameters.

Note here, that the calculation of the asymptotic parameters can be done at the ingress router which is connected to the Internet or to any other technology; i.e., the gateway, while building the empirical distribution and fitting it to a standard distribution can be calculated at the mesh routers along the path and sent as updates to the gateway. The information included in the update message is just the probability density function parameters and its type. We remark here that this short message can be appended to the current messages exchanged by nodes in the mesh network. For example, in 802.16 mesh, the message can be included in MSH-NCFG, MSH-CSCH,…etc.

We proceed by taking the logarithm of the general form of the asymptotic CCDF,

$$\begin{aligned} F^c_{\mathbf{x}_i} = 1 - F_{\mathbf{x}_i} = p_i^r &\leqslant \alpha e^{-\mathbf{x}_i \eta}, \\ \ln(p_i^r) &\leqslant \ln(\alpha e^{-\mathbf{x}_i \eta}), \\ &\leqslant \ln(\alpha) - \eta \mathbf{x}_i, \end{aligned} \tag{7}$$

Eq. (7) relates the link violation probability to the asymptotic parameters of the CCDF, which is in turn related to the nth order moment of the distribution probability function that reflects the status of the wireless link; i.e., link utilization and link channel quality.

At this point we are ready to introduce the mathematical program which has the objective of optimally maximizing the partition of each link of the QoS pair (d_i^r, p_i^r) while taking into account the status of the links.

$$\max \sum\nolimits_{i=1}^{M} \mathbf{x}_i + \sum\nolimits_{i=1}^{M} \ln(p_i), \tag{8}$$

subject to:

---

[1] There are several methods to find the asymptotic parameters such as the moment based algorithm, asymptotic analysis with transform or moment generating function inversion method. How to use these methods or the advantages of one over the other are out of the scope of this paper.

$$\sum\nolimits_{i=1}^{M} \mathbf{x}_i \leqslant D_r, \tag{9}$$

$$\sum\nolimits_{i=1}^{M} \ln(p_i) \leqslant \ln(P_r), \tag{10}$$

$$\ln(p_i) + \mathbf{x}_i\eta \leqslant \ln(\alpha), \tag{11}$$

$$\mathbf{x}_i \leqslant d_i^{bw}, \tag{12}$$

$$-\ln(p_i) \geqslant 0, \tag{13}$$

$$\mathbf{x}_i \geqslant 0, \tag{14}$$

The objective function (8) attempts to allocate to each link the maximum link delay and violation probability possible such that the constraints are met. Constraints (9) and (10) ensure that the summation of the QoS link pairs, $(d_i^r, \ln(p_i^r))$ will not exceed the end-to-end QoS pair $(D_r, \ln(P_r))$. Constraint (11), which is Eq. (7), facilitate allocating the QoS link pair taking into account the link status condition, load and channel quality, thus, avoid generating any future bottleneck links and balance the load over the whole path. As aforementioned the condition link status is reflected through the two asymptotic parameters, since these values change with the link conditions. Thus, heavy loaded links or links in deep fade will have larger value of link delay and violation probability. Consequently, ODP algorithm allocates each link along the path portion of the delay and violation probability such that the link will be capable to meet the required link delay or violation probability. However, the heuristic algorithms, i.e., equal and proportional algorithm partition the delay irrespective of the link status or utilization thus packets crossing this link may exceed their delay bound by a value larger than the required violation probability. The last constraint, (12), is concerned with the case if the WMN connection has a data rate requirement in addition to the statistical delay requirement. Thus, the delay corresponding to the data rate requirement denoted by d_i^{bw} – this depends on the type of schedular used at the mesh node – is accounted for. If delay partition due to statistical delay requirements, $\mathbf{x}_i$, is grater than d_i^{bw} at a link, the partitioned value should be d_i^{bw}, otherwise, it is the optimum value that solves the above linear program (8). For example, if the statistical delay bound is 30 ms and the delay corresponds to the required traffic rate is 25 ms then the link partition will be equal to 25 ms. For instance, if the schedular is TDMA then the required time slots to meet the required rate will correspond to the bandwidth delay bound.

