Channel Clustering and Probabilistic Channel Visiting Techniques for WLAN Interference Mitigation in Bluetooth Devices

Qixiang Pang and Victor C. M. Leung, *Fellow, IEEE*

***Abstract*—Since Bluetooth and wireless local area network (WLAN) technologies both operate at the 2.4-GHz industrial, scientific, and medical (ISM) band, the two types of devices may suffer from mutual interference and performance degradations. In this paper, we propose two new techniques, channel clustering and probabilistic channel visiting, to effectively improve the existing coexistence and interference mitigation mechanisms. The channel clustering technique employs statistical pattern recognition to classify the status of Bluetooth channels more accurately. The probabilistic channel visiting is used to more equitably allocate the channel resources between Bluetooth and WLAN devices. The effectiveness of these techniques is quantified by simulations. Results show that both techniques are beneficial in improving the performance of the existing mechanisms.**

***Index Terms*—Bluetooth, coexistence, interference mitigation, wireless local area network (WLAN).**

I. Introduction

IEEE 802.11 wireless local area networks (WLANs) [1] and IEEE 802.15.1 Bluetooth (BT) wireless personal area networks (WPANs) [2] have been widely deployed. Because both systems operate in the 2.4-GHz industrial, scientific, and medical (ISM) frequency band, mutual interference between them may result in severe performance degradations [3], and coexistence of these systems over a shared electromagnetic spectrum becomes an important issue.

Recently, the interference issue has attracted many research and standardization activities [3]–[8]. For example, the IEEE 802.15.2 standard [3] includes eight coexistence and interference mitigation mechanisms that can be classified into two categories: collaborative and noncollaborative. The collaborative mechanisms can be used when the WLAN and BT devices are integrated within the same physical unit and exchanging information between them is feasible. The noncollaborative mechanisms are designed for application in BT devices when they are not collocated with the interfering WLAN devices within the same physical units. In practice, since many WLAN and BT devices are independently implemented and installed, the noncollaborative mechanisms are more practical. In the following discussions, we shall concentrate on the noncollaborative mechanisms.

The existing noncollaborative mechanisms all use similar techniques to detect the presence of WLANs in the ISM band and classify the BT channels as "bad" if they are subject to WLAN interference, or "good" otherwise. For example, each BT device can estimate the packet-error rate (PER) or received signal strength indication (RSSI) for each channel, and a channel is marked "bad" if the PER or RSSI value exceeds a predefined threshold [3]–[5]. The device then modifies its frequency-hopping pattern to avoid the "bad" channels, or chooses not to transmit when hopping onto a "bad" channel. The existing channel classification methods are simple, but they are effective only when a single BT piconet exists in the area of interest. When there are multiple piconets within radio range, these methods can be misled as a high PER or RSSI can result when more than one piconet hops onto the same channel. To address this issue, sensing of actual WLAN carriers could be an effective option, which is, however, quite complex and possibly too costly to implement in low-cost BT devices due to physical layer differences.

Besides channel classification, a related problem is how often a BT piconet should visit the channels after they have been classified. In the existing mechanisms, once a channel is marked as "bad," it will not be visited at all. However, channel conditions are subject to changes and the channel classification can be misled by the presence of other BT piconets, as discussed earlier. When multiple WLANs and/or BT piconets coexist, the number of "good" channels can be very low and the BT devices may be starved of radio resources due to their excessive courtesy.

Therefore, the existing coexistence mechanisms may not work effectively in some practical situations. This paper contributes novel solutions to the respective problems discussed earlier by proposing the channel clustering classification method and probabilistic channel visiting. The current work substantially improves and extends our preliminary work [9] in the following aspects: 1) a detailed analysis of the problems in the existing PER-based and RSSI-based channel classification methods; 2) a detailed analysis of the problem in the existing channel visiting methods; 3) a refined channel classification method compared with [9] and other previous work [3]–[7]; 4) an analytical evaluation of the proposed scheduling mechanism; and 5) additional simulation scenarios and more detailed discussions on the performance results.

Manuscript received October 3, 2006; revised February 27, 2007 and May 24, 2007. This work was supported in part by the Canadian Natural Sciences and Engineering Research Council under Grant STPGP 257684-02 and in part by the OPNET University program. This paper was presented in part at the IEEE VTC-Spring, May 2006.

The authors are with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada (e-mail: qixiangp@ece.ubc.ca; vleung@ece.ubc.ca).

Digital Object Identifier 10.1109/TEMC.2007.908262

The rest of this paper is organized as follows. Section II reviews the problems with the existing noncollaborative coexistence mechanisms and channel classification methods in more detail. Section III describes the two new techniques: channel clustering and probabilistic channel visiting. Section IV gives the performance evaluation results. Section V concludes this paper.

II. Problems in Existing Coexistence Mechanisms

A. *Problems in Existing Channel Classification Methods*

Channel classification is performed by BT devices as the first step in all noncollaborative coexistence mechanisms. Each BT device follows a pseudorandom sequence to hop over 79 or 49 channels that are each 1 MHz wide. The purpose of channel classification is to determine whether a channel is subject to interference from other communication systems. Current coexistence mechanisms [3] are mainly concerned with interference between WLANs and BT piconets, which is also the focus of this paper.

Several methods of BT channel classification are recommended by the IEEE 802.15.2 standard [3] and related literature: PER, RSSI, and carrier sensing.

1) PER—At the expiry of the classification period, the PER for each of the BT channels is estimated. A channel is declared "bad" if its PER exceeds the system-defined threshold.
2) RSSI—If the RSSI in a BT channel is higher than a threshold and an error is detected, the channel is likely suffering from interference and is considered as a "bad" channel.
3) Carrier Sensing—It requires that the BT device has the ability to carrier-sense the existence of IEEE 802.11 PHY signals interfering with one or several BT channels.

Among the three methods, carrier sensing is the most robust. However, the challenge of implementing in a cheap BT device the ability to sense the carrier of another communication system employing a totally different signaling method is not easy to overcome. The other two methods (PER and RSSI estimation) are simpler to implement and have been the channel classification methods considered in most literature. However, the existing methods all use predefined PER and RSSI thresholds that can be inappropriate when multiple BT piconets coexist, as in the cases described in [11]–[13], since a high PER or RSSI can be caused not only by a coexisting WLAN, but by other BT devices in coexisting piconets as well.

