Techniques to Improve Blueooth Performance in Interference Environments

Nada Golmie and Nicolas Chevrollier
National Institute of Standards and Technology
Gaithersburg, Maryland 20899
Email: `nada.golmie@nist.gov`

Abstract—*Bluetooth is a radio technology for Wireless Personal Area Networks operating in the 2.4 GHz ISM band. Since both Bluetooth and IEEE 802.11 devices use the same frequency band and may likely come together in a laptop or may be close together at a desktop, interference may lead to significant performance degradation. The main goal of this paper is to propose solutions to the interference problem consisting of power control adjustments and scheduling policies to be implemented by the Bluetooth device. Simulation results are given for selected scenarios and configurations of interest.*

Keywords—**Bluetooth, Interference, Power Control, MAC scheduling**

I. INTRODUCTION

The Bluetooth [1] technology is an emerging short range cable replacement protocol operating in the 2.4 GHz ISM band. Since both the Bluetooth and the IEEE 802.11 [2] protocols operate in the 2.4 GHz, it is anticipated that interference may severely degrade the performance of both systems.

Our goal is to propose solutions to the interference problem pertaining to the Bluetooth radio operating in proximity to an IEEE 802.11 network. We assume that the source of interference to the Bluetooth system is an IEEE 802.11 system operating in a direct sequence spread spectrum (DSSS) mode. In the rest of this sequel, the terms IEEE 802.11 DSSS and WLAN will be used interchangeably.

We investigate two techniques aimed at alleviating the interference problem for Bluetooth. One technique is based on controlling the transmitted power and keeping it proportional to the signal-to-interference ratio (SIR) measured at the receiver. The other technique takes advantage of the frequency hopping sequence of Bluetooth and uses scheduling with the aim of avoiding interference. Simulation results for scenarios of interest are discussed. Performance is measured in terms of the mean access delay, the probability of packet loss, and the transmitted power.

This paper is organized as follows. In sections II and III, we describe the distributed power control algorithm and the scheduling mechanism respectively and give numerical results. Concluding remarks are offered in section IV.

II. POWER CONTROL

Given that some devices provide the ability to dynamically modify their transmission power, we would like to investigate the dynamics of a power control (PC) strategy as a means of alleviating the impact of interference.

We use a distributed algorithm to implement a PC procedure. The basic idea is to adjust the interference level in the system to no more than what is needed. We assume that the receiver does not have any knowledge of other systems except for the system it is communicating with. Interference from other systems is measured in terms of the SIR level at the receiver. Note that SIR is a wide-spread link quality measure and has been used in many previous studies for power control and dynamic channel allocation for interference limited systems [3] [4] [5]. The power update algorithm works as follows. Initially, $P_0 = P_{max}$, then every update interval U, the power at the transmitter, $P(t+1)$ is updated as follows:

$$P(t+1) = \min(P_{max}, \max(P_{min}, \frac{\tau_t}{SIR(t)} \times P(t)) \qquad (1)$$

where $\tau(t)$ is the target SIR and SIR(t) is based on an average value over many measurements. The power update rule takes into consideration the SIR(t) statistic measured at the receiver side. The receiver can then relay this information to the transmitter every update interval U.

Implementation Considerations Although the exact details of a power control algorithm have been left undefined for the most part, the Bluetooth specifications have included the necessary hooks in the protocol in order to implement a power control algorithm. Furthermore, the Bluetooth specifications classifies devices into three power classes as summarized in Table I

TABLE I
BLUETOOTH DEVICE POWER CLASSES

Power Class	Maximum Output Power	Minimum Output Power
1	100 mW (20 dBm)	1 mW (0 dBm)
2	2.5 mW (4 dBm)	0.25 mW (-6 dBm)
3	1 mW (0 dBm)	N/A

Class 1 requires power control limiting the transmitted power over 0 dBm, while power control is optional for classes 2 and 3. The specifications suggest that the transmitted power should be adjusted based on the received signal strength indicator (RSSI) measurements at the receiver. Note that in an interference-limited environment, RSSI corresponds to the the SIR (assuming that noise can be neglected). Furthermore, the specifications define Link Management Protocol (LMP) messages for adjusting the power control as shown in Table II. The general format of a Link Manager Protocol (LMP) message is illustrated in Figure 1.

