Quality of Service Support in Wireless Ad Hoc Networks Connected to Fixed DiffServ Domains

M.C. Domingo and D. Remondo

Telematics Eng. Dep., Catalonia Univ. of Technology (UPC)
Av. del Canal Olímpic s/n.
08860 Castelldefels (Barcelona), Spain
{cdomingo, remondo}@mat.upc.es

Abstract. This paper analyzes the provision of end-to-end Quality of Service between nodes in a mobile ad hoc network and a fixed IP network that supports Differentiated Services. The ad hoc network incorporates the Stateless Wireless Ad Hoc Networks (SWAN) model to perform admission control for real-time traffic flows. We propose a new protocol, named DS-SWAN (Differentiated Services-SWAN), where end-to-end delays and loss rates of real-time traffic are monitored continuously at the destination nodes in the fixed network and at the edge routers respectively. In this way, nodes in the ad hoc network are warned when congestion is excessive for the correct functioning of a real-time application (specifically, Variable Bit Rate Voice-over-IP), so that the nodes restrain best-effort traffic in order to favour real-time flows. The results indicate that DS-SWAN significantly improves end-to-end delays without starvation of background traffic, adapting itself to changing traffic and network conditions in a relatively small ad hoc network. Besides, we compare different notification procedures in DS-SWAN aimed to improve scalability.

1 Introduction

Ad hoc networks [1] are formed by mobile devices that are able to communicate without having to resort to a pre-existing network infrastructure. In an ad hoc network, terminals can communicate with each other even if they are out of range because they can reach each other via intermediate nodes acting as routers.

At first glance, it may seem incoherent to deal with Quality of Service (QoS) support in such dynamic systems with unreliable wireless links. However, some authors have presented proposals to support QoS in wireless ad hoc networks including QoS oriented MAC protocols [2], QoS aware routing protocols [3] and resource reservation protocols [4]. Moreover, a flexible QoS model for mobile ad hoc networks has been proposed in [5]. This paper explores the dynamics of a system where a resource reservation mechanism within the ad hoc network co-operates with the Differentiated Services (DiffServ) domain of the fixed network to which the ad hoc network is attached. The aim of this work is to investigate whether aiding resource reservation mechanisms at the ad hoc network by DiffServ based QoS support could yield satisfactory end-to-end QoS properties.

I. Niemegeers and S. Heemstra de Groot (Eds.): PWC 2004, LNCS 3260, pp. 262–271, 2004.

Specifically, we consider a scenario where an ad hoc network is connected via a single gateway to a fixed IP network that supports DiffServ. The ad hoc network incorporates the SWAN [10] scheme to provide QoS. The authors in [10] study the behavior of CBR voice traffic in this context but voice transmission of Variable Bit Rate (VBR) real-time traffic has not yet been analyzed. There are also some works related to voice transmission in IEEE 802.11, but only very few in the ad hoc mode [6]. To our knowledge, there has been little or no prior work on analyzing voice transmission between an ad hoc network and a fixed IP network providing end-to-end QoS for real-time traffic that shares resources with background traffic.

The paper is structured as follows: Section 2 describes related work about how to support QoS in mobile ad hoc networks. Section 3 presents the protocol that supports end-to-end QoS in the mentioned context, which we have named DS-SWAN (DiffServ-SWAN). Section 4 presents and shows our simulation results. Finally, Section 5 concludes this paper.

2 QoS in Mobile Ad Hoc Networks

In a mobile environment it is difficult to provide a certain QoS because the network topology changes dynamically and in wireless networks the packet loss rates are much higher and more variable than in wired networks. Some authors have adapted the DiffServ [7] model for mobile ad hoc networks [8]. However, when DiffServ is compared with the SWAN model in an isolated ad hoc network, SWAN clearly outperforms DiffServ in terms of throughput and delay requirements [9]. For this reason, we will concentrate on the SWAN scheme.

2.1 SWAN

SWAN is a stateless network scheme that has been specifically designed to provide end-to-end service differentiation in wireless ad hoc networks employing a best-effort distributed wireless MAC [10]. It distinguishes between two traffic classes: real-time UDP traffic and best-effort UDP and TCP traffic.