1) Problem With RSSI-Based Channel Classification Method: The RSS at the receiver is determined by the transmit power, path loss, and spectrum factor. We consider two types of signals, 802.11b WLAN and BT. Usually, the transmit power of a BT device is 0 dBm (1 mW) and that of a WLAN device is 14 dBm (25 mW). At 2.4 GHz, as described in [3] and [10], the path loss model is given by

$$\text{path\_loss} = \begin{cases} 40.2 + 20\log_{10}(d), & 0.5m \le d \le 8m \\ 58.5 + 33\log_{10}(d/8), & d > 8m \end{cases} \quad (1)$$

where d is the distance (meters). The spectrum factor represents the combined effects of transmitter and receiver masks and the offset between the interference and the wanted signal center frequencies. As calculated by [3], if the frequency offset is 0, the spectrum factor from WLAN to BT is -12.6 dB and that from BT to BT is unity or 0 dB.

Although the transmit power of a WLAN device is much higher than that of a BT device, when all three aspects: transmit power, path loss, and spectrum factor have been taken into account, it can be concluded that a BT device in a different piconet may generate an interference signal as strong as that from a WLAN device. For example, the RSS due to a WLAN device that is 3 m away is about -48.4 dBm, which is the same as that due to a BT device that is 2.6 m away. Therefore, if the BT device uses the RSSI-based method to classify the channel status, it may wrongly mark the channels in which frequency hopping collisions with other BT devices occur as "bad." Thus, over time, the BT device may have few or no channels marked as "good" to use.

2) Problems With PER-Based Channel Classification Method: We use the BT system model proposed in [13] to analyze the collision probability when multiple piconets coexist. Consider N collocated piconets that are sufficiently close to one another such that a collision between two or more packets over the same hop channel will destroy all packets involved. Forward error correction and capture effect are neglected. Assume that all packets occupy a single slot and the probability to have a packet in a slot is G.

First, we consider the case where all piconets are synchronized so that the start of each time slot is aligned. For a one-slot packet in piconet A, the probability of an unsuccessful transmission in the presence of another synchronized piconet B is equal to GP_0, where $P_0 = 1/m$ is the probability that piconet B chooses the same hop frequency as piconet A and m is the size of the hop set. With N collocated piconets, piconet A has $N-1$ adversary piconets. The probability of an unsuccessful transmission is

$$p_{\mathrm{bt}} = 1 - (1 - GP_0)^{(N-1)}. \quad (2)$$

Next, we consider a more general case where different piconets are not synchronized and the actual duration of a packet is smaller than the duration of a slot. Assume the duration of a slot is t_s μs and the transmission time of the valid bits of a single-slot packet is t_d μs. Depending on the relative time phase, one or two slots from the adversary piconet B can interfere with the packet of interest in piconet A. The time shift between A and B is a random variable uniformly distributed between 0 and t_s. The probability that the time shift is such that a packet in piconet B is a potential threat to two packets in piconet A can be expressed by $2t_d/t_s - 1$. The probability that the time shift is such that a packet in piconet B is a potential threat to only one packet in piconet A is the complement, $r = 2(1 - t_d/t_s)$. When all possible time shifts are taken into consideration, the mean value of the collision probability, p_{bt}, for the N piconets is given by

$$p_{\mathrm{bt}} = 1 - \left(r(1 - GP_0) + (1 - r)(1 - GP_0)^2\right)^{N-1}. \quad (3)$$

Fig. 1 shows the collision probabilities given by (2) for synchronized piconets and (3) with $r = 0$ for unsynchronized

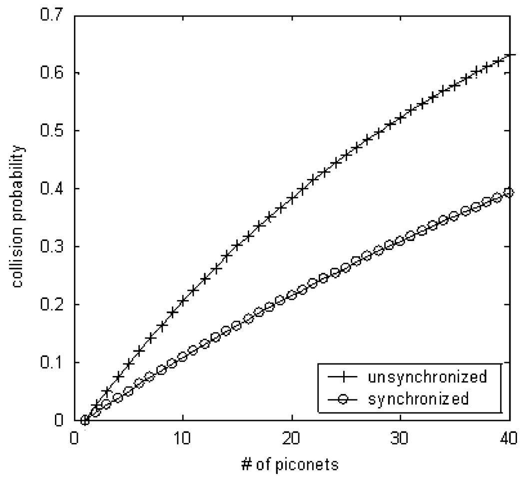

Fig. 1. Frequency hop collision probability between multiple BT piconets.

piconets assuming $G = 1$ and $m = 79$. When the number of piconets is large, the collision probability can be quite high. When multislot packet transmissions are considered, the collision probability between piconets can be even higher [14], [15]. On the other hand, since the traffic from the WLAN devices is variable, the collision probabilities between BT and WLAN devices are also variable. Therefore, it is inaccurate to label a channel as one experiencing "bad" interference from WLANs, simply according to its PER without considering the existence of other piconets.

The existing literature only considers using a fixed threshold, which is predefined and can only be changed manually after it has been set. Different papers considered different threshold values without any common selection rule; e.g., in [4], the PER threshold is 0.5, while, in [5], the bit-error rate (BER) threshold is 0.001 (equivalent to a PER of about 0.3 if the packet size is 366 bits). In [6] and [7], a channel is considered "bad" if a single packet error has occurred, which can be very inaccurate in classifying a channel for WLAN interference.

Besides the PER threshold problem, another problem is with the PER estimation period in the existing methods. In order to obtain reliable PER statistics, the estimation period must be long enough so that sufficient numbers of packet transmissions have occurred over all BT channels. Consider the following example.

Assume 50% of the BT time slots are utilized for transmissions of single-slot packets and the PER estimation period is 2 s; then, we have $2 \times 50\%/(1/800) = 800$ packet transmissions over this time period. As the 800 packet transmissions span the 79 channels, on average, there are about ten transmissions in each channel. According to the theory of statistics [16], when the sample size n is so small, the confidence that the estimated PER $(\hat{p}_e)$ is close to the actual PER (p_e) is very low. For example, let the allowable difference between $\hat{p}_e$ and p_e, D, be 0.1 and $p_e = 0.5$. With $n = 10$, the confidence given by prob$(|\hat{p}_e - p_e|) < D$ is only 46%. If the estimation period is doubled to 4 s ($n \cong 20$), the confidence is still only 62%. When the PER takes on other values, e.g., from 0.1 to 0.9, the computed confidence level is very low too.

Therefore, the existing channel classification methods, which classify the channels individually, can give results that are far from accurate. In order to increase the accuracy of PER estimations, the period must be very long, which can increase the interference between WLAN and BT devices, because during this period, all BT channels regardless of good or bad conditions must be visited for BT packet transmissions.

In order to solve the two problems related to PER-based methods, we utilize the fact that the bandwidth of an IEEE 802.11b/g channel is 22 MHz and, therefore, a WLAN transmission may interfere with a cluster of BT channels.[1] This property tells us that the "bad" BT channels, in general, span a range of consecutive frequencies. The existing channel classification methods do not take advantage of this available information and are, therefore, not very efficient.

