

Software TDMA for VoIP Applications over IEEE802.11 Wireless LAN

Fanglu Guo Tzi-cker Chiueh
Computer Science Department
Stony Brook University, NY 11794
{fanglu, chiueh}@cs.sunysb.edu

Abstract—An emerging killer application for enterprise wireless LANs (WLANs) is voice over IP (VoIP) telephony, which promises to greatly improve the reachability and mobility of enterprise telephony service at low cost. Most commercial IEEE802.11 WLAN-based VoIP products cannot support more than ten voice conversations over a single IEEE 802.11b channel, even though its peak transmission rate is more than two orders of magnitude higher than an individual VoIP connection's bandwidth requirement. There are two main reasons why these VoIP systems' effective capacity is lower than expected: stringent latency requirement and substantial per-WLAN-packet overhead. Time-Division Multiple Access (TDMA) is a well-known technique that provides per-connection QoS guarantee as well as maximizes the radio channel utilization efficiency. This paper presents a software-based TDMA (STDMA) protocol that is designed to support VoIP applications and successfully works on commodity IEEE802.11 WLAN interfaces. The resulting STDMA prototype can support more than 50 two-way G.729 voice conversations over a single IEEE 802.11b channel.

I. INTRODUCTION

Voice over Wireless LAN (VoWLAN) is touted as a killer application for enterprise WLANs because it significantly improves the coverage and mobility of enterprise telephony services. In fact, vendors such as Aruba Networks, Symbol, SpectraLink and Cisco have been shipping VoIP phones specifically designed for IEEE 802.11-based WLANs. However, there are still several technical barriers facing the VoWLAN technology that need to be overcome before it can truly take off.

The first barrier is lack of Quality of Service (QoS) support. ITU-T G.114 [1] recommends the the maximum one-way voice packet delay be below 150 msec. Because this corresponds to end-to-end path delay, the delay due to wireless LAN channel access must be considerably less than 150 msec. In addition Most voice codec specifications such as G.729 [2] require the packet loss ratio in voice connections to be less than 1 % to avoid audible errors. The original IEEE802.11 WLAN standard supports only best-effort service model. Almost all existing WLAN deployments operate in the DCF (Distributed Coordination Function) mode, in which each wireless station accesses the shared radio channel using an Ethernet-like medium access mechanism, which is inadequate for real-time voice applications for the following reasons. First, it is impossible to have precise control over the exact transmission timings of voice frames because of collision and random back-off. When the injected traffic load is high,

there is more collision and interference, which increase packet loss rate, packet delay and packet delay jitter. Second, it is impossible to prioritize traffic flows. Consequently, large-volume non-real-time data traffic such as FTP may consume a large proportion of the network capacity, leaving time-sensitive voice traffic to suffer the consequences.

The second barrier associated with IEEE802.11 WLAN is substantial per-packet transmission overhead. Because the maximum transmission rate of an IEEE802.11b WLAN link is 11 Mbps, in theory an IEEE802.11b link should be able to support hundreds of VoIP connections if each of them requires 8Kbps such as G.729. In practice, several benchmarking tests [3], [4] reported that existing VoWLAN products on the market cannot support more than ten concurrent VoIP calls with comparable quality to toll calls over a single IEEE802.11b channel. A major cause for this gap between theory and practice is the considerable overhead associated with a WLAN packet's transmission. This overhead includes per-packet header bits, link-layer acknowledgment, back-off delay to avoid contention, retransmission cost due to interference and inter-frame spacing for synchronization.