A classifier (see Fig. 1) differentiates between real-time and best-effort traffic. Then, a leaky-bucket traffic shaper handles best-effort packets at a previously calculated rate, applying an AIMD (Additive Increase Multiplicative Decrease) rate control algorithm. Every node measures the per-hop MAC delays locally and this information is used as feedback for the rate controller. Every T seconds, each device increases its transmission rate gradually (additive increase with increment rate of c bit/s) until the packet delays at the MAC layer become excessive. As soon as the rate controller detects excessive delays, it reduces the rate of the shaper with a decrement rate (multiplicative decrease of r %).

Rate control restricts the bandwidth for best-effort traffic so that real-time applications can use the required bandwidth. On the other hand, the bandwidth not used by real-time applications can be efficiently used by best-effort traffic. The total best-effort and real-time traffic transported over a local shared channel is limited below a certain ‘threshold rate’ to avoid excessive delays.

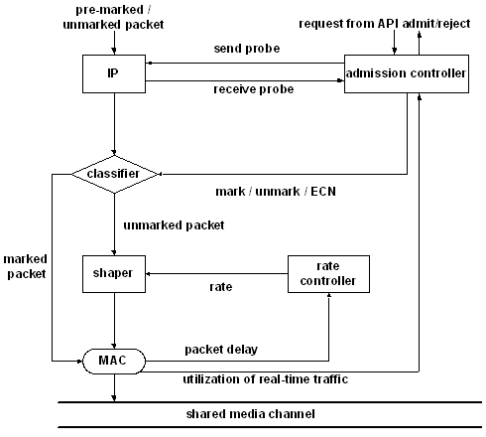

Fig. 1. SWAN model

SWAN also uses sender-based admission control for real-time UDP traffic. The rate measurements from aggregated real-time traffic at each node are employed as feedback. This mechanism sends an end-to-end request/response probe to estimate the local bandwidth availability and then determine whether a new real-time session should be admitted or not. The source node is responsible for sending a probing request packet toward the destination node. This request is a UDP packet containing a "bottleneck bandwidth" field. All intermediate nodes between the source and destination must process this packet, check their bandwidth availability and update the bottleneck bandwidth field in the case that their own bandwidth is less than the current value in the field. The available bandwidth can be calculated as the difference between an admission threshold and the current rate of real-time traffic. The admission threshold is set below the maximum available resources to enable that real-time and best-effort traffic are able to share the channel efficiently. Finally, the destination node receives the packet and returns a probing response packet with a copy of the bottleneck bandwidth found along the path back to the source. When the source receives the probing response it compares the end-to-end bandwidth availability and the bandwidth requirement and decides whether to start a real-time flow accordingly. If the flow is admitted, the real-time packets are marked as RT (Real-Time packets) and they bypass the shaper mechanism at the intermediate nodes and are thus not regulated.

Since the traffic load conditions and network topology change dynamically, real-time sessions might not be able to maintain the bandwidth and delay bound requirements and they will have to be rejected or readmitted. For this reason, it is said that SWAN offers soft QoS. SWAN incorporates the Explicit Congestion Notification mechanism (ECN), which regulates real-time sessions as follows. When a mobile node detects congestion or overload conditions, it starts marking the ECN bits in the IP header of the real-time packets. The destination monitors the packets with the marked ECN bits and informs the source sending a 'regulate' message. Then the source node tries to re-establish the real-time session with its bandwidth needs accordingly.

In SWAN, intermediate nodes do not keep any per-flow information and thus avoid complex signaling and state control mechanisms. This makes the system relatively simple and scalable.

3 DS-SWAN (Differentiated Services-SWAN)

To support end-to-end QoS it is not only necessary to provide service differentiation inside the ad hoc network: the fixed IP network must use a QoS architecture, such as DiffServ, to provide scalable service differentiation in the Internet. In the DiffServ architecture [7], each priority class is associated to a different PHB (Per-Hop Behavior). The PHB defines how packets are forwarded by the routers. Each packet carries a particular marking ('codepoint') that is unique for each PHB. Edge routers perform the marking of the incoming packets and the core routers only need to examine the packets' codepoints and forward them acccording to the associated PHBs. One DiffServ service class corresponds to the EF (Expedited Forwarding) PHB, that provides low loss, low latency, low jitter and end-to-end assured bandwidth service. It provides a Premium Service. In our study the EF aggregates correspond to real-time traffic and are policed with a token bucket meter. Some bursts are tolerated but the traffic that exceeds the profile is marked with a different codepoint and then it is dropped. The number of dropped packets at the edge router and the end-to-end delay of the real-time connections are associated with the QoS parameters of the SWAN model in the ad hoc network. We observe that if the rate of the best-effort leaky bucket traffic shaper is lower then best-effort traffic is more efficiently rate controlled and real-time traffic is not so much influenced by best-effort traffic and it is able to maintain the required QoS parameters. For this reason, it is necessary that the SWAN model cooperates with the DiffServ model in the ad hoc network.