5. Performance evaluation

We used Waxman model [18] to generate the simulation network of 30 mesh nodes and one gateway. The paths are calculated between each mesh router and a gateway, and are used to evaluate the effectiveness of the proposed ODP algorithm. During simulation, the paths are used to admit the WMN requests until the links are fully utilized and/or the connection request cannot be admitted. A connection request is rejected if the algorithm returns no or invisible solution for the mathematical program.

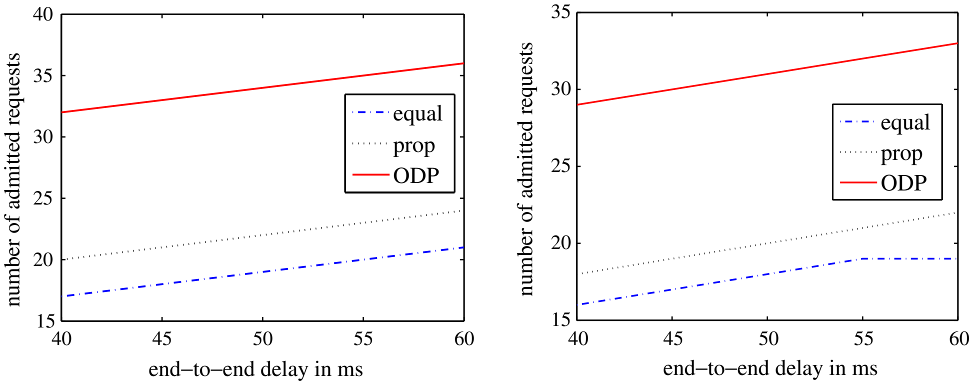
We simulated the OFDM physical layer using 20 MHz of bandwidth per channel which has a symbol duration of 12.5 μs. The channel is shared based on the Time Division Multiple Access (TDMA) of a frame size of 10 ms. We implemented the adaptive modulation and coding (AMC) for IEEE 802.16 OFDM physical layer. The links are simulated as a Raleigh fading channels and the SNR value is mapped into the AMC levels to provide for the raw data bit rates [24]. We did not simulate all the AMC levels, the lowest AMC level simulated is QPSK1/2 and the highest level simulated is 16 QAM 3/4. This range provides a row data rate specifically between 15.36 and 46.08 Mbps. Even though the simulation is run for 802.16 mesh network, we tried to provide a simulation environment, which is suitable for 802.11s as well. 802.11s employ TDMA for channel access and OFDM and AMC at the physical layer if it is overlaid over 802.11a as the physical layer technology. Thus, it will provide the closest physical layer specification to 802.16, since the AMC in 802.11a provides 12 Mbps data rate for QPSK and 48 Mbps data for 16 QAM. Thus, concerning the access model and the physical layer 802.11s/802.11a is comparable to IEEE 802.16. We remark here that the wireless link quality and the link utilization and capacity are captured in the delay distribution parameters.

IEEE 802.16 standard defines the messages for three scheduling mechanisms to schedule resources in the mesh mode, however, the standard does not define the data schedular itself. All of 802.16 data schedulers are TDMA-based: coordinated centralized, coordinated distributed and uncoordinated distributed. Each mechanism accommodates specific types of communications and potentially offers different levels of guarantees. For instance, the coordinated centralized scheduling is aimed at Internet traffic flowing in and out of the network through the gateway BS, while the distributed mechanisms accommodate intranet traffic. There are many proposals to implement schedulers as centralized coordinated scheduling schemes. We chose to implement the WFQ scheduler as a coordinated centralized scheduling to provide the required QoS assurance in the 802.16 TDMA WMN with a frame size of 10 ms.