IEEE 802.11e [5] is designed to provide QoS support for time-sensitive applications on IEEE802.11 WLAN. It supports two new channel access mechanisms: EDCA (Enhanced Distributed Channel Access) and HCCA (Hybrid coordination function (HCF) Controlled Channel Access). EDCA improves upon DCF (Distributed Coordination Function) by introducing four queues, each of which corresponds to a different priority of accessing the shared radio channel. More specifically, each queue is assigned a different combination of medium access parameters, including AIFS (Arbitrary Inter-Frame Space), CWmin and CWmax. Although EDCA can effectively prioritize voice traffic over data traffic, it cannot guarantee the QoS of individual VoIP connections when they are competing with one another for a shared medium. In fact, when the number of voice connections contending for access to a radio channel increases, EDCA actually increases the probability of collision because of its aggressive medium access parameters such as smaller CWmin and CWmax. HCCA is an enhanced version of PCF (Point Coordination Function), and supports a centralized polling scheme that could schedule network connections according to their bandwidth demand and priority. Unfortunately, most commodity WLAN interface products do not implement HCCA or PCF. In summary, IEEE 802.11e takes the right first

step toward solving the IEEE802.11 WLAN's QoS problem by supporting traffic prioritization. But it is ineffective in the face of a large number of voice connections. Neither does it solve the problem of substantial per-packet transmission overhead.

This project aims to simultaneously solve the QoS and per-packet transmission overhead problem associated with an infrastructure-mode IEEE802.11-based wireless LAN by layering a software-based Time Division Multiple Access (TDMA) protocol on top of DCF, which is the default operating mode of most IEEE802.11 WLAN interfaces. Measurements from a fully operational software-based TDMA (STDMA) prototype show that this approach drastically improves an IEEE802.11 WLAN channel's capacity for VoIP traffic because it reduces the collision probability to the minimum and because it largely eliminates the back-off overhead. In addition, this STDMA approach provides QoS guarantee for individual VoIP connections regarding packet loss, packet delay and delay jitter.

II. RELATED WORK

The Wireless Rether project [6] tried to provide hard QoS guarantee on 802.11-based WLANs through a software-based token passing scheme. Unfortunately software-based token passing incurs too much overhead to be practical. The IEEE 802.11e standard [5] enhances WLAN's QoS by supporting traffic prioritization. Its EDCA mechanism can give better QoS to voice traffic when compared with data traffic. However, this mechanism is not at all effective when all traffic sources carry voice traffic. Gu and Zhang [7] simulate the EDCA mechanism to evaluate its effectiveness in supporting voice connections' QoS. They used 4 stations each transmitting at a different priority, and found that EDCA indeed can provide higher throughput and lower latency to high-priority traffic. But they didn't test EDCA under the scenario in which all 4 stations transmit at the highest priority. In addition to EDCA, Garg et al. [8] also evaluated the HCCA, and found that HCCA can improve channel utilization and provide better QoS support. They also found that EDCA may require significant tuning to offer better QoS for high priority traffic. SpectraLink [9], a leading VoWLAN phone vendor, proposes a simple QoS enhancement mechanism called SpectraLink Voice Priority (SVP), which features two basic ideas: AP transmitting voice frames with backoff value of zero and always queuing voice frames in the head of the transmission queue. This solution is used to improve the AP's priority of acquiring the WLAN channel for downstream voice traffic.

Unlike PCF and HCCA, which relies on explicit polling to support contention-free access to the shared radio medium, TDMA uses implicit time slotting to achieve the same effect without incurring a per-STA polling frame overhead. In exchange, STDMA needs a per-scheduling-cycle announcement frame to avoid scheduling the radio channel for STAs that are idle.

Given the limited capacity of existing VoWLAN systems, it is not surprising that many researchers try to improve its capacity. Wang et al. [10] proposes to aggregate downstream

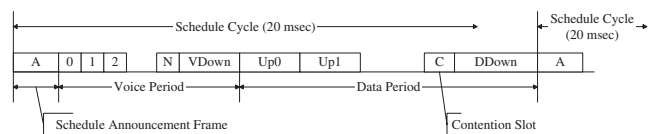


Fig. 1. The channel time allocation of the proposed software-based TDMA protocol. Time is divided into cycles, each of which consists of a *schedule announcement slot*, which notifies all STAs of their channel time slots during a schedule cycle, a *voice period*, which is for voice frames, and a *data period*, which is for non-voice data frames.