We propose a new protocol that enables the co-operation between the described DiffServ architecture at the fixed network and the explained SWAN scheme in the ad hoc network to improve end-to-end QoS support. We consider a scenario where best-effort CBR background traffic and real-time VBR traffic are transmitted as the mobile nodes in the ad hoc network communicate with one the fixed hosts located in the Internet through the gateway. In the proposed model, DS-SWAN, the edge router that is close to the gateway periodically monitors the number of packets of EF (real-time) traffic that are dropped because they are out of the established profile for this kind of traffic. Besides, the destination nodes in the wired IP network periodically monitor the average end-to-end delays of the real-time flows that have been established. It is thus required that the real-time application provides time-stamps in the data packets. Specifically, we use an interesting real-time VBR application: VBR Voice-over-IP (VoIP). In this context, if the end-to-end delay of one or more VBR VoIP flows is larger than 140 ms, then the destination nodes send a QoS_LOST packet to the edge router near the gateway to warn it. We have chosen this value because the ITU-T (International Telecommunication Union) recommends in its standard G.114 that the end-to-end delay should be kept below 150 ms to maintain an acceptable conversation quality in VoIP [11].

For PCM encoding with the G. 711 codec, the VoIP packet loss should never drop over a percentage of 5% of all generated frames to prevent significant losses in quality [6]. We have observed from initial simulation runs that the number of dropped

VoIP packets in the ad hoc network is always kept under 1%. Therefore, we establish that if the number of dropped VoIP packets at the edge router is less than 4 % and the edge router has received a notification that the end-to-end delays for VoIP flows are excessive, then the edge router must send a QoS_LOST message to the nodes in the ad hoc network to inform them that the system is too congested to maintain the desired QoS. In this way, we can change the parameter values of SWAN dynamically according to the traffic conditions not only in the ad hoc network but also in the fixed IP network.

The nodes in the ad hoc network use a queue to store packets at the MAC layer waiting for medium access. This queue uses priority scheduling to prioritize routing packets. QoS_LOST packets are treated as routing packets because they are warnings and must arrive to their destinations as soon as possible.

When the mobile nodes in the ad hoc network are warned, they will react by modifying the parameter values in the SWAN's AIMD rate control algorithm mentioned above. In DS-SWAN, every time that a QoS_LOST message is received, the node decreases the value of c by $\Delta c-$ with a certain minimum value. When no QoS_LOST message is received during T seconds, the node increases the value of c by $\Delta c+$ bits/s unless the initial value has been reached. This is done to prevent starvation of best-effort traffic.

When a node receives a QoS_LOST message, it increases the value of r by $\Delta r+$ up to a maximum value. When no QoS _LOST message has been received in the period T, the value of r is decreased by $\Delta r-$ up to the initial value.

SWAN has a minimum rate m for the best-effort leaky bucket traffic shaper. In DS-SWAN nodes are also allowed to reduce m. When a node receives a QoS_LOST message, it reduces the minimum rate by $\Delta m-$ Kbit/s. However, this parameter value is kept above a minimum value of m_0 Kbit/s and is increased $\Delta m+$ bits/s every second up to the initial value when the mobile nodes do not receive a warning message in T seconds. Table 1 shows the specific parameter values that we have selected for the simulations. However, operators and users are free to set these values according to their own needs, based on the characteristics of the targeted network.

