# Bandwidth Estimation and Distributed Traffic Regulation in Wireless Local Area Networks

*Shahrokh Valaee*

*Abstract*— **In this paper, we introduce distributed traffic regulation for wireless local area networks (WLANs) operating in ad hoc mode. We use the concept of service curve to determine the minimum guaranteed service given to a backlogged connection. In a WLAN, the service curve depends on the number of active nodes, their traffic load in active periods, and the back-off procedure used for contention resolution. We will show that service curve can be used for distributed traffic regulation. We use some of the data packets, denoted as the probing packets, to estimate the service curve. The call is accepted if the service curve is higher than a preselected threshold, called the universal lower bound. The universal lower bound is independent of the number of nodes and traffic fluctuation and acts as a worst-case reference point for the network performance.**

**Keywords:** System design, deterministic network calculus, WLANs, ad hoc networks, service curve, traffic regulation.

## I. Introduction

Measurement-based algorithms have recently been proposed for call admission control in the Internet, see [1] and references therein. In this paper, we propose a measurement-based bandwidth estimation and traffic regulation for wireless local area networks (WLANs) that use ad hoc mode of operation. Examples of such networks can be found in vehicular communications, in which neighbouring vehicles form an ad hoc network to exchange safety packets and private data. The IEEE 802.11p is a new initiative to standardize vehicle-to-vehicle and vehicle-to-roadside communications. In the absence of roadside units, vehicles form ad hoc clusters and share safety and private information. This paper proposes a traffic regulation technique that can be used in single-hop ad hoc networks. In our approach, a node with a pending traffic probes the network over an appropriate time interval and uses the collected information to regulate the input traffic.

We use the concept of *service-curve* from *network calculus* [2] [3], defined as the minimum guaranteed service for a call over its backlogged interval in both deterministic [4] [5] and stochastic [6] setting. In [7], we used service curve for call admission control in a single-hop ad hoc network. That technique can be used for quality-of-service provisioning in single-hop ad hoc networks [8]. In the present paper, we extend our earlier technique and propose a distributed traffic regulator.

In this paper we compute the service curve using the time packets spend at the head of the queue. The time that a packet spends at the head of the queue is the *service time* that reflects the load of the network. Loosely, if the network is lightly loaded, the service time will be short and if the network is heavily loaded, the service time will be long. We will define the service curve with a line that illustrates the total service time as a function of the number of packets.

Once the service curve is estimated, we compare it to a *universal lower bound* and regulate the traffic so that the induced service curve — after the regulated traffic enters the network — remains above the universal lower bound. The universal lower bound is a linear curve independent of the number of sources and their activities and reflects the worst-case behavior of the network. In this paper, we do not discuss how the universal lower bound should be selected, rather we assume that it is given by the network designer. However, it should be noted that the selection of the universal lower bound is an off-line procedure that depends on the quality-of-service that is expected to be supported by the network.

In the proposed technique, the input traffic is decomposed into *conforming* and *nonconforming* packets. Transmission of a conforming packet keeps the service curve above the universal lower bound and transmission of a nonconforming packet moves the service curve below the universal lower bound. We will show that if an appropriate amount of delay is added to a nonconforming packet, it will become conforming. Hence, the proposed service curve provisioning technique creates a mechanism to smooth bursty traffic and improve network performance.

This work relates to the concept of scheduling and fairness in the MAC layer [9][10]. These techniques alter the size of the backoff window to achieve the fair share for each user. They use either explicit data shared among the nodes or implicit information collected from the behavior of the present node. In [9], the authors propose a distributed approach to fair scheduling in wireless LANS. They vary the length of the backoff window to obtain the *generalized processor sharing* (GPS) [4] fairness. The total waiting time spent at the head of the transmit queue is used in [9] to set the backoff interval. We use this time to measure the network load. In [11], an asymptotically optimal algorithm has been proposed for IEEE 802.11 WLANs. A metric, called slot utilization, has been defined and has been used, along with the average size of transmitted frames, to set the length of the backoff window. Our work uses the waiting time at the head of the transmit queue to estimate the network load. Our metric is the service curve.