B. Problem With Existing Channel Visiting Methods

Once the channels have been classified into two classes: "bad" and "good," in terms of interference from WLANs, the next issue is how a BT system visits them differently.

In the existing coexistence mechanisms [3], once a channel has been marked as "bad," it will not be visited at all. For example, in the existing coexistence mechanisms employing packet scheduling [3], [6], if one frequency in the current hop frequency pair (master to slave and slave to master) is "bad," the BT packet(s) will be backlogged until a pair of "good" frequencies appear. This rigid channel visiting scheme may cause problems in the existing coexistence mechanisms.

In this paper, we mainly investigate the coexistence mechanism employing packet scheduling. We shall leave the evaluation of other coexistence mechanisms for future work. The advantages of scheduling over other noncollaborative mechanisms are that: 1) it can be implemented entirely in the master node and 2) it does not require changes to the BT physical specifications. In the following, we derive some analytical results for the scheduling mechanism.

Assume there are multiple BT piconets that are subject to interference from WLANs. These BT piconets follow the same assumptions as stated in Section II-A. Both master and slave have packets to transmit. Assume there are m_g "good" channels and $m_b = m - m_g$ "bad" channels in both directions (i.e., master to slave and slave to master). Let $p_g = m_g/m$ and $p_b = m_b/m$. We define the normalized throughput S of a BT piconet as the ratio of packets transmitted successfully.

When the scheduling mechanism is not applied, the probability that a downlink (master to slave) BT packet transmission is successful (no collision with another BT packet or a WLAN packet), p_d, is given by

$$\begin{aligned} p_d &= p_g(1 - p_{\mathrm{bt}}) + p_b(1 - p_{\mathrm{bt}})(1 - p_w) \\ &= (1 - p_{\mathrm{bt}})(1 - p_b p_w) \end{aligned} \tag{4}$$

where p_{bt}, the probability of a BT packet transmission colliding with other BT packets in a channel, is given by (2) or (3)

[1]The two channels adjacent to the two edges of the 22-channel cluster may also be interfered by the WLAN. In this paper, for simplicity, we assume that only 22 BT channels are interfered by a WLAN.

depending on whether the piconets are synchronized or not, and p_w is the probability that a BT packet collides with a WLAN packet in a "bad" channel. The probability p_u that an uplink (slave to master) BT packet transmission is initiated and the transmission is successful is given by

$$p_u = p_d(1 - p_{\mathrm{bt}})(1 - p_b p_w). \tag{5}$$

Therefore, the normalized throughput can be expressed by

$$\begin{aligned} S &= G(p_d + p_u)/2 \\ &= G\left((1-p_{\mathrm{bt}})(1-p_b p_w)+(1-p_{\mathrm{bt}})^2(1-p_b p_w)^2\right)/2 \end{aligned} \tag{6}$$

where G is the packet transmission probability in a slot, as defined in Section II-A.

When the existing scheduling mechanism is applied, it requires both downlink and uplink BT channels to be good to transmit [3], [6]. Therefore, the probability that a downlink BT packet transmission is successful (i.e., it is neither delayed by the scheduler nor lost due to collision with another BT packet) becomes

$$p_d = p_g^2(1 - p_{\mathrm{bt}}). \tag{7}$$

In this case, p_u is given by

$$p_u = p_d(1 - p_{\mathrm{bt}}). \tag{8}$$

The normalized throughput is then

$$S = G\left(p_g^2(1 - p_{\mathrm{bt}}) + p_g^2(1 - p_{\mathrm{bt}})^2\right)/2. \tag{9}$$

Assume each WLAN interferes with 22 BT channels and the BT piconets are unsynchronized with the probability that a packet from an interfering piconet overlaps with only one packet in the interfered piconet $r = 2(1 - 366/625) = 0.8288$. Let $m = 79$. Further, assume the probability to have a BT packet transmission in a BT slot is 0.5 (i.e., $G = 0.5$) and the WLAN traffic occupies its channel 50% of the time (i.e., $p_w = 0.5$). The numerical results in Fig. 2 show that using the existing scheduling method, interference avoidance with WLAN may sacrifice the BT throughput when BT piconets have high throughput requirements and the WLAN does not occupy its channel 100% of the time.

When there are multiple WLANs, or the BT system uses only 49 channels, or a WLAN interferes with 24 BT channels, the problem can become even worse. In other coexistence mechanisms, e.g., adaptive frequency hopping [3], similar problems exist as well.

III. Proposed Mechanisms

From the discussions in Section II, we have the following three observations.

1) When there are multiple BT piconets within radio range of each other, the classification of channels using a predefined PER or RSSI threshold can be invalid.
2) The interference from WLAN spans multiple consecutive BT channels.
3) In the existing coexistence mechanisms, the rigid channel visiting method may not satisfy the BT throughput and delay requirements, but, instead, cost BT devices processing power and energy resource, which can be unfair to these devices.

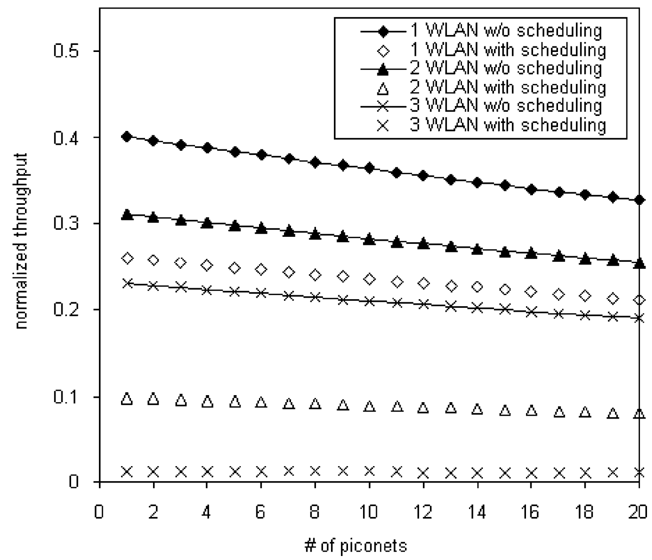

Fig. 2. Comparison of BT throughputs with and without the original scheduling mechanism.

Based on observations 1) and 2), we propose a new channel classification technique—channel clustering. Based on observation 3), we propose a new probabilistic channel visiting technique. These techniques are complementary in improving the performance of the existing BT/WLAN coexistence mechanisms.

A. Channel Clustering and Classification

The proposed channel clustering technique also utilizes the estimated PER. However, different from the existing PER-based channel classification methods, it does not depend on a predefined PER threshold, but uses the intrinsic pattern of the observed PER values.