Table 1. Parameter values in our simulations

Paremeters	Initial value of c	$\Delta c-$	$\Delta c+$	Minimum value of c	Initial value of r	$\Delta r+$	$\Delta r-$	Maximum value of r	Initial minimum rate	$\Delta m-$	$\Delta m+$	m_0
Values in our simulations	41 Kbit/s	10 Kbit/s	50 bits/s	11 Kbit/s	50 %	10%	1%	90%	31 Kbit/s	10 Kbit/s	50 bits/s	11 Kbit/s

4 Simulations

The aim of DS-SWAN is that real-time traffic can satisfy its bandwidth and delay requirements and best-effort traffic can use the remaining bandwidth effectively. The end-to-end delays of individual real-time flows will be reduced if there is congestion due to excess of best-effort traffic. However, it is important to remark that, on the

contrary, if end-to-end delays become excessive because of reasons such as failures of physical links, then the question remains whether our algorithm will be able to maintain end-to-end QoS requirements for these real-time flows. Therefore, we have run simulations with the NS-2 [12] tool in order to investigate the performance of DS-SWAN with a relatively realistic system model that incorporates effects of all relevant communication layers.

The system framework is shown in Fig. 2. We consider a single DiffServ domain (DS-domain) covering the whole network between the wired corresponding hosts and the gateway. The chosen scenario consists of 20 mobile nodes, 1 gateway, 3 fixed routers and 3 fixed hosts. The mobile nodes are distributed in a square region of 500 m by 500 m

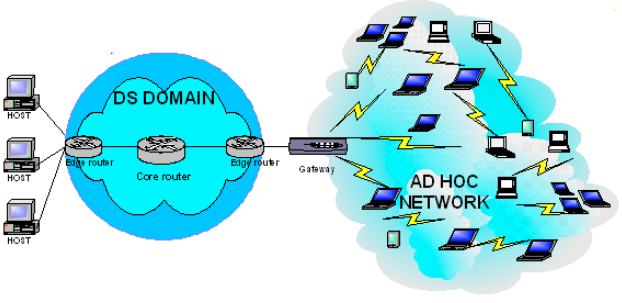

Fig. 2. Simulation framework

We assume that two traffic classes are transmitted: best-effort CBR background traffic and real-time VBR VoIP traffic. The mobile nodes communicate with one of the three fixed hosts located in the Internet through the gateway. Thus, the destination of all the CBR and VBR VoIP traffic is one of the three hosts in the wired network and some nodes in the ad hoc network will act as intermediate nodes or routers forwarding the packets from other nodes. In order to represent best-effort background traffic, 13 of the 20 mobile nodes are selected to act as CBR sources and fifteen nodes are selected to send VBR VoIP traffic.

The CBR best-effort packets that need to be sent are first processed by a leaky bucket traffic shaper so that they are delayed accordingly to a rate determined by the shaper. Afterwards, they are put in a queue at the MAC layer and should wait for medium access. The VBR VoIP packets are put in the same queue as well. This queue uses priority scheduling to prioritize routing packets and QoS_LOST packets. The rest of the traffic (VBR VoIP and CBR packets) are served without priorities so that always the oldest request is handled first.

The dynamic routing algorithm is AODV [13] and the mobile hosts use IEEE 802.11b. Each node selects a random destination within the area and moves toward it at a velocity uniformly distributed between 0 and 3 m/s. Upon reaching the destination the node pauses a fixed time period of 20 seconds, selects another destination and repeats the process.

To avoid synchronization problems due to deterministic start time, background traffic is generated with CBR traffic sources whose starting times are drawn from a uniform random distribution in the range [15 s, 20 s] for the first source, [20 s, 25 s] for the second one and so on. They have a rate of 48 Kbit/s with a packet size of 120

bytes. The VBR mode is used for VoIP traffic. We employ a silence suppression technique in voice codecs so that no packets are generated in silence period. For the voice calls, we use the ITU G.711 a-Law codec [14]. The VoIP traffic is modelled as a source with exponentially distributed on and off periods with 1.004 s and 1.587 s average each. Packets are generated at a constant inter-arrival time during the on period. Fifteen VoIP connections are activated at a starting time chosen from a uniform distribution in the range [10 s, 15 s]. Packets have a constant size of 128 bytes.

Shaping of EF (VoIP) and BE (Best-Effort) (CBR) traffic is done in two different drop tail queues of size 30 and 100 packets respectively. The EF and BE aggregates are policed with a token bucket meter with CBS = 1000 bytes and CIR = 200 Kbit/s. CBS (Committed Burst Size) refers to the maximum size of the token bucket and it is measured in bytes. CIR (Committed Information Rate) refers to the rate at which tokens are generated and it is specified in Kbit/s. We have run 40 simulations to assess the end-to-end delay and packet loss of VoIP traffic and the throughput of background traffic.