In this paper, we assume a single-class network, that is, all connections have the same bandwidth requirements. This will result in *max-min* fairness. The proposed ap-

The author is with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, Ontario, Canada, M5S 3G4, Email: valaee@comm.utoronto.ca

978-1-4244-2644-7/08/$25.00 ©2008 IEEE

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.

proach can easily be extended to multi-class networks with the GPS fairness. A weighted version of the true service curve be compared to the universal lower bound to satisfy the GPS fairness. The weight depends on the class to which the input flow belongs. In the sequel, we will only discuss the single-class case.

## II. Problem Formulation

We assume that data packets are tagged with a single-bit that shows whether or not they have arrived at a backlogged queue. A data packet entering an already backlogged queue is tagged and can be used as a *probing packet* to estimate the available bandwidth. Throughout the paper we refer to such packets as the probing packet, however we note that these are indeed the data packets also used as bandwidth estimator. A packet, which is not the first packet in a backlogged period, can be used as a probing packet. At each node, exit time between two consecutive packets indicates the level of congestion in the network. If the network is lightly loaded, the temporal distance between the exit instances is small. Otherwise, if the network is heavily loaded, the temporal distance between the exit instances is large.

It is necessary to note that probing packets issued by a node can be used by all other nodes to estimate the service curve. Therefore, all active or inactive nodes can estimate the service curve by listening to the wireless channel and measuring the time difference between probing packets transmitted by any active node.

Let $\{\tau_i\}$ denote the time instant at which packets exit the node. If the $i$th packet is a probing packet, the total delay of the packet at the head of the queue is $\delta_i \triangleq \tau_i - \tau_{i-1}$. Note that the $(i-1)$th packet does not have to be a probing packet. The delay $\delta_i$ can be written as

$$\delta_i = w_i + d \qquad (1)$$

where $w_i$ is the total waiting time spent at the head of the queue of the probing node for the $i$th probing packet and $d$ is the transmission time of the probing packet. $w_i$ is a function of the number of backlogged nodes, the size of the transmitted packets the channel characteristics in the physical layer, and the length of backoff window. Note that in general, $w_i$ is a function of $d$ since the number of active sources might increase during the transmission time of the probing packet. However, for small $d$, we can assume that $w_i$ is independent of $d$. We will use computer simulations to show that with proper weighting, it is possible to obtain a service curve that is independent of the size of the probing packets.

In the DCF mode of 802.11, $d = t_{\text{OH}} + t_{\text{PRB}}$ where $t_{\text{PRB}}$ is the transmission time of the probing packet and $t_{\text{OH}}$ is the total overhead transmission time defined as (for the system using RTS-CTS signalling)

$$t_{\text{OH}} \triangleq t_{\text{DIFS}} + 3t_{\text{SIFS}} + t_{\text{RTS}} + t_{\text{CTS}} + t_{\text{ACK}} \qquad (2)$$

where $t_{\text{DIFS}}$ is the length of the DIFS period, $t_{\text{SIFS}}$ is the length of the SIFS period, $t_{\text{RTS}}$ is the transmission time of RTS, $t_{\text{CTS}}$ is the transmission time of CTS, and $t_{\text{ACK}}$ is the transmission time of ACK. Let $b$ be the size of a probing packet and $C$ be the transmission rate of the network. Then, we have $t_{\text{PRB}} = \frac{b}{C}$.

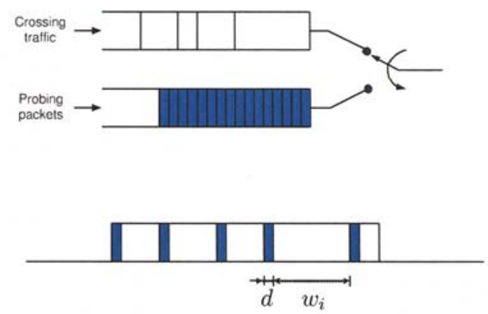


Fig. 1. The whole network is modelled by two queues and a round-robin scheduler. The inter-arrival time between two consecutive probing packets is occupied by a packet of the crossing traffic and represents the total load of the network.