voice traffic into a smaller number of larger frames that are then multicasted. This scheme can nearly double the capacity of VoWLAN systems because it dramatically decreases the transmission overhead of downstream voice frames. The Atheros chipsets [11] support several new features to improve the capacity of WLAN links. The *bursting* feature allows a WLAN card to transmit frames back to back without incurring the per-frame backoff overhead. The *fast frame* feature allows a WLAN card to aggregate two frames into one larger frame. The *compression* feature allows a WLAN card to compress data frames. But these techniques cannot be applicable to VoIP traffic because typically consecutive frames within a voice connection are spaced out by 20 to 30 msec. It is difficult to aggregate them to larger frames or to exploit bursting. Compression is not that useful either because voice data is already compressed.

Neufeld et al. [12], [13] in their SoftMAC project mentioned some techniques to override the 802.11 MAC, for example, controlling ACK and backoff through software. They also mentioned an TDMA MAC project on 802.11-based WLAN NIC without providing concrete design and implementation details. Furthermore, SoftMAC requires each WLAN card to work in the RF monitor mode, and therefore could not be applied to VoWLAN phones. Rao and Stoica [14] proposed an overlay MAC layer on 802.11-based WLAN NIC to mitigate the poor fairness and performance problem due to hidden nodes and to mitigate other 802.11 MAC anomalies. It also uses TDMA to regulate frame transmission but its design is not geared towards VoIP applications. Its scheduling slot size is 10 msec and detects inactive nodes through long timeout. This coarse timing granularity control method cannot be reused in VoWLAN traffic.

III. STDMA PROTOCOL DESIGN

As shown in Figure 1, the proposed STDMA protocol divides the channel time into schedule cycles, each of which is set to 20 msec in the current prototype and corresponds to the packetization interval of VoIP applications¹. Each schedule cycle in turn is broken down into three parts: a *schedule announcement* frame, a *voice period*, and a *data period*. The schedule announcement frame specifies how the channel time is allocated among WLAN nodes in the current schedule cycle. The voice period consists of multiple time slots, one for each

¹If a VoIP application's packetization interval is different from 20 msec, its voice samples should be grouped into 20-msec chunks before being sent out.

active voice connection's *upstream* traffic. The AP transmits all *downstream* voice traffic in a single time slot, which is at the end of the voice period. The data period is also divided into time slots, each of which is used for transmission of non-voice data frames. Again upstream data traffic is scheduled first, then a contention slot is scheduled for non-active STAs to send their *traffic request* frames (explained later), and finally comes the downstream data time slot allocated to the AP. For example, in Figure 1, the first frame in the schedule cycle is the schedule announcement frame. Then the voice period starts. Each voice traffic-carrying STA uses one of the N voice time slots to transmit upstream voice frames. At the end of the voice period, the AP uses the *VDown* time slot to transmit downstream voice frames. In the data period, STAs transmit upstream non-voice data frames in time slots *Up0*, *Up1*, etc. After the upstream time slots, STAs that are not scheduled in the current cycle can send traffic request frames in the contention slot, *C*. Finally the AP uses the downstream data time slot, *DDown*, to transmit non-voice data frames to STAs.

A. The Schedule Announcement Frame

The AP broadcasts a schedule announcement frame at the beginning of each schedule cycle. This frame serves the double purposes of synchronizing STAs and telling each STA its associated channel slot times for upstream voice and data transmission. Because each STA is guaranteed its own slots, upstream data transmission is largely free of collision. Consequently, the packet loss, delay and jitter can be strictly controlled.