Both LMP messages, $LMP\_incr\_pow\_req$ and $LMP\_decr\_pow\_req$, include one byte of contents reserved for future use. We propose using this byte to transmit the measured

TABLE II

LMP POWER CONTROL MESSAGES

Message	Op_code	Contents
LMP_incr_pow_req	31	1 byte- future use
LMP_decr_pow_req	32	1 byte- future use
LMP_max_power	33	1 byte
LMP_min_power	34	1 byte

SIR at the receiver in order for the transmitter to implement the update rule given by Equation 1.

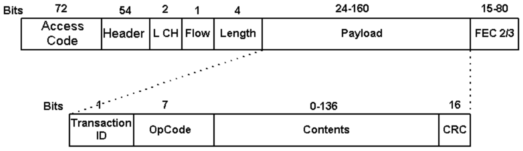


Fig. 1. LMP Message Format

Annother implementation issue to consider is the value of the update interval, U. Andersin *et al.* [6] demonstrate that for a system such as GSM, the SIR can be accurately estimated within 0.1 to 0.3 seconds. The values are for heavily interfered system with an interference level 20 dB above the noise floor. In our case, the value of SIR depends on the main signal and the interference spectral shape (i.e. whether the main signal falls inside or outside of the interfering signal band). Therefore, given 79 frequency channels, U can be chosen proportionally to 4 or 5 times 79. There is a trade-off between the value of U and the amount of signaling traffic required. A small value for U allows the system to be perhaps more responsive at the cost of having to exchange additional signaling information.

Numerical Results

We present simulation results to evaluate the effect of the power control algorithm. We use a 4-node topology as illustrated in Figure 2, and the simulation parameters presented in Table III. We vary the traffic distributions for WLAN and Bluetooth as follows.

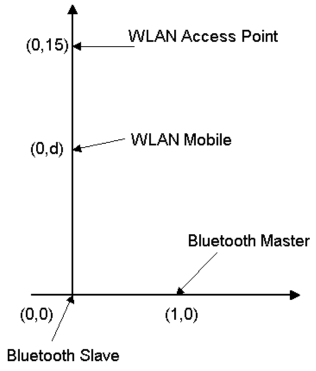


Fig. 2. Experiment Topology

We assume that the WLAN Mobile device is transmitting data packets to the AP device which is responding with ACKs. The WLAN packet payload is set to 7776 bits transmitted at 11 Mbits/s, while the packet header is set to 224 bits transmitted at 1 Mbits/s. We assume that the WLAN packet interarrival rate is exponentially distributed with a mean of 1.86 ms corresponding to 50% of the offered load. For Bluetooth, we assume that both master and slave devices are transmitting DM1 packets with a mean arrival rate of λ where $\lambda = \frac{2*0.000625}{l} - 2*0.000625$ seconds, and $l = 30$ is the offered load in percent of the channel capacity. Our setup parameters are summarized in Table III. We measure the probability of packet loss and the mean access delay measured at the Bluetooth slave device.

TABLE III

ADAPTIVE POWER SIMULATION PARAMETERS

Simulation Parameters	**Values**
Update Interval (U)	300 packets
Bluetooth Parameters	**Values**
ACL Baseband Packet Encapsulation	DM1, DM3, DM5
Packet Interarrival Time for DM1	2.91 ms
Packet Interarrival Time for DM3	8.75 ms
Packet Interarrival Time for DM5	15.58 ms
P_{min}	1 mW
P_{max}	100 mW
Slave Coordinates	(0,0)
Master Coordinates	(1,0)
WLAN Parameters	
Packet Interarrival Time	1.86 ms
Offered Load	50 % of Channel Capacity
Transmitted Power	25 mW
Data Rate	11 Mbits/s
AP Coordinates	(0,15)
Mobile Coordinates	(0,d)
Packet Header	224 bits
Payload Size	7776 bits

The power update rule given by Equation 1 was implemented at the Bluetooth master and slave devices. Initially, the power was set to $P_{max} = 100$ mW, then updated according to the rule. SIR was measured over an update interval, U, equal to 300 packets. Figure 3 shows the transmitted power (after 5 U) for the Bluetooth master device versus the distance of Bluetooth slave from the source of interference. Note that if there is no change in the interference signal, the transmitted power should converge to its final value in one step, i.e. 1 U.