The channel classification problem is similar to the well-studied topic of segmentation or edge detection in statistical pattern recognition [17]. The "bad" channels occupied by a WLAN usually cluster together and have a higher average PER than the "good" channels due to the possibility of collisions with both WLAN and BT packets, as illustrated in Fig. 3. In order to detect a "bad" channel cluster, we only need to know the edges of the cluster. Therefore, the essential task in channel clustering is to detect the edges of the clusters.

The basic idea in edge detection is straightforward, and it is described as follows. Let m be the number of channels and $\hat{p}_{e,i}$ be the estimated PER of the ith $(1 \le i \le m)$ channel. The edges are located at the places where the PERs change most dramatically. In order to smooth out the randomness caused by the stochastic fluctuations of the estimated PERs, we define block PER (BPER) as the average PER over B (e.g., 5) consecutive channels starting from the ith channel

$$\mathrm{BPER}_i = \frac{1}{B}\sum\nolimits_{j=0}^{B-1} p_{e,i+j}. \tag{10}$$

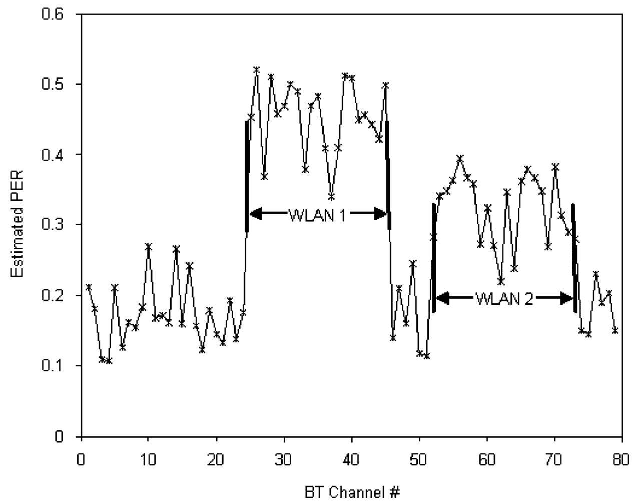

Fig. 3. Example of channel clustering based on PER estimates.

The first step to detect the lower edge of a channel cluster is to find the place where the block PERs show the steepest ascent

$$s = \underset{i:1\leq i\leq m}{\arg\max}(\mathrm{BPER}_i - \mathrm{BPER}_{i-B}). \tag{11}$$

When the potential lower edge of a WLAN channel cluster (i.e., channel s) has been found, the second step is to check whether it is really a channel cluster. If the majority (e.g., 75%) of the channels from s to $s+W-1(W$ is the width of the cluster; for 802.11b, $W=22$) have a higher PER than their left adjacent "good" channels (BPER_{s-B}), then very likely the channels s to $s+W-1$ form a "bad" channel cluster. Otherwise, it is not regarded as a "bad" cluster. The procedure can be iterated to detect the existence of multiple WLANs.

By using this procedure, in most cases, the channel clusters occupied by WLANs can be correctly recognized. The accuracy can be further increased if we consider the upper edge of a WLAN channel cluster as well. The procedure to identify the upper edge is similar to the one to identify the lower edge. The difference is that the point where there is the steepest descent in PER is detected. Both identified edges can be used to more accurately determine the locations of the WLAN channels. This is particularly useful when partially overlapping WLAN channels are in use and the bad channel cluster consists of more than 22 BT channels.

One prominent advantage of the proposed method is that it does not depend on a predefined PER threshold. Other advantages of the new technique include taking less time and generating less interference to the WLAN channels compared with the existing classification methods. In order to classify the channels, all channels must be visited and their PERs estimated. During this all-channel-visiting phase, interference between WLAN and BT devices is unavoidable. In order to achieve a sufficient confidence in the PER statistics, this phase must be long enough. In the proposed method, because the cluster of BT channels interfered by a WLAN is considered as a whole, the requirement on the accuracy of the estimated PER of a single channel is not necessarily high. Therefore, the proposed method consumes less time to estimate PERs and generates less interference between WLAN and BT devices. This is illustrated by the numerical results in Section IV.

B. Probabilistic Channel Visiting

We have shown in Section II-B that when the existing coexistence mechanisms are enabled, the performance of multiple coexisting BT systems can be degraded. In order to mitigate interference and achieve an overall better performance of the involved wireless systems, we introduce a new channel visiting technique—probabilistic channel visiting. Instead of never visiting a channel that is marked as "bad," the channel is visited with probability p_v. This visiting probability can be dynamically adjusted according to the variations of the BT traffic load and the environmental conditions. Using this idea, the existing coexistence mechanisms can be improved. For example, the existing coexistence mechanism employing packet scheduling can be enhanced to incorporate probabilistic scheduling as follows. If the channel for the current time slot is "good," the packet is transmitted. If it is a "bad" channel, the probability to transmit the packet is p_v while the probability to postpone packet transmission is $1-p_v$. When the collision probability with WLANs is less than 1, a higher p_v can lead to a higher BT throughput. It can be seen that the probabilistic scheduling mechanism includes the existing scheduling mechanism as a special case, in which the visiting probability is always 0. When the visiting probability is always 1, it is equivalent to the case where the coexistence mechanism is disabled.

We use the same assumptions in Section II to analyze the performance of the probabilistic scheduling mechanism. When the master has a packet to transmit, it checks the states of both the slave's receiving frequency (f_d) and its own receiving frequency (f_u). There are, in total, four possible cases: 1) both f_d and f_u are good; 2) f_d is good and f_u is bad; 3) f_d is bad and f_u is good; and 4) both f_d and f_u are bad. With the existing scheduling method, the packet is transmitted only if Case 1 occurs, while, in all the other cases, the packets are delayed. In the probabilistic scheduling method, in all the cases, it is possible to transmit as long as the visiting probability p_v is not 0.