An input flow at each node views the whole network as a system consisting of two parts: the buffer at which the flow is stored and the outer world where the crossing traffic is being transmitted. The present flow should compete with the crossing traffic for the available bandwidth. Therefore, from the perspective of the bandwidth estimator, we can model the whole network by two virtual queues and a round-robin scheduler. Fig. 1 illustrates this model. In this model, the size of the packets of the crossing traffic, denoted by $w_i$, includes the actual packet transmission time, idle times, collisions, and backoff periods. In fact, the size of the packets of crossing traffic encompasses all bandwidth consumptions due to source activities, contention resolution, and possible delays induced by the activity of hidden terminals. If the network is lightly loaded, $w_i$ will be small. On the contrary, if the network is heavily loaded, $w_i$ will usually be large. In the sequel, we will use the sequence $\{w_i\}$ to devise a distributed traffic regulator.

Let $W_i^{(k)}, k = 1, \ldots, K$ be the total waiting delay at the head of the probing queue for transmitting a batch of $k$ probing packets starting at the $i$th packet; $K$ is the maximum size of a batch of probing packets used in traffic regulator. Then, $W_i^{(k)}$ can be represented by

$$W_i^{(k)} \triangleq \sum_{j=i}^{k+i-1} w_j. \qquad (3)$$

We consider a stationary environment in which the probability distribution function of the delay elements $\{w_i\}$ is constant. We further assume that $\{w_i\}$ is an independent identically distributed (i.i.d.) sequence.

Fig. 2 illustrates the empirical probability density function (pdf) of the sequence $\{w_i\}$ in a simulated WLAN. Notice that the empirical pdf closely imitates an Exponential distribution. The slope of the line can be used to estimate the parameter of the exponential distribution.

Let $w_i$ be an Exponentially distributed random variable with parameter $\mu$. The parameter $\mu$ depends on the number of active sources, the volume of the crossing traffic, and

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.

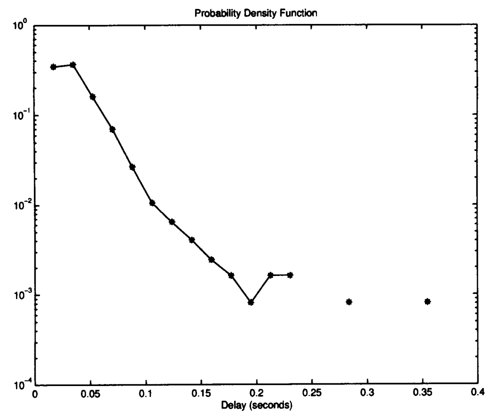


Fig. 2. The empirical probability density function of the delay sequence $\{w_i\}$ of a simulated ad hoc network in a logarithmic scale.

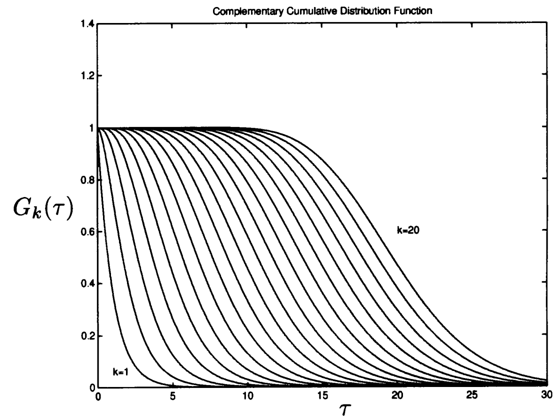


Fig. 3. The complementary probability distribution function of an Erlang distributed random variable as a parametric function of $k$.