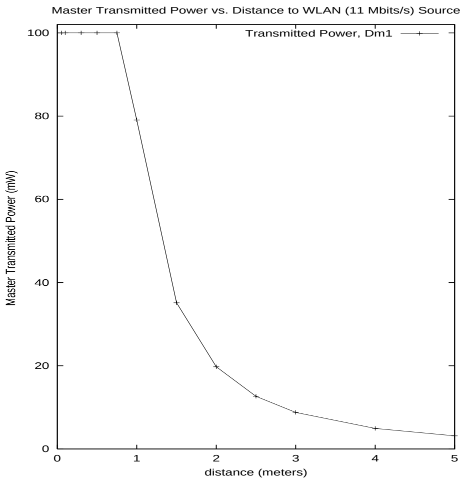


Fig. 3. Bluetooth Transmitted Power

As expected the transmitted power in Figure 3 varies between P_{max} and P_{min}. Figure 4 (a) and (b) give the packet loss and the access delay respectively with and without the power

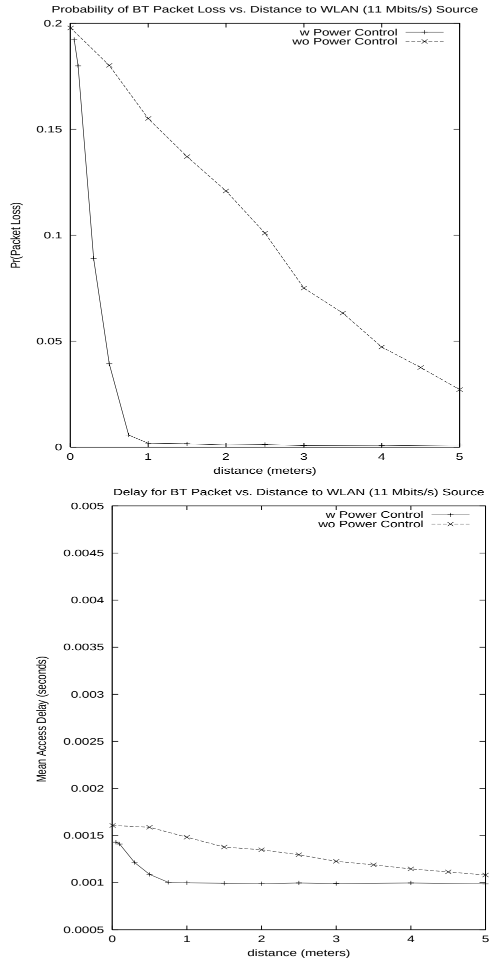


Fig. 4. $\boxed{\frac{(a)}{(b)}}$ Effect of Adaptive Power Control on Bluetooth Performance. (a) Probability of Packet Error vs. Distance. (b) Mean Access Delay vs. Distance.

control algorithm. Note that the WLAN transmitted power is fixed at 25 mW. For distances equal to 0.5 m from the interference source, increasing the transmitted power leads to lower packet losses, ≈ 4 % with power control instead of 18% without power control. A similar reasoning applies to the delays shown in Figure 4(b). However for distances less than 0.5m, the transmitted power is capped by P_{max} and the packet loss remains higher than $\approx$ 9%. A couple of observations are in order. We note that the power control algorithm can be effective in some scenarios; in the case studied here, lower packet losses and access delays are obtained for distances greater than 0.5m from the interference source. However, it should be made clear that this performance gain comes at the cost of increasing the interference level for other systems. As expected, increasing the Bluetooth transmitted power, has a negative impact on the interfering system; in Figure 5 we note a 17% packet loss at the WLAN AP device, even if it is about 15 meters away from the Bluetooth devices. As the Bluetooth transmitted power is weakened, the packet loss at the WLAN AP device drops to zero.

In a way, adjusting the power control can only be a partial solution. This may or may not constitute a problem for other systems depending on the configuration and the parameters used.