Therefore, by taking all the four cases into consideration, p_d and p_u, as defined in Section II-B, are given, respectively, by

$$\begin{aligned} p_d &= \underbrace{p_{d1}}_{\text{case 1}} + \underbrace{p_{d2}}_{\text{case 2}} + \underbrace{p_{d3}}_{\text{case 3}} + \underbrace{p_{d4}}_{\text{case 4}} \\ &= \underbrace{p_g^2(1-p_{\mathrm{bt}})}_{\text{case 1}} + \underbrace{p_g p_b p_v (1-p_{\mathrm{bt}})}_{\text{case 2}} \\ &\quad + \underbrace{p_b p_v p_g (1-p_{\mathrm{bt}})(1-p_w)}_{\text{case 3}} + \underbrace{(p_b p_v)^2 (1-p_{\mathrm{bt}})(1-p_w)}_{\text{case 4}} \end{aligned} \tag{12}$$

$$\begin{aligned} p_u &= \underbrace{p_{d1}(1-p_{\mathrm{bt}})}_{\text{case 1}} + \underbrace{p_{d2}(1-p_{\mathrm{bt}})(1-p_w)}_{\text{case 2}} \\ &\quad + \underbrace{p_{d3}(1-p_{\mathrm{bt}})}_{\text{case 3}} + \underbrace{p_{d4}(1-p_{\mathrm{bt}})(1-p_w)}_{\text{case 4}} \\ &= \underbrace{p_g^2(1-p_{\mathrm{bt}})^2}_{\text{case 1}} + \underbrace{p_g p_b p_v (1-p_{\mathrm{bt}})^2(1-p_w)}_{\text{case 2}} \end{aligned}$$

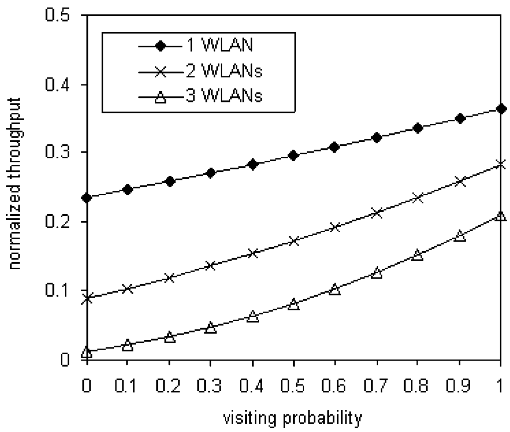

Fig. 4. Impact of different visiting probabilities on BT throughput.

$$+ \underbrace{p_b p_v p_g \left(1 - p_{\mathrm{bt}}\right)^2 \left(1 - p_w\right)}_{\text{case 3}}$$
$$+ \underbrace{\left(p_b p_v\right)^2 \left(1 - p_{\mathrm{bt}}\right)^2 \left(1 - p_w\right)^2}_{\text{case 4}}. \qquad (13)$$

The normalized throughput is then given by $S = G(p_d + p_u)/2$.

Fig. 4, which is based on the same parameters as Fig. 2, shows that by increasing the visiting probability, the throughput that BT devices can achieve is significantly improved.

In order to accommodate different traffic requirements, a dynamic p_v adjustment mechanism must be implemented. It is possible to build complicated mathematical models based on different objectives such as quality-of-service (QoS) or fairness, and to use complex control schemes, e.g., proportional, integral, derivative (PID) control. However, in keeping with the restriction imposed by the low processing ability of BT devices and the noncollaborative nature of the coexistence mechanism, in this paper, we propose a simple binary linear control scheme.

The p_v value is adjusted periodically according to the buffer occupancy $(\hat{B})$ in the BT device, which reflects the BT traffic load. If $\hat{B}$ is bigger than a threshold B^*, p_v is increased by a value Δ. Otherwise, it is reduced by the same amount. The adjustment period can be specified by a timer or a counter and $\hat{B}$ can be sampled and calculated by a moving average. In implementation, an upper bound (e.g., 0.9) can be set for the p_v values. Also, note that when a channel is extremely bad (say PER > 0.9), the payback for a BT device to visit this channel is limited and, in this case, the visiting probability can be set to 0.

In this scheme, the top priority for a BT device to consider is to satisfy its own throughput and delay requirements. In practice, this consideration is reasonable because the BT and WLAN devices likely belong to two different users. As the coexistence mechanism is implemented in the BT devices only, it is unreasonable to allow such a mechanism to unduly degrade BT performance when it is applied. On the other hand, once the BT throughput requirement has been met, p_v can be minimized so that the interference to WLANs can be reduced. Besides, one advantage of the self-adaptive probabilistic scheduling mechanism is that it is more resilient to the possible inaccuracy of the channel classification due to the dynamic p_v adjustment.

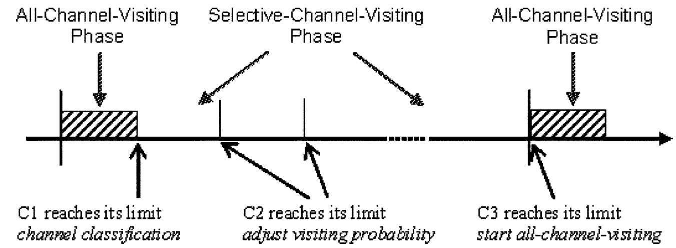

Fig. 5. Control procedure of the new noncollaborative coexistence mechanisms.

It is possible that more than one WLANs coexist with the BT system and the traffic loads on these WLANs are different. In this case, the BT PERs at different WLAN channels may be different. One may consider using different p_v values for the different channels, e.g., assigning a higher p_v for a channel with a lower PER, so that the BT throughput may be even higher. On the contrary, we suggest using the same p_v value. The rationale is that when a WLAN offers a lower traffic load, it expects to experience a better QoS (e.g., lower collision probability and delay). If the BT traffic on this WLAN channel is purposely adjusted to be higher than on other channels, it may violate the initial expectation of the low-traffic WLAN and is unfair to its users.

We summarize the features of the enhanced scheduling mechanism as follows: 1) probabilistic channel visiting; 2) self-adaptive; 3) controlled by BT traffic load; 4) better fairness for coexisting BT and WLAN devices; and 5) simple to implement.

C. Common Control Procedure

The common procedure of the new noncollaborative coexistence mechanisms proposed in this paper based on channel clustering and probabilistic channel visiting is shown in Fig. 5. There are two different channel visiting phases: all-channel-visiting and selective-channel-visiting. The all-channel-visiting phase is used for PER estimation and channel classification. The selective-channel-visiting phase is for interference mitigation.

Both timer-based (e.g., [4] and [6]) or counter-based methods can be used to delimit the phases. In this paper, we used a counter-based method as implementation of timers generally takes more resources and is less attractive for low-cost BT devices. The phases in the control procedure are delimited by counters C1, C2, and C3, which are incremented for every BT packet transmitted.

1) C1: The counter C1 is started from zero at the beginning of the all-channel-visiting phase. When C1 reaches a predefined limit (C1_LIMIT), the all-channel-visiting phase is terminated, the channels are classified using the cluster-based classification method described in Section III-A, and the selective-channel-visiting phase starts.
2) C2: The counter C2 is used during the selective-channel-visiting phase, and is started from zero at the beginning this phase; when C2 reaches its limit (C2_LIMIT), a visiting probability (p_v) adjustment is performed and the counter restarts from zero.