In the first simulations, we have implemented DS-SWAN in a way that the edge router sends a QoS_LOST message only to the VoIP sources generating flows that have problems to keep their end-to-end delays under 150 ms and to the intermediate nodes along the routes ("DS-SWAN- VoIP sources" label in the figures). We have evaluated and compared the performance of this implementation of DS-SWAN with the existing SWAN scheme.

Fig. 3 shows the average end-to-end delay for VoIP traffic in both cases. We observe that using SWAN the end-to-end delays increase progressively because the system is congested due to the large number of VoIP flows and background VoIP traffic. From the second 115 until the end of the simulation the end-to-end delays are too high for an acceptable conversation quality [11]. In DS-SWAN there exist flows that suffer end-to-end delays larger than 140 ms; hence, the destination nodes warn the edge router, which checks the percentage of lost packets and after verifying that it is less than 4%, it warns the nodes in the wireless ad hoc network to react accordingly. Then the nodes in the ad hoc network increase or decrease the pertinent parameters following the already explained DS-SWAN implementation and thus the system prevents the end-to-end delay to become larger than 150 ms.

Fig. 4 shows the average throughput for background traffic. In DS-SWAN, the average throughput for this kind of traffic is lower than in SWAN because the nodes in the ad hoc network react by decreasing the rate of the best-effort traffic shaper when they receive a warning. We must recall that a node may be part of a real-time and a background route at the same time. In any case, DS-SWAN functions correctly because there is no starvation of background traffic.

All simulations indicate that the packet loss rate for VoIP is well below the required 5%.

Now we have evaluated and compared the performance of the DS-SWAN protocol in the already explained scenario using two different implementations:

- The already explained implementation, where warnings are sent only to the VoIP sources having problems to keep their end-to-end delays under 150 ms and to the intermediate nodes along their respective routes ("Case 1, DS-SWAN - VoIP sources").

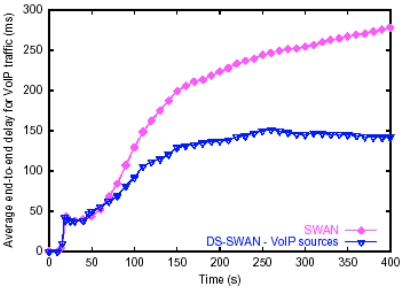

Fig. 3. Average end-to-end delay for VoIP traffic: DS-SWAN vs. SWAN

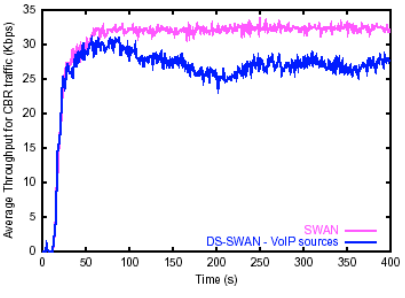

Fig. 4. Average throughput for CBR traffic: DS-SWAN vs. SWAN

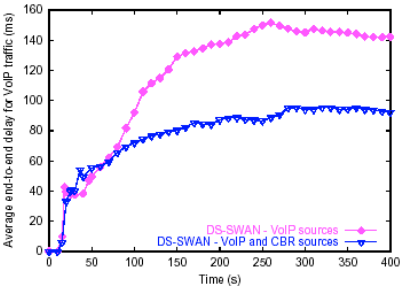

Fig. 5. Average end-to-end delay for VoIP traffic in the two DS-SWAN implementations (Case 1 and Case 2)

- Where warnings are sent to the VoIP sources having problems to keep their end-to-end delays under 150 ms, to all the CBR sources and to the intermediate nodes along the routes ("Case 2, DS-SWAN - CBR and VoIP sources").

Fig. 5 shows the average end-to-end delay for VoIP traffic in the two cases. Average end-to-end delays are kept well below 150 ms in both cases, but in Case 2 it is significantly smaller. This is because two neighbouring nodes that belong to two different routes, each carrying a different type of traffic, may still compete to access the medium. In Case 2, nodes carrying best-effort traffic that are in the proximity of a VoIP route and may contend with it for the medium access, are forced to reduce their data rates.