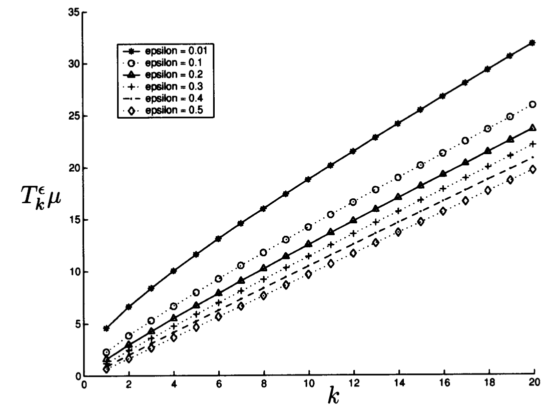


Fig. 4. The normalized $\epsilon$-percentile delay curve $T_k^\epsilon \mu$ as a parametric function of $\epsilon$.

the activity of hidden and exposed terminals. A small $\mu$ represents a lightly loaded network and a large $\mu$ represents a heavily loaded network. With this assumption, we have

$$E\{w_i\} = \frac{1}{\mu} \tag{4}$$

$$\sigma^2\{w_i\} = \frac{1}{\mu^2} \tag{5}$$

where $E\{X\}$ and $\sigma^2\{X\}$ are respectively the mean and the variance of the random variable $X$. Similarly, $\{W_i^{(k)}\}$ is a sequence of Erlang distributed random variables with $k$ degrees of freedom and we have

$$E\{W_i^{(k)}\} = \frac{k}{\mu} \tag{6}$$

$$\sigma^2\{W_i^{(k)}\} = \frac{k}{\mu^2} \tag{7}$$

In the present paper, the service curve will be obtained using the fact that the waiting time for a sequence of $k$ packets in the probing queue is $W_i^{(k)}$ seconds. The service curve is defined as a percentile of the delay random variable. For a given $0 < \epsilon < 1$, define

$$T_k^\epsilon \triangleq \inf\left\{\tau \,\middle|\, G_k(\tau) \le \epsilon\right\} \tag{8}$$

where

$$G_k(\tau) \triangleq Pr(W_i^{(k)} > \tau) \tag{9}$$

is the "complementary cumulative distribution function". For Exponentially distributed $w_i$ with parameter $\mu$, we have

$$T_1^\epsilon = -\frac{\ln(\epsilon)}{\mu} \tag{10}$$

$$T_k^\epsilon = \frac{1}{\mu}\inf\left\{\tau \,\middle|\, G_k(\tau) \le \epsilon\right\} \tag{11}$$

where

$$G_k(\tau) = \sum_{j=0}^{k-1} \frac{\tau^j}{j!} e^{-\tau}. \tag{12}$$

Fig. 3 illustrates $G_k(\tau)$ as a function of $k$. The intersection of the curves in Fig. 3 and a horizontal line with the height $\epsilon$ gives $T_\epsilon^k$ in (11). Fig. 4 illustrates the normalized $T_\epsilon^k$. Note that $T_\epsilon^k$ is approximately a linear function of $k$.

From Fig. 4, we can define the service curve as an affine function that is a best fit to the set of pairs $(T_k^\epsilon, k), k = 1, \ldots, K\}$. Using a linear service curve complies with the fact that in a single priority network, all nodes receive identical service (bandwidth). Ideally, the whole bandwidth should be equally divided among all nodes.

For simplicity, we will define the service curve as a function that for all $k = 1, \ldots, K-1$ connects $(T_k^\epsilon, k)$ and $(T_{k+1}^\epsilon, k+1)$ to each other. Therefore, the $\epsilon$-*effective service curve* is defined as

$$S_\epsilon(t) \triangleq \frac{t - T_k^\epsilon}{T_k^\epsilon - T_{k-1}^\epsilon} + k, \qquad \text{for } T_{k-1}^\epsilon \le t \le T_k^\epsilon. \tag{13}$$

From this definition, we have

$$S_\epsilon(T_k^\epsilon) = k. \tag{14}$$

Using $T_k^\epsilon \le T_{k+1}^\epsilon$, we conclude that $S_\epsilon(t)$ is a nondecreasing function.