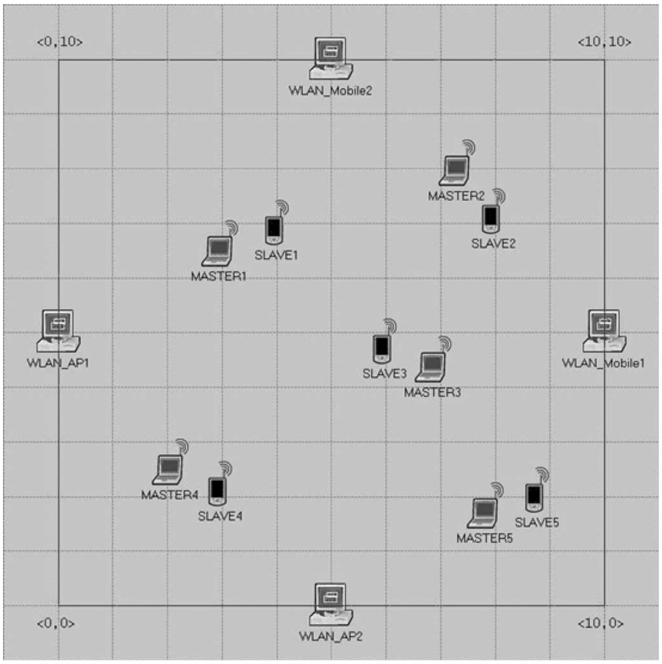

Fig. 6. Example of the simulated network topology in OPNET.

3) C3: The counter C3 is started from zero at the beginning of the selective-channel-visiting phase. When C3 reaches its limit (C3_LIMIT), an all-channel-visiting phase starts.

IV. Performance Evaluation by Simulations

A. Simulated Network Topology

The proposed coexistence mechanisms were implemented in OPNET with the SuiteTooth BT model from Highland Systems [18], and evaluated by simulations using the network topology shown in Fig. 6. The BT piconets are randomly placed in a rectangular area bounded by (0, 0), (0, 10), (10, 0), and (10, 10) (units in meters). The distance between the BT master and its slave is 1 m. The distance between the WLAN access point (AP) and its mobile station is 10 m. In the WLAN 1, the AP WLAN_AP1 and its mobile WLAN_Mobile1 are located at (0, 5) and (10, 5), respectively. In WLAN 2, WLAN_AP2 and WLAN_Mobile2, are located at (5, 0) and (5, 10), respectively. The numbers of WLANs and piconets vary in different scenarios. The physical data rate of each WLAN is 11 Mbps. The transmit power of a WLAN node is 25 mW (14 dBm), and that of a BT node is 1 mW (0 dBm).

B. Performance of Channel Clustering

In the simulations, the data traffic between the WLAN AP and mobile stations is bidirectional. The WLAN packet (medium access control (MAC) payload) size follows the NIST (National Institute of Standards and Technology) Internet packet size distribution [18] given in Table I, with an average size of 368 bytes. The interarrival time is exponentially distributed. The WLAN higher layer traffic data rate is 1 Mbps, and the traffic is evenly distributed between downlink (WLAN AP to station) and uplink (WLAN station to AP). The data traffic between BT master and slave is also bidirectional. The higher layer BT packets are 100 bytes with exponential interarrival times. The BT higher layer traffic data rate is 100 kb/s and the traffic is evenly distributed between downlink (master to slave) and uplink (slave to master). At the baseband layer of the BT system, the DH1 packet format is used.

TABLE I
NIST Packet Size Distribution Implemented in SuiteTooth Simulator

Packet size (bytes)	64	128	256	512	1024	1518
Probability	0.6	0.06	0.04	0.02	0.25	0.03

The value of C1_LIMIT has a direct impact on the effectiveness of channel clustering. To determine its proper value, both measurement accuracy and control promptness should be taken into account. As explained in Section II-A, when 800, or even 1600, packet transmissions are observed for channel classification, the estimated PER in each channel is far from accurate, which limits the effectiveness of the original channel classification method. In contrast, in our proposed method, because the channels are classified in clusters, the classification can be very accurate even if only a small number of packets have been transmitted over a short estimation period. The numerical results in the following will illustrate this point.

To avoid possible link asymmetry, channel classification can be performed for the downlink and uplink directions separately. The asymmetry may be caused by different distances between the interferer and the BT master and slaves, resulting in different received signal strengths and presence of "hidden terminals." However, the BT master and slaves are usually close to each other, and the quality of the BT link in either direction is comparable. Therefore, as an alternative, we can combine the PERs over both directions to get more accurate statistics.

Same as in Section II-A, we consider 800 and 1600 as the total number of packet transmissions over all BT channels and investigate the following three cases:

Case 1: C1_LIMIT = 800; downlink and uplink are classified separately.
Case 2: C1_LIMIT = 1600; downlink and uplink are classified separately.
Case 3: C1_LIMIT = 800; downlink and uplink PERs are combined to classify the channels.

To measure the effectiveness of the proposed channel clustering technique, define the channel identification ratio (idr) as

$$\mathrm{idr} = \frac{\sum_{i=1}^{N}\sum_{j=0}^{1}\sum_{k=1}^{m} b_{i,j,k}}{N \times 2 \times m} \tag{14}$$

where N is the number of piconets, m is the number of channels, and $b_{i,j,k}$ is defined by

$$b_{i,j,k} = \begin{cases} 1, & \text{if the } k\text{th channel in direction } j \text{ of} \\ & \text{the } i\text{th piconet is correctly identified} \\ 0, & \text{if the } k\text{th channel in direction } j \text{ of} \\ & \text{the } i\text{th piconet is incorrectly identified.} \end{cases} \tag{15}$$

Different numbers of WLANs and BT piconets are simulated, as described in Table II. For each simulation, a new random seed

TABLE II
COMPARISON OF CHANNEL CLASSIFICATION EFFECTIVENESS (GIVEN BY CHANNEL IDENTIFICATION RATIO) BETWEEN EXISTING AND PROPOSED CLUSTERING METHOD

# of WLANs	# of piconets	Case 1		Case 2 or 3	
		Clustering	Old method	Clustering	Old method
0	5	99.7%	99.6%	99.8%	99.7%
1	1	99.0%	85.7%	99.7%	86.4%
1	5	97.9%	78.6%	99.2%	78.8%
2	5	96.1%	60.7%	98.6%	61.3%
2	10	94.7%	60.5%	98.1%	60.9%

is used. Ten simulations were run for each scenario and the averages of the results are presented. Table II gives the simulation results of the channel clustering algorithm and the existing PER-based classification method. For the new classification method based on the proposed channel clustering technique, we identify the lower edges only. For the existing classification method, we adopt the PER threshold given by [5], i.e., 0.3: when the PER at a channel is higher than 0.3, it is regarded as a bad channel.