B. The Voice Period

In each voice period, every active STA transmits its upstream voice traffic in its own slot according to the schedule announcement frame. The AP aggregates and transmits all downstream voice frames in one voice slot. Ideally, if silence suppression works well, a voice connection only needs one channel slot for both upstream and downstream traffic because voice communication is half-duplex in nature. Unfortunately, several studies [15], [16] showed that state-of-the-art silence suppression cannot effectively suppress most of the background noise. As a result, VoIP traffic is largely full-duplex in practice and therefore downstream and upstream traffic of a voice connection cannot be time-multiplexed on a single channel time slot. One way to transport a voice connection's downstream traffic is to allocate a separate downstream time slot immediately after each STA's upstream time slot. This design saves power because each voice traffic-carrying STA in principle only needs to wake up during its associated time slots in each schedule cycle. However, it is inefficient. A more efficient design, which the current STDMA prototype chooses, is to aggregate and transmit all downstream voice traffic to all STAs in one time slot. This design allows an AP to exploit statistical multiplexing among multiple voice connections, to aggregate multiple voice frames into one physical frame to reduce per-packet transmission overhead [10], and to use

802.11e's TXOP mechanism [5] to transmit a batch of frames consecutively at the physical layer.

C. The Data Period

The data period is used for STAs and the AP to transmit non-voice data frames. Unlike voice traffic, which has constant per-connection bandwidth requirement, the bandwidth demand of data traffic fluctuates considerably. The key challenge in scheduling the data period is how to maximize its channel utilization efficiency without disrupting the timing of voice frames. Without proper control, too many STAs may try to transmit frames during a data period, stretch the data period into the next schedule cycle and disrupt the QoS guarantee of subsequent voice frames. To avoid this disruption, STDMA first schedules upstream data traffic from STAs, then a contention slot *C* and finally downstream data traffic from the AP. STAs that did not transmit data in previous schedule cycles submit a traffic request frame to the AP during the contention slot. Because traffic request frames are small, even though they could cause collision, they are not likely to stretch the contention slot significantly. Even if a contention slot is stretched, the following downstream data slot in the same data period can be shortened to accommodate any such stretch.

In the initial state, as soon as an STA has data to transmit, it submits a traffic request frame in the next available contention slot. The traffic request frame includes traffic load information, i.e., the amount of time required to empty the data currently in its queue. From an STA's traffic request frame and its physical transmission speed, the AP could determine the amount of channel time allocated to this STA and schedules a data slot for it in the following data period. After an STA starts transmitting upstream packets, whenever it transmits an upstream packet, it piggybacks into the packet a report of its current traffic load. Based on load information in traffic requests and piggybacked reports, the AP could accurately keep track of each STA's load, and schedules their packets in a way similar to weighted fair queuing [17]. After an STA reports to the AP that its transmission queue is empty, the AP will stop scheduling for the STA until receiving a traffic request frame from the STA in some future contention slot.

Compared with polling, which is used in PCF and requires the AP to constantly poll all STAs to determine their load and transmission schedule, TDMA is much more efficient especially for software implementation, because the overhead associated with software-based polling on WLAN is quite high [6]. In addition, because data traffic load tends to be sparse and bursty, constantly polling STAs incurs a great deal of scanning overhead without producing any obvious benefits.

IV. PERFORMANCE EVALUATION

We used the testbed shown in Figure 2 to evaluate the performance of the STDMA prototype and compare it with the IEEE802.11e standard. There are four computers connected through a 100Mbps Ethernet switch. All of them are DELL PowerEdge 400SC, which features a 2.26 GHz Pentium-4 CPU and 256 Mbytes of memory. One computer serves as an

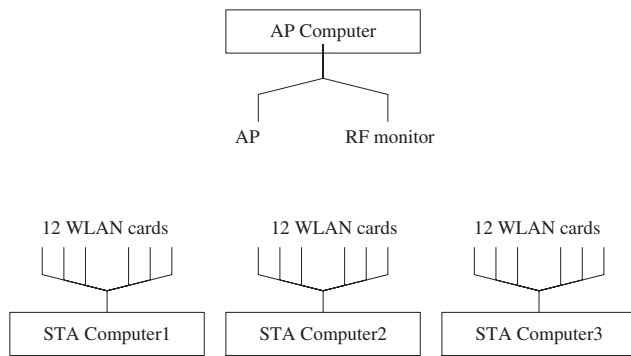


Fig. 2. The testbed used to evaluate the performance of the STDMA prototype and the IEEE802.11e standard. There are totally 36 WLAN cards associated with one AP, each of which logically corresponds to a distinct STA. The 36 WLAN cards transmit voice frames to the AP and the QoS of each voice connection in terms of packet delay and delay jitter is measured and reported.