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.

In (13), if we let $\epsilon = 0$, we will have the *maximum service curve* defined by

$$S_0(t) = \frac{t - T_k^0}{T_k^0 - T_{k-1}^0} + k \quad \text{for } T_{k-1}^0 \le t \le T_k^0 \tag{15}$$

where $T_k^0 \triangleq \max_i\{W_i^{(k)}\}$.

Similarly, the *mean service curve* is defined as

$$S_m(t) \triangleq \frac{t - T_k^m}{T_k^m - T_{k-1}^m} + k \tag{16}$$

where $T_k^m \triangleq E\{W_i^{(k)}\}$. Assuming an Exponentially distributed sequence $\{w_i\}$, the mean service curve will be

$$S_m(t) = \mu t. \tag{17}$$

In words, the mean service curve is a linear function with the slope $\mu$.

## III. Traffic Regulation

Upon the transmission of the new burst, the true service curve should not be moved below the universal lower bound. This can be performed by considering all packets inside a backlogged interval. Consider the first packet of a backlogged period. If the traffic is admitted into the network, the total delay for the first packet will be

$$D_1 = w + d_1 \tag{18}$$

where $w$ is a generic random variable indicating the waiting time inside the queue and $d_1$ is the transmission time of the first packet, which can be written as

$$d_1 = t_{\text{OH}} + \frac{L_1}{C}, \tag{19}$$

where $t_{\text{OH}}$ is defined in (2). We assume that $w$ is independent of the length of the transmitted packet $L_1$. Using the measurements in the probing phase, we conclude that with probability $1 - \epsilon$, the total delay will be smaller than $T_1^\epsilon + d_1$.

The first packet is called *conforming* if

$$T_1^\epsilon + d_1 \le \bar{T}_1^\epsilon \tag{20}$$

where $\bar{T}_1^\epsilon$ is the time index of the universal lower bound at 1, that is

$$\bar{S}_\epsilon(\bar{T}_1^\epsilon) = 1. \tag{21}$$

The packet is indicated as *nonconforming* if (20) is not satisfied.

The test of conformance should also be performed for all subsequent packets. Assume that the first packet is conforming and it will be transmitted at the due time over the wireless channel. The total delay for the second packet can then be represented by

$$D_2 = w_1 + w_2 + d_1 + d_2 \tag{22}$$

where $w_1$ and $w_2$ are two random variables indicating the competition intervals, and $d_1$ and $d_2$ are the transmission time of the first and the second packets, respectively. Our probing scheme indicates that with probability $1 - \epsilon$, the total competition time will be bounded by $T_2^\epsilon$. We will use this fact to compare the total delay to the corresponding index in the universal lower bound. In particular, we call the second packet conforming if

$$T_2^\epsilon + d_1 + d_2 \le \bar{T}_2^\epsilon \tag{23}$$

and nonconforming if (23) is violated.

If the input traffic is insensitive to delay, it is usually advised to use a regulator to reduce the traffic burstiness. In this section, we will show that it is possible to apply a traffic regulator on delay-insensitive connections using the universal lower bound. Again, the objective would be to maintain the true service curve above the universal lower bound. We will instrument a regulating algorithm that will shape the nonconforming traffic.

Assume that the first packet is nonconforming, that is

$$T_1^\epsilon + d_1 > \bar{T}_1^\epsilon. \tag{24}$$

Since the packet is nonconforming, it cannot be transmitted in the first contention period. The regulator will now compare the estimated service curve and the universal lower bound in the second contention period. In fact, the regulator checks the validity of the following inequality

$$T_2^\epsilon + d_1 \le \bar{T}_2^\epsilon. \tag{25}$$

If the inequality holds, the regulator will learn that postponing the packet transmission to the second contention period will change the status of the packet into a conforming packet. Therefore, the regulator simply defers the packet transmission time to the next contention period. If (25) does not hold, the regulator will evaluate the conformance test in the third contention period and this process will continue until a valid condition is found. The packet will then be transmitted in the corresponding contention period. If no valid condition is found for all instants, the packet should be dropped.