It can be seen that the proposed channel clustering technique is very effective and efficient even when the estimation period is short and the network topology is complex (e.g., multiple piconets and multiple WLANs). In all scenarios studied, the idr values are higher than or close to 95%. In contrast, a high accuracy cannot be achieved by the existing channel classification method, especially when many BT piconets and multiple WLANs coexist. The results also show that when C1_LIMIT is larger (Case 2) or the PERs in the two directions are considered together (Case 3), the idr value can be even higher. Note that for Case 2 and Case 3, the differences in the simulation results are negligible and the results for the two cases are presented in the same column in Table II.

Next, we evaluate the effectiveness of detecting both lower and upper edges. We consider two overlapping WLAN channels and purposely make 50% of their channel bandwidth overlap (note that, in practice, this does not happen often as WLANs are usually configured to operate over nonoverlapping channels). The number of piconets is set to 5. Again, C1_LIMIT and PER estimation period follow Case 1. When only the lower edges are considered, its idr is 85.7%. When both the edges are considered, the idr is improved to 97.6%. When the two WLAN channels are not overlapped, which is more commonly found in practice, a couple of more percents increase in idr can be obtained by considering both edges. As the idr values achieved by detecting the lower edges only are already very high in this case, the improvement is not as significant as in the case of overlapping WLAN channels.

C. Performance of Probabilistic Scheduling

In the following, we evaluate the performance of probabilistic scheduling and demonstrate how it can improve the original scheduling method. Three network topologies were simulated: 1) one piconet, one WLAN; 2) five piconets, one WLAN; and 3) five piconets, 2 WLANs with nonoverlapping channels.

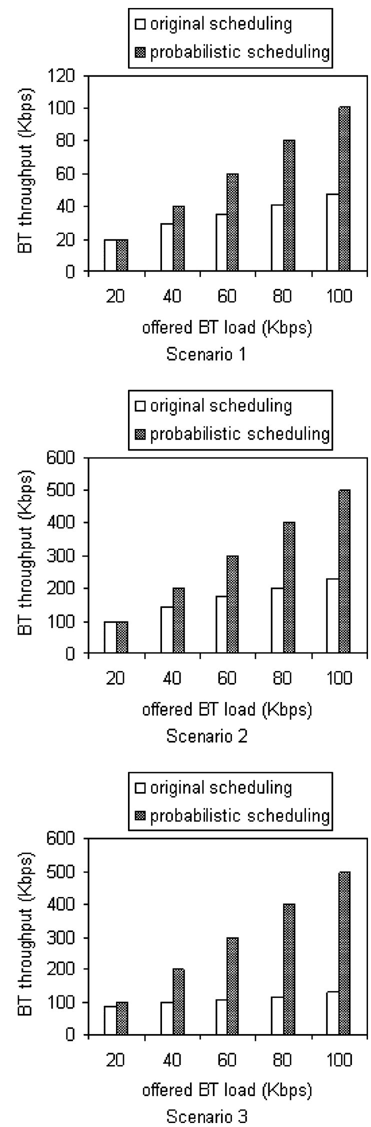

Fig. 7. Comparison of BT throughput per piconet.

In the channel clustering algorithm, only the lower edges are detected and the settings of C1_LIMIT and PER estimation period follow Case 1 defined in Section IV-B, as the classification effect is good enough. The WLAN traffic parameters are the same as those in Section IV-B. The BT packet interarrival time is varied to give different BT traffic loads.

The Δ value for p_v adjustment is 0.1 in the simulations. Buffer occupancy threshold B^* is assumed to be 1000 bytes. Appropriate values of B^* depend on the delay requirements of the BT traffic. A larger B^* generally results in longer delays for BT packets and vice versa. The value of C3_LIMIT depends on how often the network topology changes significantly. Heuristically, this time is in the order of minutes. When C3_LIMIT $= 48000$, and if the probability to have a BT packet transmission in a BT slot is 0.5, an all-channel-visiting is performed every 60 s. C2_LIMIT determines how often a p_v adjustment is needed. In our simulations, C2_LIMIT is set to 800.

Figs. 7–9 present the simulation results. It can be seen that the existing scheduling mechanism perform well only under a light

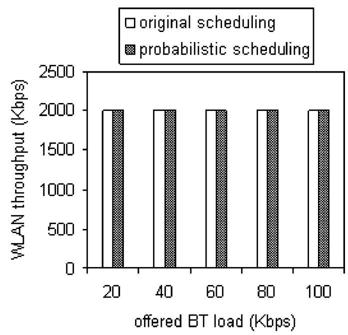

Fig. 8. Comparison of WLAN throughput (Scenario 3).

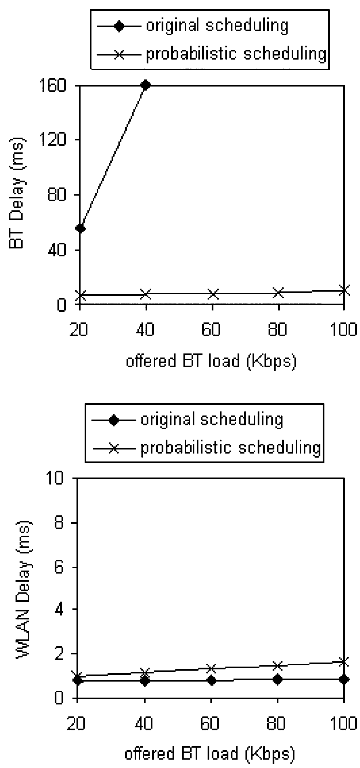

Fig. 9. Comparison of packet delay (Scenario 2).

BT traffic load, whereas the new probabilistic scheduling mechanism can meet the throughput and delay requirements of both BT and WLAN devices over a much wider range of traffic loads.

Figs. 7 and 8 show that the BT throughput is much improved by the probabilistic scheduling, while WLAN throughput is not affected. As in [19], we define the utility of the coexistence system to be the product of BT and WLAN throughputs. Evidently the new scheduling mechanism gives a much higher system utility. Fig. 9 shows that when the offered BT traffic load $\geq$ 40 kb/s, the original scheduling scheme results in delays that are much higher than 100 ms, which are unacceptable for most applications. If we simply reduce the BT buffer size to reduce delay, the packet drop ratio caused by buffer overflow will be high. In contrast, with the proposed scheduling scheme, the BT packet delays are very small and no BT buffer overflow is observed even when the offered BT traffic load is increased to 100 kb/s. Although WLAN delay is slightly increased by the probabilistic scheduling compared with the existing scheduling method, the delay values ($\ll$ 10 ms) are still much lower than those required by most real-time applications, e.g., voice over Internet protocol.