AP and the other three serve as STAs. Each STA computer hosts 12 Wistron NeWeb IEEE802.11a/b/g mini-PCI cards (Model No. CM9), which are connected to the computer through three 4-port mini-PCI to PCI adaptors. The NeWeb IEEE802.11a/b/g mini-PCI card uses the AR5004X chipset, which consists of an AR5213 MAC controller chip supporting IEEE802.11e's EDCA, and an AR5112 dual-band radio. The AP computer hosts two Wistron NeWeb IEEE802.11a/b/g mini-PCI cards, one of them working as an AP while the other working as an RF monitor to verify that the STDMA prototype indeed works as expected. All WLAN cards operate in the IEEE802.11b mode with short preamble support enabled. ACK frames are encoded in 11Mbps. These computers are located within a 5-meter range and thus always transmit frames at the encoding rate of 11Mbps.

All computers run Linux 2.6.10, which includes the MadWifi driver `madwifi-ng-r1457` [18] and the High Res POSIX timer `i386-hrt-2.6.10.patch` [19], and use NTP to synchronize their clocks over the wired Ethernet link so that we can measure the one-way packet latency. We use a UDP sender program to generate traffic at a specific packet rate (packet per second), per-packet payload size, sequence number, and time stamp. To emulate VoIP traffic, we set the UDP payload size to 32 bytes (12 bytes RTP + 20 bytes G.729 voice). An UDP receiver program measures the throughput and packet loss ratio based on the sequence number. The packet delay and delay jitter are computed based on the time stamps taken at the socket send/receive interface to reflect the end-to-end delay perceived by the VoIP application. Packet delay jitter is the difference between maximum and minimum packet delay measurements.

The goal of this experiment is to compare the effective throughputs of STDMA, IEEE802.11e and IEEE802.11 in terms of the number of two-way 8Kbps voice connections they can support over a single channel of an IEEE802.11b WLAN. In this experiment, each WLAN card on the STA computers transmits to the AP packets of the total size of 98 bytes and at a rate of 50 packets per second (pps). At the

MAC	Cards	Direction	Loss Ratio (%)	Latency (msec)	Jitter (msec)
802.11	18	Up	0.23	23	30
802.11	18	Down	0.8	14	28
802.11	19	Up	0.26	18	44
802.11	19	Down	6	40	36
802.11e	22	Up	0.97	26	54
802.11e	22	Down	0.01	1	6
802.11e	23	Up	3.1	30	121
802.11e	23	Down	0.01	3	30
STDMA	50	Up	1.4	20	6
STDMA	50	Down	0.4	20	6

TABLE I
Comparison among IEEE802.11, IEEE802.11e and STDMA in terms of the number of voice calls that they can support over a single channel of an IEEE802.11b WLAN assuming each call is a two-way 8Kbps voice connection.

same time, the AP transmits packets to each WLAN card on the STA computers with the same packet size and rate. This set-up emulates a constant-rate two-way voice communication. To emulate IEEE802.11, we use the Atheros chipset's best-effort queue. To emulate 802.11e, we use the Atheros chipset's voice queue, which is the same as in the baseline test. For STDMA, we use the Atheros chipset's voice queue and use a 200- μ sec voice slot size. To emulate more than 36 STAs using 36 WLAN cards, some of them need to support two voice connections, i.e., transmit/receive every 10 msec rather than every 20 msec.

Table I summarizes the number of concurrent 8Kbps voice connections that IEEE802.11, IEEE802.11e and STDMA can support on a single IEEE802.11b channel whose transmission rate is 11 Mbps. Our measurement shows that the IEEE802.11 standard can actually support up to 18 simultaneous voice connections. When the 19th voice connection is added, the packet loss ratio of downstream voice traffic from the AP goes up drastically, but the upstream voice traffic experiences much fewer losses. In IEEE802.11, the AP uses the same AIFS and CWmin as the STAs, and therefore is not given a higher priority than STAs as far as accessing the shared channel is concerned. When the input traffic load of a WLAN channel is close to its capacity, the AP is given less channel time share than it needs to transmit downstream packets and thus becomes the bottleneck.