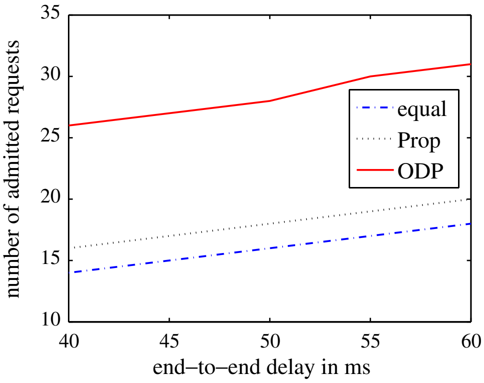
Given the token bucket traffic model, (σ, ρ), Parekh et al. [19] proved that the worst case delay contributed by the burstiness of the traffic is $\frac{\sigma_i}{g_i}$, where g_i is the bandwidth reserved at a link along the path to meet the QoS requirement of the flow. On the other hand, Georgiadis et al. [20] advocate that running a single traffic shaper at the minimum reserved link bandwidth is sufficient to completely smooth the traffic along the path. Accordingly, having different shapers at the downstream links operating at higher reserved bandwidth is not fruitful. Implementing a traffic shaper at the minimum reserved link bandwidth at the ingress router will therefore prevent the bursting of traffic at the intermediate links. Accordingly, the delay at the first mesh router consists of a link delay and a traffic shaping delay of $\frac{\sigma_i}{g_i^{\min}} \cdot g_i^{\min}$ is the minimum reserved link bandwidth along the path between the source and the destination.

The network links' capacity used in simulation are chosen to be between 12 and 50 Mbps. The delay distribution is calculated on intervals when the utilization of the link is increased by 10%. The shortest path is chosen to be 5 hops and the longest path is 10 hops. The two algorithms used in simulation to compare the performance of the ODP algorithm are the Equal Load partitioning Algorithm (ELA) and the Proportional Load partitioning Algorithm (PLA). ELA algorithm partitions the end-to-end delay equally among the links along the path [9]. The PLA algorithm partitions the end-to-end delay proportional to the load over the path links in wired networks [21]. Both algorithms, PLA and ELA partition the end-to-end delay only. Accordingly, we extend them to support partitioning the violation probability as well. PLA algorithm is adapted to be implemented in wireless mesh networks. The algorithms proposed in wired networks by Orda et al [14] are limited for single end-to-end QoS requirement, additionally, the proposed optimization algorithm uses an unspecified generic global optimization objective. Therefore, only PLA and ELA qualify for comparison.

Figs. 7 and 8 show the change in the number of admitted requests as the required end-to-end delay and end-to-end violation probability are varied. The performance of the ODP is carried out over paths with different number of hops. As the figures show, ODP algorithm has better performance than PLA and ELA. The ODP algorithm admitted about one time and a half the number of connections. This is because the ODP algorithm uses optimization to partition the end-to-end delay and the end-to-end violation probability bounds. This makes a better balance of the load along the path. As a result a larger number of WMN requests are admitted, since ODP minimizes the probability of generating future bottleneck links along the path. Fig. 9 presents the results of changing the number of hops along the paths at a required end-to-end delay and violation probability of 50 ms and 10^{-5}, respectively. For all algorithms, the number of admitted requests decreases as the number of hops increases. This is because the same amount of QoS requirements are to be partitioned over the paths with smaller number of hops as the paths with larger number of hops. Accordingly, for smaller number of hops, the amount of link delay is larger than that for larger number of hops – translated as a smaller amount of reserved bandwidth. Fig. 9 shows that ODP still outperforms ELA and PLA for the same reason which is the ability of ODP to balance the load along the path. The ODP algorithm balances the load over the path by assigning larger delay over