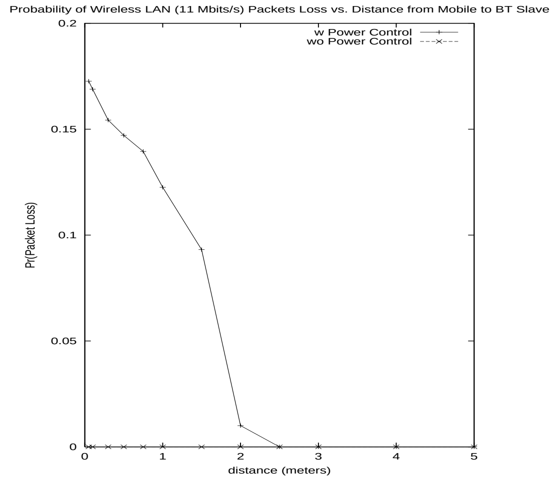


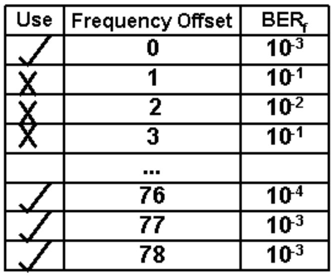
Fig. 5. Impact on the WLAN AP Device

III. MAC SCHEDULING

In this section, we investigate how scheduling techniques can be used to alleviate the impact of interference. We devise a mechanism for the Bluetooth MAC scheduler consisting of two components:

1. *Interference Estimation*
2. *Master Delay Policy*

In the *Interference Estimation* phase, the Bluetooth device detects the presence of an interfering device occupying a number of frequencies in the band. In this sequel, interfering devices are assumed to be WLAN DSSS systems. In order to detect the presence of interference, the Bluetooth device maintains a *Frequency Usage Table* where a bit error rate measurement, BER_f, is associated to each frequency as shown in Figure 6. Note that, a frame error rate or a packet loss measure can be used instead of the bit error rate (BER). Frequencies are classified according to a criteria that measures the level of interference in the channel and marked *used* or *unused* depending on whether their corresponding BER is above or below a threshold value, BER^T, respectively. This *Frequency Usage Table* is maintained at each receiver's side for both master and slave devices.



Use	Frequency Offset	BER_f
✓	0	10^{-3}
X	1	10^{-1}
X	2	10^{-2}
X	3	10^{-1}
	...	
✓	76	10^{-4}
✓	77	10^{-3}
✓	78	10^{-3}

Fig. 6. Frequency Usage Table

The *Master Delay Policy* makes use of the measurements collected during the *Interference Estimation* phase in order to avoid a packet transmission in a "bad" receiving channel, or a channel with a high level of interference. The basic idea is to "wait" for or choose an *unused* frequency for the receiver in the frequency hopping pattern. Thus the transmitter needs to consult the receiver's *Frequency Usage Table* before transmitting any packets. Alternatively, the receiver, can send status updates on its usage table to the transmitter. In Bluetooth, since the master device controls all transmissions in the piconet, the delay rule has to be implemented only in the master device. Furthermore, since following each master's transmission, there is a slave transmission, the master checks both the slave's receiving frequency and its own receiving frequency before chosing to transmit a packet in a given frequency hop as illustrated in Figure 7.

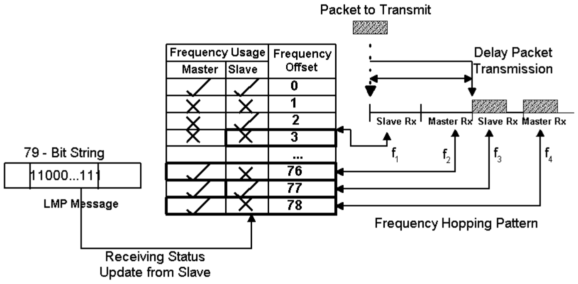


Fig. 7. Delay Scheduling Policy at Bluetooth Master

The main steps of the scheduling policy are summarized as follows.

1. Slave's End.
   (a) For every packet received, update BER_f which is an average value of the BER per frequency.
   (b) Every update interval, U, refresh the *Frequency Usage Table* by marking the frequencies, and
   (c) Send a status update message to the Master;
2. Master's End.
   (a) For every packet received, update BER_f which is an average value of the BER per frequency.
   (b) Every update interval, U, refresh the *Frequency Usage Table*, and
   (c) Before sending a packet, check slave's receiving frequency and master's following receiving frequency, delay transmission until both master and slave's receiving frequencies are available.

Implementation Considerations One of the advantages in using this scheduling policy is that it does not require any changes in the FCC rules. In fact, title 47, part 15 of the FCC rules on radio frequency devices [7], allows a frequency hopping system to recognize the presence of other users within the same spectrum band so that it adapts its hopsets to avoid hopping on occupied channels. However, coordination among hopping frequency systems in order to avoid simultaneous channel occupancy is not allowed.

Furthermore, scheduling in the Bluetooth specifications is vendor implementation specific. Therefore, one can easily implement a scheduling policy with the currently available Bluetooth chip set. Most importantly, the proposed scheduling algorithm does not require any changes to the Bluetooth frequency hopping pattern which is implemented in ASICs, and devices implementing scheduling can easily interoperate with other devices that do not.