For IEEE802.11e, it can support 22 concurrent voice connections. When the 23rd voice connection is added, unlike IEEE802.11, the packet loss ratio of upstream traffic from STAs goes up while the downstream traffic experiences much fewer losses. In IEEE802.11e, the AIFS of the AP's voice queue is PIFS, which is smaller than DIFS, the AIFS of STAs' voice queue. In addition, the AP's voice queue uses a large TXOP limit to transmit multiple frames once it acquires the channel. This is why it is the STAs rather than the AP that become the bottleneck when a WLAN channel is fully loaded.

STDMA can support 50 voice calls successfully. Because

downstream voice traffic can effectively leverage the TXOP mechanism, the STDMA prototype can transmit downstream packets associated with 50 voice calls within 9.3 msec, leaving 0.7 msec for the schedule announcement frame and the contention slot in each schedule cycle, whose length is 20 msec. The throughput improvement of STDMA over IEEE802.11e mainly arises from two optimizations: TDMA-based medium access control and ACK elimination. To isolate out the throughput improvement contribution due to TDMA-based medium access control, we ran the same test with IEEE802.11e but disabled link-layer ACK this time. Under an offered load of 1 Kpps from 4 WLAN cards, IEEE802.11e with link-layer ACK disabled experiences a packet loss ratio of more than 25%! In contrast, the same 4 WLAN cards can achieve around 3 Kpps throughput with 0.4% loss ratio under IEEE802.11e without disabling link-layer ACK.

In summary, compared with IEEE802.11 and IEEE802.11e (22 voice calls), STDMA (50 voice calls) can improve the voice capacity by a factor of 2.8 and 2.3, respectively, while offering comparable packet delay and smaller delay jitter. The packet loss ratio of the upstream traffic under STDMA is slightly higher than those in IEEE802.11 or IEEE802.11e, mainly because of the mysterious collision behavior when link-layer ACK is disabled. For IEEE802.11 and IEEE802.11e, they also show decent capacity (18 and 22 voice calls, respectively). The result for IEEE802.11 is somewhat surprising because previous tests [3], [4] showed that state-of-the-art IEEE802.11-based VoWLAN products can only support around 10 concurrent voice calls over a single IEEE802.11b channel. The performance difference between STDMA and these commercial systems is mainly attributed to careful reduction in per-packet transmission overhead and downstream voice traffic batching at the AP. The performance improvement of IEEE802.11e over IEEE802.11 mainly comes from the fact that 802.11e gives the AP a higher priority over the STAs and thus significantly reduces the packet loss ratio of downstream voice traffic.

V. CONCLUSION

The number of VoIP calls that existing VoWLAN products can support on an IEEE802.11 WLAN channel is disappointingly small because of lack of QoS support and inefficient WLAN packet transmission. Real world experiments [3], [4] showed that they can support up to ten concurrent VoIP calls on a vanilla IEEE802.11b link. To overcome this VoIP capacity problem, this paper proposes a software-based TDMA approach to simultaneously solve the QoS and transmission efficiency problems of commodity IEEE802.11 WLAN interface hardware. Although STDMA is conceptually simple, implementing it on commodity operating system and WLAN hardware poses substantial challenges, specifically, how to synchronize WLAN nodes and how to support fine-resolution timer. We have successfully implemented the first known STDMA prototype on Linux and commodity WLAN NIC without modifying the WLAN device driver. The prototype could be easily ported to other VoWLAN appliances with-

out much modification. Performance measurements on the STDMA prototype show that STDMA can increase the number of concurrent VoIP calls on an IEEE802.11b WLAN channel to 50, assuming each VoIP call uses G.729 Codec (8 Kbps).