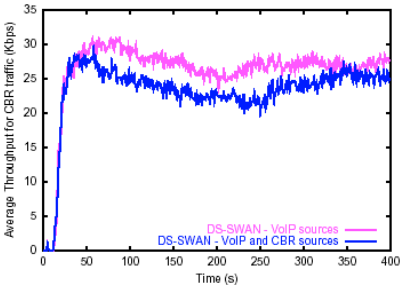

Fig. 6. Average throughput for CBR traffic (Case1 versus Case 2)

Fig. 6 shows the average throughput for background traffic obtained with the two DS-SWAN implementations. In Case 1, the average throughput is larger than in Case 2 because signalling is lighter. In any case, there is not starvation of background traffic in Case 2.

5 Conclusions

This work presents simulations of DS-SWAN in a relatively small mobile ad hoc network connected to a DiffServ domain. We have analyzed the functioning of the systems when multiple CBR background traffic flows and VBR VoIP flows have been established from mobile nodes to correspondent hosts at the fixed network. The parameter values of the traffic shaper that control the delay undergone by best-effort traffic are changed dynamically accordingly to the traffic conditions in the whole route. Simulation results demonstrate that DS-SWAN clearly outperforms SWAN in this scenario and best-effort traffic does not undergo starvation and can use the remaining bandwidth effectively.

Since sending warnings to all nodes in the ad hoc network may not be scalable, we have studied the performance of two implementations where only a selection of the mobile nodes receive signalling messages from the edge router. The two implementations show similar performance, and the choice of one over the other depends on the trade-off between end-to-end delay of real-time flows and throughput of background traffic. It still remains to be seen what the performance will be for larger networks.

Acknowledgements. This work was partially supported by the "Ministerio de Ciencia y Tecnología" of Spain under the project TIC2003-08129-C02, which is partially funded by FEDER, and under the programme Ramón y Cajal.

References

[1] D. Remondo and I. G. Niemegeers, “Ad hoc networking in future wireless communications”, *Computer Communications*, vol 26, no. 1, Jan. 2003, pp. 36-40.

[2] M. Benveniste, G. Chesson, M. Hoeben, A. Singla, H. Teunissen, and M. Wentink, “EDCF proposed draft text*”, IEEE working document 802.11-01/131r1,* March 2001.

[3] A. Iwata, C-C. Chiang, G. Yu, M. Gerla, and T-W. Chen, “Scalable routing strategies for ad hoc wireless networks”, *IEEE Journal on Selected Areas in Communications,* 17(8):1369-1379, August 1999.

[4] S. B. Lee and A. Campbell, “INSIGNIA”*, Internet Draft,* May 1999.

[5] H. Xiao, K.G. Seah, A. Lo and K.C. Chua, "A flexible quality of service model for mobile ad hoc networks", *IEEE Vehicular Technology Conference (VTC Spring 2000)*, Tokyo, Japan, May 2000, pp. 445-449.

[6] P.B. Velloso, M. G. Rubinstein and M. B. Duarte, “Analyzing Voice Transmission Capacity on Ad Hoc Networks”, *International Conference on Communications Technology - ICCT 2003*, Beijing, China, April 2003.

[7] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, “An architecture for differentiated service”, *Request for Comments (Informational) 2475, Internet Engineering Task Force,* December 1998.

[8] H. Arora and H. Sethu, “A Simulation Study of the Impact of Mobility on Performance in Mobile Ad Hoc Networks,” *Applied Telecommunications Symposium*, San Diego, Apr. 2002.

[9] H. Arora, Ll. Greenwald, U. Rao and J. Novatnack, “Performance comparison and analysis of two QoS schemes: SWAN and DiffServ”, *Drexel Research Day Honorable Mention,* April 2003.

[10] G.-S. Ahn, A. T. Campbell, A. Veres and L.-H. Sun, “SWAN”, *draft-ahn-swan-manet-00.txt*, February 2003.

[11] ITU-T Recommendation G.114, “One way transmission time", May 2000.

[12] NS-2: Network Simulator, *http://www.isi.edu/nsnam/ns*.

[13] C.E. Perkins, E.M. Royer, “Ad-hoc On-demand Distance Vector routing,” in *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications,* New Orleans, U.S.A., Feb. 1999.

[14] D. Chen, S. Garg, M. Kappes and K.S. Trivedi, “Supporting VBR Traffic in IEEE 802.11 WLAN in PCF Mode,” in *Proc. OPNETWORK’02*, Washington D.C., Aug. 2002.