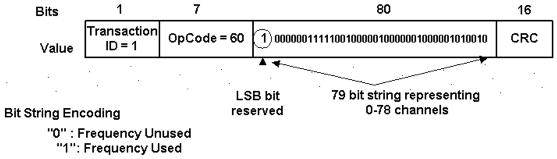


Fig. 8. LMP Interference Status PDU

As far as the status update message is concerned, we define an $LMP\_Interference\_Status$ PDU as shown in Figure 8. We use an Op_code value of 60 and set the $Transition\ ID$ to 1 in order to indicate that the message is sent from the slave to the master. The content field uses 10 bytes to encode the slave's *Frequency Usage Table*. In fact, we reserve one bit for future use, and map the 79 channels in the *Frequency Usage Table* to a 79-bit string of 0's and 1's indicating the *used* and *unused* receiving frequencies respectively.

Numerical Results We simulate our proposed scheduling policy. We use the simulation environment, network topology and parameters described in section II. We use three types of Bluetooth packet encapsulations, namely, $DM1$, $DM3$, and $DM5$, that occupy 1, 3 and 5 slots respectively. The offered load for Bluetooth is set to 30% of the channel capacity which corresponds to a packet interarrival of 2.91 ms, 8.75 ms and 14.58 ms for $DM1$, $DM3$ and $DM5$ packets respectively. The transmitted power for Bluetooth and WLAN is fixed at 1mW and 25 mW respectively. Simulation parameters are summarized in Table III. Figure 9 (a) and (b) gives the packet loss and the mean access delay measured at the Bluetooth slave for varying distances of the interference source from the Bluetooth receiver. From Figure 9 (a) we observe that using the scheduling policy, leads to a packet loss of zero. We are basically able to avoid the channels occupied by the interfering system. When no scheduling policy is used the packet loss is $\sim$ 24% for $DM5$, and $DM3$, and 19% for and $DM1$ packets respectively when the Bluetooth receiver is at a distance of 0.005 meters from the interference source. As the distance from the interference source is increased the packet loss drops to around 2.7% for $DM1$ packets. It is still around 6.7% for $DM3$ and $DM5$ packets.

For $DM1$, we observe an increase in delay from 1.6ms to 2.6ms when the scheduling policy is applied. On average the scheduling policy yields to a delay increase of 1ms ($\sim$ 1.6 Bluetooth slots). On the other hand, the scheduling policy reduces the delays by 0.8 ms and 2.6 ms for $DM3$ and $DM5$ respectively. Thus, delaying transmission to avoid bad channels pays off for packets occupying more than one slot. Note that, when bad channels are used, packets are dropped and have to be retransmitted which yields large delays. This effect does not apply to $DM1$ packets since they occupy only one slot.

In summary, we note that the scheduling policy is effective in reducing packet loss and delay (especially for multi-slot Bluetooth packets). Another advantage worth mentioning, are the additional savings in the transmitted power since packets are not transmitted when the channel is bad. Moreover, we note that by avoiding channels occupied by other devices, we elimi-

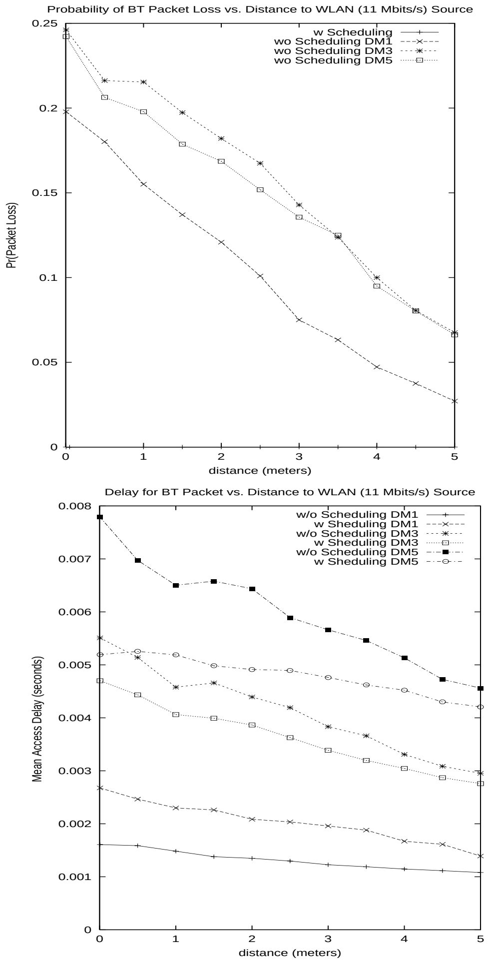


Fig. 9. $\boxed{\frac{(a)}{(b)}}$ Effect of MAC Scheduling on Bluetooth Performance. (a) Probability of Packet Error vs. Distance. (b) Mean Access Delay vs. Distance