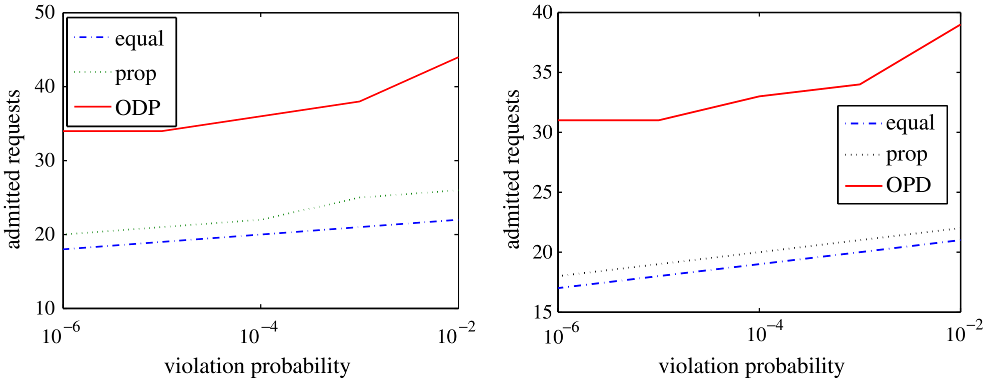


(a) 5hops (b) 7hops

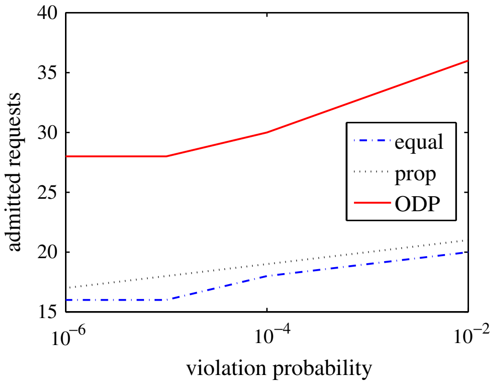


(c) 9hops

Fig. 7. Number of admitted requests versus end-to-end delay at a required violation probability of 10^{-6}, the tree depth is (a) 5 hops, (b) 7 hops, and (c) 9 hops.



(a) 5 hops (b) 7 hops



(c) 9 hops

Fig. 8. Number of admitted requests versus the violation probability at a required end-to-end delay of 60 ms, the tree depth is (a) 5 hops, (b) 7 hops, and (c) 9 hops, respectively.

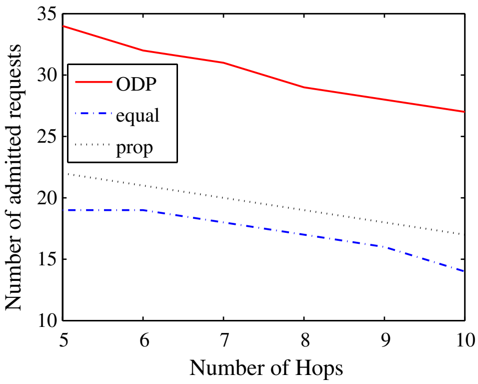


Fig. 9. Number of admitted requests versus different path depths for a required end-to-end delay of 50 ms and a required violation probability of 10^{-5}.

the links of smaller capacity. Consequently, the ODP algorithm avoids generating potential bottleneck links, improving links utilization.

6. Conclusions and future work

We proposed an approach to map end-to-end QoS requirements into link QoS requirements. To this end, we proposed a partitioning algorithm that is capable of simultaneously partitioning multiple end-to-end QoS requirements in WMNs onto link requirements. Such a partitioning scheme, the first scheme proposed for WMNs, represents a significant and pioneering step towards providing QoS guarantees at the link and the end-to-end levels. The proposed optimal delay partitioning (ODP) algorithm provides end-to-end QoS delay guarantees by optimally partitioning the end-to-end QoS requirement into link QoS requirements. We define the end-to-end QoS requirement as the pair of the required delay and the probability required to guarantee this delay. The algorithm partitions the statistical delay pair simultaneously into a set of link statistical delays such that the upper bound of the violation probability is minimized. The ODP algorithm does not assume any knowledge about the traffic characterization or any restrictions on the number of connections multiplexed at the wireless link. However, the delay distribution of the link delays are required to be known without any restrictions on their type. Extensive simulation verified the effectiveness of the algorithm compared to two representative QoS partitioning algorithms.

As for future work, we intend to implement the proposed algorithm over a WMN testbed to practically study its effectiveness. Furthermore, we will investigate the interaction of the proposed algorithm with various scheduling schemes, specially opportunistic schedulers.

References

[1] IEEE Standard 802.11 Working Group. IEEE 802.11e-2005, IEEE standard for local and metropolitan area networks specific requirements – part 11: Wireless lan medium access control (MAC) and physical layer (PhY) specifications mac enhancements for qos. Standard, 2005.

[2] K. Wang, P. Ramanathan, End-to-end delay assurances in multihop wireless local area networks , in: Globecom, 2003, pp. 2962–2966.