The existing scheduling mechanism in BT devices has a negative impact on BT throughput and delay and is unfair to the BT devices. In contrast, the proposed probabilistic scheduling scheme solves the problem in a self-adaptive and flexible manner so that not only WLAN but also BT systems can benefit from the coexistence mechanism.

V. Conclusion

In this paper, we have presented two new techniques, channel clustering and probabilistic channel visiting, to improve the existing noncollaborative BT/WLAN coexistence mechanisms. Channel clustering does not depend on a predefined PER threshold and can complete channel classification over a much shorter period of time with a higher accuracy, thus reducing interference between systems. Probabilistic scheduling is self-adaptive and can satisfy the QoS requirements of BT without significantly degrading WLAN performance. Simulations show that the new channel classification method is effective and the new probabilistic scheduling scheme can overcome the unfairness problem in the existing scheduling mechanism.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE Standard 802.11b-1999, Sep. 1999.

[2] *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)*, IEEE Standard 802.15.1, Jun. 2002.

[3] *Coexistence of Wireless Personal Area Networks With Other Wireless Devices Operating in Unlicensed Frequency Bands*, IEEE Standard 802.15.2, Aug. 2003.

[4] N. Golmie, N. Chevrollier, and O. Rebala, "Bluetooth and WLAN coexistence: Challenges and solutions," *IEEE Wireless Commun. Mag.*, vol. 10, no. 6, pp. 22–29, Dec. 2003

[5] N. Golmie, "Performance evaluation of a Bluetooth channel estimation algorithm," in *Proc. IEEE PIMRC 2002*, pp. 156–160.

[6] N. Golmie, "Bluetooth dynamic scheduling and interference mitigation," *Mobile Netw. Appl.*, vol. 9, no. 1, pp. 21–31, Feb. 2004.

[7] M.C.-H. Chek and Y.-K. Kwok, "On adaptive frequency hopping to combat coexistence interference between Bluetooth and IEEE 802.11b with practical resource constraints," in *Proc. ISPAN 2004*, pp. 1–6.

[8] C. F. Chiasserini and R. R. Rao, "Coexistence mechanisms for interference mitigation in the 2.4-GHz ISM band," *IEEE Trans. Wireless Commun.*, vol. 2, no. 5, pp. 964–975, Sep. 2003.

[9] Q. Pang and V. C. M. Leung, "Improved channel classification and scheduling for non-collaborative Bluetooth/ WLAN coexistence," in *Proc. IEEE VTC 2006—Spring*, May 2006, pp. 1303–1307.

[10] K. Marquess, *Physical Model Sub-Group Discussion and Questions,* IEEE 802.15/138R0, Nov. 1999.

[11] M. A. Marsan, C. F. Chiasserini, and A. Nucci, "Forming optimal topologies for Bluetooth-based wireless personal area networks," *IEEE Trans. Wireless Commun.*, vol. 5, no. 4, pp. 763–773, Apr. 2006.

[12] Z. Jiang, Q. Pang, V. C. M. Leung, and V. Wong, "High capacity Bluetooth access point design for interference elimination," in *Wireless LANs and Bluetooth*, Y. Xiao and Y. Pan, Eds. Commack, NY: Nova Science, 2005, ch. 13, pp. 271–294.

[13] A. El-Hoiydi, "Interference between Bluetooth networks—Upper bound on the packet error rate," *IEEE Commun. Lett.*, vol. 5, no. 6, pp. 245–247, Jun. 2001.

[14] T. Y. Lin and Y. C. Tseng, "Collision analysis for a multi-Bluetooth picocells environment," *IEEE Commun. Lett.*, vol. 7, no. 10, pp. 475–477, Oct. 2003.

[15] G. Pasolini, "Analytical investigation on the coexistence of Bluetooth piconets," *IEEE Commun. Lett.*, vol. 8, no. 3, pp. 144–146, Mar. 2004.
[16] R. Jain, *The Art of Computer Systems Performance Analysis—Techniques for Experimental Design, Measurement, Simulation, and Modeling*. New York: Wiley, 1991.
[17] A. R. Webb, *Statistical Pattern Recognition*. New York: Wiley, 2002.
[18] SuiteTooth: Bluetooth simulation model suite for OPNET, Highland Systems, Durango, CO, Nov. 2004.
[19] F. Kelly, "Charging and rate control for elastic traffic," *Eur. Trans. Telecommun.*, vol. 8, pp. 237–252, Jan. 1997.

Qixiang Pang received the B.A.Sc. (Hons.) degree from Tianjin University, Tianjin, China, in 1993, and the Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 1998.

He is currently a Research Engineer in the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada. Prior to this, he has been a Senior Systems Engineer and Senior Software Engineer in the Architecture and System Engineering Group, Motorola Canada, Vancouver, BC, Canada, where he was engaged in development of 2.5G/3G wireless systems. He has also been a Research Fellow at the Chinese University of Hong Kong, Shatin, NT, Hong Kong, SAR, and at the University of British Columbia. His current research interests include various topics of wireless networking, such as, wireless personal area network (WPAN), wireless local area network (WLAN), wireless metropolitan area network (WMAN), 3G/4G networks, sensor networks, performance and capacity improvement, protocol design, QoS, etc.

Mr. Pang has served as a Technical Committee Member for several IEEE/IEE international conferences.

Victor C. M. Leung (S'75–M'79–SM'97–F'03) received the B.A.Sc. (Hons.) and Ph.D. degrees from The University of British Columbia (UBC), Vancouver, BC, Canada, in 1977 and 1981, respectively, both in electrical engineering.

From 1981 to 1987, he was a Senior Member of Technical Staff at MPR Teltech Ltd., Burnaby, BC, where he was engaged in research on satellite communication systems. In 1988, he was a Lecturer in the Department of Electronics, Chinese University of Hong Kong, Shatin, NT, Hong Kong, SAR. He rejoined UBC as a Faculty Member in 1989, where he is currently a Professor and the TELUS Mobility Research Chair in Advanced Telecommunications Engineering in the Department of Electrical and Computer Engineering, and a member of the Institute for Computing, Information, and Cognitive Systems. His current research interests include the areas of architectural and protocol design and performance analysis for computer and telecommunication networks, with applications in satellite, mobile, personal communications, and high-speed networks. He is an Editor of the *International Journal of Sensor Networks*.

Prof. Leung is a voting member of the Association for Computing Machinery (ACM). He is currently an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He has served on the Technical Program Committee (TPC) of numerous conferences, as the TPC-Vice Chair of IEEE WCNC 2005, General Co-chair of ACM/IEEE MSWiM 2005, and General Chair of QShine 2007.