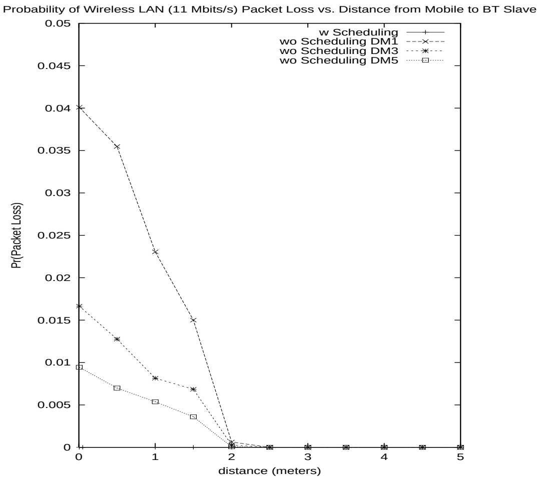


Fig. 10. Impact of MAC Scheduling on the WLAN Mobile Device

nate interference on the other system sharing the same spectrum band. Figure 10 shows the packet loss for the WLAN Mobile device (receiving ACKs). We note that scheduling reduces the ACK packet loss to zero. Therefore scheduling can be considered as a neighbor friendly policy. Note that the packet loss at the WLAN AP located at (0,15) m is negligible in this case since the Bluetooth signal is too weak.

Finally, we note that scheduling policy proposed here works only with data traffic since voice packets need to be sent at fixed intervals. However, if the delay variance is constant and the delay can be limited to a slot (as was shown here), it may be worthwhile to use DM packets for voice using the same scheduling technique proposed here. This will constitute the basis of future work.

IV. CONCLUDING REMARKS

In this paper we explored two techniques for alleviating the impact of interference on the Bluetooth performance. While the power control approach may be useful and simple to use in some limited scenarios, it can only be a partial solution and thus can not be considered by itself. Our plan is to test the dynamics of the power control algorithm simultaneously on both the WLAN and the Bluetooth systems in order to gain additional insights on its strengths and limitations in the context of interference.

Conversely, our simulation results indicate that the simple scheduling technique that we propose to delay the transmission of Bluetooth data packet once interference is detected can significantly lower the probability of packet loss for Bluetooth without much increase in the mean access delay.

The performance evaluation results obtained for the Bluetooth ACL link seems to be promising. We are currently looking at additional scenarios, and traffic conditions. We are also investigating the use of combined approaches such as packet encapsulation, scheduling, and ARQ flow control. Other future directions consist of exploring the interoperation of the coexistence techniques developed for Bluetooth and WLAN in dynamically changing environments in order to unravel their strengths and limitations.

REFERENCES

[1] Bluetooth Special Interest Group, “Specifications of the Bluetooth System, vol. 1, v.1.0B ’Core’ and vol. 2 v1.0B ’Profiles’,” December 1999.

[2] IEEE Std. 802-11, “IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification ,” June 1997.

[3] G. Foschini, Z. Miljanic, “A Simple Distributed Autonomous Power Control Algorithm and its Convergence ,” in *IEEE Journal on Vehicular Technology*, November 1993, vol. 42.

[4] J. Zander , “Distributed Cochannel Interference Control in Cellular Radio Systems,” in *IEEE Journal on Vehicular Technology*, February 1992, vol. 41.

[5] Bambos, N., “Toward power-sensitive network architectures in wireless communications: concepts, issues, and design aspects,” in *IEEE Personal Communications*, June 1998, vol. 5, pp. 50–59.

[6] M. Andersin, N. Mandayan, J. Zander, “A Subspace based Estimation of the Signal to Interference Ratio for TDMA Cellular Systems,” in *IEEE VTC’96*, Atlanta, GA, April 1996.

[7] Federal Communications Commission, “Title 47, Code for Federal Regulations, Part 15,” October 1998.