[3] R. Draves, J. Padhye, B. Zill, Routing in multi-radio multi-hop wireless mesh networks, in: MobiCom, 2004, pp. 114–128.

[4] D. De Couto, D. Aguayo, J. Bicket, R. Morris, High-throughput path metric for multi-hop wireless routing, in: MobiCom, 2003.

[5] H. Rio, D. Sarkar, L. Stelling. Packet delay estimation for ad hoc networks, in: International Conference on Computer Communications and Networks (ICCCN), October 2005, pp. 289–296.

[6] G. Narlikar, G. Wilfong, L. Zhang, Designing multihop wireless backhaul networks with delay guarantees, in: Infocom, 2006.

[7] H. Shetiya, V. Sharma, Algorithms for routing and centralized scheduling ieee 802.16 mesh networks, in: Proceedings of the Wireless Communications and Networking Conference, April 2006, pp. 147–152.

[8] H. Shetiya, V. Sharma, Algorithms for routing and centralized scheduling to provide qos in ieee 802.16 mesh networks, in: International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems, October 2005, pp. 140–149.

[9] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, E. Knightly, Distributed multi-hop scheduling and medium access with delay and throughput constraints, in: ACM SIGMOBILE, 2001, pp. 200–209.

[10] R. Nagarajan, J. Kurose, D. Towsley, Local allocation of end-to-end quality-of-service in high-speed networks, in: Proceedings of IFIP Workshop on Performnace analysis of ATM Systems, January 1993, pp. 99–118.

[11] V. Firoiu, D. Towsley, Call admission and resource reservation for multicast sessions, in: Proceedings of INFOCOM, 1996, pp. 94–101.

[12] D. Roz, Y. Shavvitt, Optimal partition of QoS requirements with discrete cost functions, IEEE Journal on Selected Areas in Communication 18 (12) (2000) 2593–2602.

[13] A. Orda, A. Sprintson, A scalable approach to the partition of QoS requirements in unicast and multicast, IEEE/ACM Transactions on Networking 10 (1) (2002) 102–114.

[14] D. Lorenz, A. Orda, D. Raz, Optimal partition of QoS requirements for many-to-many connections, in: Proceedings of INFOCOM 2003, vol. 3, March 2003, pp. 1670–1679.

[15] F. Ergun, R. Sinha, L. Zhang, QoS routing with performance-dependent costs, in: Proceedings of INFOCOM, 2000, pp. 137–146.

[16] I. Atov, H. Tran, R. Harris, Efficient QoS partition and routing in multi-service IP networks, in: Proceedings of the 22nd IEEE International Conference on Automation and Information (ICAI), May 2003, pp. 435 – 441.

[18] B. Waxman, Routing of multipoint connections, IEEE Journal of Selected Areas in Communication 6 (9) (1988) 1617–1622.

[19] A. Parekh, R. Gallager, A generalized processor sharing approach to flow control in integrated services networks: the multiple node case, IEEE/ACM Transactions on Networking 2 (1994) 137–150.

[20] L. Georgiadis, R. Guerin, V. Peris, K. Sivarajan, Efficient network QoS provisioning based on per node traffic shaping, IEEE/ACM Transactions on Networking 4 (1996) 482–501.

[21] V. Firoiu, D. Towsley. Call admission and resource reservation for multicast sessions. Technical Report TR 95-17, University of Massachusetts, September 1995.

[22] L. Choudhurgy, M. Lucantondi, Numerical computation of large number of moments with applications to asymptotic analysis, in: Proceedings of the International Teletraffic Congress Sponsored Seminar on Teletraffic Analysis Methods for Current and Future Telecom Networks, Bangalore, India, 1993, pp. 113–120.

[23] L. Choudhurgy, M. Lucantondi, Numerical computation of the moments of a probability distribution from its transform, Operation Reseasrch To appear (1995).

[24] IEEE Standard 802.16 Working Group, IEEE 802.16-2004 Standard for Local and Metropolitan Area Networks, Part 16, Air Interface for Fixed Broadband Wireless Access Systems, June 24, 2004.