ACKNOWLEDGMENT

This research is supported by NSF Awards CNS-0435373, CNS-0410694 and CCF-0342556.

REFERENCES

- [1] V. Wiki, "VOIP Qos Requirements," <http://www.voip-info.org/wiki-QoS>, 2005.
- [2] I. T. Union, "G.729: Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)," 1999. [Online]. Available: <http://www.itu.int/rec/T-REC-G.729-199603-I/en>
- [3] D. Newman, "Review: Voice over Wireless LAN," *Network World*, 01/10/05, 2005. [Online]. Available: <http://www.networkworld.com/reviews/2005/011005rev.html>
- [4] —, "Aruba conquers challenge of Wi-Fi scalability," *Network World*, 11/06/06, 2006. [Online]. Available: <http://www.networkworld.com/reviews/2006/110606-wifi-test.html>
- [5] IEEE, "IEEE Standard 802.11e-2005," <http://standards.ieee.org/getieee802/802.11.html>, 2005.
- [6] S. Sharma, K. Gopalan, N. Zhu, G. Peng, P. De, and T. cker Chiueh, "Implementation Experiences of Bandwidth Guarantee on a Wireless LAN," in *Proceedings of ACM/SPIE Multimedia Computing and Networking (MMCN 2002)*, 2002.
- [7] D. Gu and J. Zhang, "Evaluation of EDCF Mechanism for QoS in IEEE 802.11 Wireless Networks," in *Proceedings of World Wireless Congress (WWC)*, 2003.
- [8] P. Garg, R. Doshi, R. Greene, M. Baker, M. Malek, and X. Cheng, "Using IEEE 802.11e MAC for QoS over Wireless," in *Proceedings of International Performance Computing and Communications Conference*, 2003.
- [9] SpectraLink Inc., "SpectraLink Voice Priority," http://www.spectralink.com/files/literature/SVP_white_paper.pdf, 2005.
- [10] W. Wang, S. C. Liew, Q. X. Pang, and V. O. Li, "A Multiplex-Multicast Scheme that Improves System Capacity of Voice-over-IP on Wireless LAN by 100%," in *Proceedings of the Ninth IEEE Symposium on Computers and Communications (ISCC)*, 2004.
- [11] Atheros Communications Inc., "Super G: Maximizing Wireless Performance," http://www.atheros.com/pt/whitepapers/atheros_superg_whitepaper.pdf, 2005.
- [12] M. Neufeld, J. Fifield, C. Doerr, A. Sheth, and D. Grunwald, "Soft-MAC - Flexible Wireless Research Platform," in *Proceedings of Fourth Workshop on Hot Topics in Networks (HotNets) '05*, 2005.
- [13] C. Doerr, M. Neufeld, J. Fifield, T. Weingart, D. C. Sicker, and D. Grunwald, "MultiMAC - An Adaptive MAC Framework for Dynamic Radio Networking," in *First IEEE Symposium on New Frontiers in Dynamic Spectrum Networks (DySPAN) '05*, 2005.
- [14] A. Rao and I. Stoica, "An Overlay MAC layer for 802.11 networks," in *Proceedings of ACM Mobisys Conference 2005*, 2005.
- [15] Cisco Systems Inc., "Voice Over IP - Per Call Bandwidth Consumption," http://www.cisco.com/warp/public/788/pkt-voice-general/bwidth_consume.html, 2006.
- [16] Yiyang Zhou and others, "Performance Analysis for VoIP System," http://www.csie.ntu.edu.tw/~acpang/course/voip_2005_fall/presentation/B2.ppt, 2005.
- [17] H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," *Proceedings of the IEEE*, vol. 83, no. 10, pp. 1374–1396, January 1995.
- [18] M. project, "MadWifi - Multiband Atheros Driver for Wireless Fidelity (WiFi)," <http://madwifi.org/>, 2006.
- [19] M. S. Inc., "High Resolution POSIX timers," <http://sourceforge.net/projects/high-res-timers/>, 2005.