Now assume that two packets are backlogged in the transmit queue. Let the first packet be considered as conforming. The second packet is denoted as conforming if (23) is satisfied. If the second packet is nonconforming, it will be delayed and will be examined with the contention instant in the service curve, that is

$$T_3^\epsilon + d_1 + d_2 \le \bar{T}_3^\epsilon. \tag{26}$$

If (26) holds the second packet is conforming, otherwise the process should continue with investigating the subsequent instants. The second packet is then transmitted in the first conforming instant.

Another issue that we have to address is the amount of delay imposed on the nonconforming packets. Assume that the first packet at the head of the queue is nonconforming, but will become conforming if it is transmitted in the second contention period. That is

$$\bar{T}_1^\epsilon - T_1^\epsilon < d_1 \le \bar{T}_2^\epsilon - T_2^\epsilon. \tag{27}$$

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.

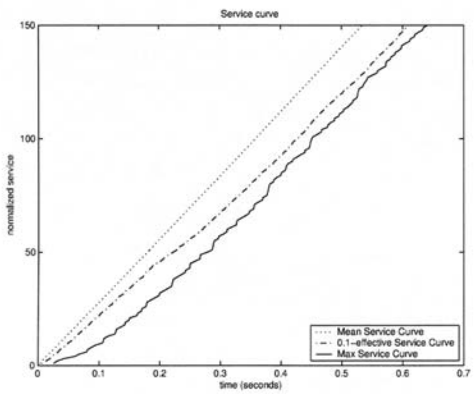


Fig. 5. The service curve in the presence of 1 UDP crossing connection.

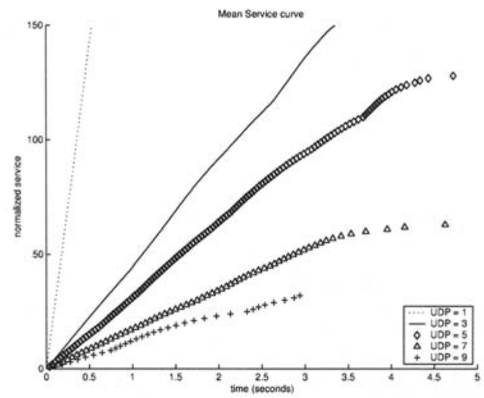


Fig. 6. The mean service curve as a function of the number of UDP connections.

This requires that the node be involved in the competition for the bandwidth but refrain from sending the first packet when it wins the competition. In fact, the node should compete for two backoff periods before it can send its first packet. As noted earlier this requirement will impose an extra delay on the packet. The excess delay can be computed as follows. Since it has been assumed that the status of the network will remain stationary over the probing phase and data transmission time, with probability $(1-\epsilon)$, the waiting time at the head of the queue will be smaller than $T_1^\epsilon$ seconds. Therefore, a conservative approach would be to postpone the transmission of the first packet for $T_1^\epsilon$ seconds. A similar approach can be used for subsequent packets. If a packet arrives at the $i$th position of a backlogged period and if it has been denoted as conforming in the $m$th $(m \geq i)$ backoff instant, the packet should be delayed at the head of the queue for $T_m^\epsilon - T_i^\epsilon$ seconds. Although this approach may not be optimal, the realization of the technique is very simple and hence practically conducive.

## IV. Simulation Results

We have used *ns*-2 network simulator to study our technique. In the first simulation, we consider 22 nodes randomly located inside an area of $250 \times 250$ meters. The transmission power of the nodes is large enough so that they all can communicate over single-hop links. We study a scenario with 10 UDP connections using FTP agents with constant-bit-rate traffics. Two nodes are used for probing and the rest (20 nodes) will simulate the crossing traffic. The crossing connections are activated at $t = 0.1$ second and the probing starts at $t = 0.5$ second. The total probing time is 10 seconds. The mean service curve, the 0.1-effective service curve, and the maximum service curve are shown in Fig. 5 when there is only 1 active crossing traffic between two nodes. Note that the mean service curve is approximately a linear function. This is an interesting observation and justifies the assumption of the exponentially distributed waiting times.

In order to show that the service curve truly represents the load of the network, we have simulated a scenario with variable number of crossing traffics. The traffic has changed from 0 to 10 UDP connections. For each scenario, we have measured the mean service curve. The results have been illustrated in Fig. 6. Note that the service curve decreases with increasing the traffic of the ad hoc network. This property indicates that the service curve is an appropriate measure of the network load.

Next, we study the ratio of nonconforming packets to total traffic in the network. We use 10 nodes with 50 randomly generated traffic patterns to calculate the percentage of the non-conforming packets as the offered load at the probing node increases. We have used a variable-bit-rate traffic at the probing node where the packet size is uniformly distributed [0,512], and the packet inter-arrival times are exponentially distributed. The offered load, at the probing node, is increased by decreasing the packet inter-arrival time. Fig. 7 illustrates the percentage of the nonconforming traffic and the induced delay as a function of the offered load. Note that, as expected, the percentage of nonconforming traffic and the delay increase with the load of the network. A call is acceptable if the total delay is bounded below certain threshold. This corresponds to a specific percentage of the nonconforming traffic. Fig. 7 shows that the percentage of nonconforming traffic is an appropriate QoS metric.

Fig. 8 illustrates the mean service curve as a function of the probing packet size. Note that the size of the probing packet is directly related to the measured service curve. In fact, increasing the probe size reduces the service curve. This is expected since the probing packets consume bandwidth. Fig. 8 indicates that the probing traffic will affect the performance of the network. Furthermore, we notice that the amount of delay is almost linearly related to the probing packet size. In other words, doubling the packet size has almost doubled the waiting time. We conclude that the service curve is directly related to the size of the input packets and the conformance test is valid.

Fig. 9 illustrates the normalized mean service curve as a function of the probing packet size. In this figure, the

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.

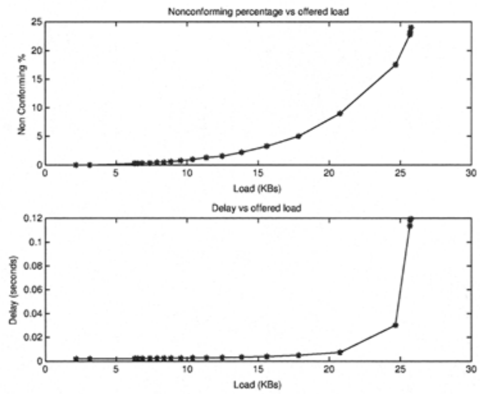


Fig. 7. The percentage of the nonconforming traffic and the induced delay as a function of the offered load.

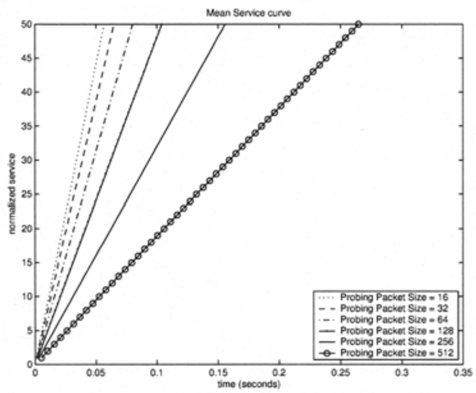


Fig. 8. The mean service curve as a function of the probing packet size.

mean service curve for each probe size has been divided by the size of the probing packet in the physical layer. This corresponds to the normalization procedure in the GPS algorithm [4]. Note that all curves almost overlap. Therefore, the normalized service curve is independent of the size of the probing packet.

## V. Conclusion and Discussion

In this paper, we have introduced a novel approach to traffic regulation in wireless local area networks. We have used the concept of service curve. Data packets arriving at a backlogged queue can be used to estimate the amount of service available for a prospective connection.

We have shown that the service curve can be represented by an affine curve. The slope of the line indicates the amount of available service. For a lightly loaded network, the slope of the service curve is large and for a heavily loaded network the slope is small. We have used this property to devise a traffic regulator. The service curve is compared to a fixed threshold, called the universal lower bound. Upon the arrival of a burst, the true service curve is estimated and compared to the universal lower bound. The input traffic is categorized to conforming and nonconforming packets. The conforming packets retain the true service curve above the universal lower bound and the nonconforming packets move the true service curve below the universal lower bound. The call is accepted if the true service curve does not stay below the universal lower bound for a long period of time. We have also shown that if an appropriate amount of delay is imposed on the nonconforming packets, they will become conforming and can be transmitted safely over the network. The amount of delay is obtained from the true service curve. Therefore, the proposed technique can also be used to regulate delay-tolerant input traffics.

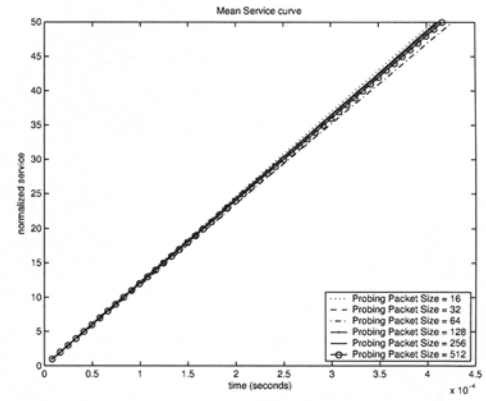


Fig. 9. The normalized mean service curve as a function of the probing packet size.

## References

[1] J. Liebeherr, M. Fidler, and S. Valaee, "A min-plus system interpretation of bandwidth estimation," in *Proceeding IEEE INFOCOM*, 2007.

[2] C. S. Chang, *Performance Guarantees in Communication Networks.* Springer Verlag, 2000.

[3] J. Y. Le Boudec and P. Thiran, *Network Calculus.* Springer Verlag, 2001.

[4] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services network: the single node case," *IEEE/ACM Trans. Networking*, vol. 1, pp. 334–357, June 1993.

[5] R. L. Cruz, "Quality of service gurantees in virtual circuit switched networks," *IEEE J., Select., Areas Commun.*, vol. 13, pp. 1048–1057, August 1995.

[6] J. Liebeherr, S. Patek, and A. Burchard, "Statistical per-flow service bounds in a network with aggregate provisioning," in *Proceeding IEEE INFOCOM*, pp. 1680–1690, 2003.

[7] S. Valaee and B. Li, "Distributed call admission control for ad hoc networks," *in Proceedings of IEEE Vehicular Technology Conference*, vol. 2, pp. 1244 –1248, Sept. 2002.

[8] H. Zhu, M. Li, I. Chlamtac, and B. Prabhakaran, "A survey of quality of service in ieee 802.11 networks," *IEEE Wireless Communications*, vol. 11, pp. 6–14, August 2004.

[9] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed fair scheduling in a wireless LAN," *MOBICOM*, 2000.

[10] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly, "Ordered packet scheduling in wireless ad hoc networks: Mechnisms and performance analysis," *MOBIHOC*, pp. 58–70, 2002.

[11] L. Bononi, M. Conti, and E. Gregori, "Design and performance evaluation of an asymptotically optimal backoff algorithm for IEEE 802.11 wireless LANs," *Proceedings of the 33rd Hawaii International Conference on System Science*, 2000.

Authorized licensed use limited to: The University of Toronto. Downloaded on December 23, 2009 at 17:39 from IEEE Xplore. Restrictions apply.