Improved Channel Classification and Scheduling for Non-collaborative Bluetooth/WLAN Coexistence

Qixiang Pang and Victor C.M. Leung

Department of Electrical and Computer Engineering
The University of British Columbia
Vancouver, BC, Canada

Abstract—**In this paper, we propose two new techniques to enhance the existing Bluetooth and WLAN coexistence mechanisms. The first one is channel clustering, which is used to classify the channel status more accurately. The second one is called probabilistic channel visiting, which is used to more reasonably allocate the channel resources between WLAN and BT devices. Both techniques are beneficial in improving the performance of the existing coexistence mechanisms. The effectiveness of these techniques is quantified by simulations.**

Keywords – Bluetooth; coexistence; non-collaborative; channel clustering; channel visiting; WLAN

I. Introduction

Two license-free wireless communication systems, IEEE 802.11 wireless local area networks (WLANs) [1] and Bluetooth (BT) wireless personal area networks (WPANs) [2], have been widely deployed in recent years. Because both systems operate in the 2.4 GHz instrumentation, scientific and medical (ISM) frequency band, mutual interference between them may result in severe performance degradations.

Recently, the interference issue has attracted many research and standardization activities and several coexistence mechanisms have been proposed [2]-[6]. For example, the IEEE 802.15.2 standard [2] includes eight coexistence mechanisms in total, which can be classified into two categories: collaborative and non-collaborative. In practice, most WLAN and WPAN devices are independently implemented and installed; therefore, non-collaborative mechanisms are more practical. In the following discussions, we shall concentrate on non-collaborative mechanisms.

The non-collaborative mechanisms ranging from adaptive frequency-hopping (AFH) to packet scheduling all use similar techniques to detect the presence of other devices in the ISM band and to classify the channels. For example, each BT device can maintain a packet error rate (PER) measurement per frequency used. The BT device infers which frequencies are occupied by other users of the ISM band by comparing the measured PERs with a predefined threshold [2][3].

The existing channel classification methods are simple, but they are effective only when a single Bluetooth piconet exists in the area of interest. When there are multiple piconets in the same area, these methods can be invalid. To address this issue, carrier sensing could be an effective option. However, sensing the carrier of a different communication system in a Bluetooth device adversely affects implementation complexity and cost.

This work was sponsored by the Canadian Natural Sciences and Engineering Research Council through grant STPGP 257684-02 and the OPNET University program.

0-7803-9392-9/06/$20.00 (c) 2006 IEEE

Besides channel classification, a related problem is how often to visit the channels after they have been correctly classified. In the existing mechanisms, the so-called "bad" channels are not visited at all. However, "bad" or "good" is a fuzzy concept. When there are multiple WLANs or multiple BT piconets, the number of "good" channels can be very low and the BT devices may not be able to obtain enough radio resources due to their excessive courtesy.

This paper addresses the respective problems discussed above by proposing two new techniques called channel clustering and probabilistic channel visiting. The existing coexistence mechanisms, e.g., AFH and scheduling, can then be enhanced to perform better when multiple WLANs or multiple BT piconets coexist.

The rest of this paper is organized as follows. In Section II, the problems with the existing non-collaborative mechanisms and channel classification methods are discussed in more detail. Section III describes the two new techniques proposed in this paper: channel clustering and probabilistic channel visiting. Section IV gives the performance comparison results. Section V concludes this paper.

II. Problems in Existing Coexistence Mechanisms

A. Problem with Channel Classification

Channel classification is the first step for all non-collaborative coexistence mechanisms to work.

BT uses 79 or 49 frequency channels, each about 1 MHz wide. The purpose of channel classification is to determine the quality of each channel. The major concern of the quality is interference from other coexisting systems (in this paper, we consider WLANs and BT piconets).

Several channel classification methods are recommended by the IEEE 802.15.2 standard [2] and related literature: PER, received signal strength indication (RSSI), and carrier sensing.

Among the three methods, carrier sensing is the most robust. However, implementing in a cheap BT device the ability to sense the carrier of another system employing a totally different signaling method is a very difficult task. The other two methods (measuring PER and RSSI) are simpler to implement and have been the channel classification methods considered in most papers. However, the existing methods use predefined PER or RSSI thresholds that can be invalid when multiple BT piconets coexist, as in the case described in [7], since a high PER or RSSI can be caused not only by a WLAN, but by other BT devices in adjacent piconets as well.

Consider N collocated and unsynchronized BT piconets independently hopping over m channels. Assume that all packets occupy a single slot and the probability that a packet is transmitted in a slot is G. Let t_d be the single slot packet duration and t_s be the duration of each slot. Define $r = 2(1 - t_d / t_s)$. The mean collision probability, p_{bt}, for N piconets is given by [8]

$$p_{bt} = 1 - \left(r(1 - G/m) + (1 - r)(1 - G/m)^2\right)^{N-1}. \qquad (1)$$

Figure 1 shows the collision probabilities given by (1) when $G = 1$ and $m = 79$. When the number of piconets is large, the collision probability can be quite high. On the other hand, the traffic from the WLAN devices is highly variable and the collision probabilities with WLAN devices can be either high or low. Therefore it is inaccurate to label a channel as one experiencing "bad" interference from WLANs, simply according to its PER/RSSI without considering the existence of other piconets.

More detailed analysis of Bluetooth systems can be found in [9]. The simple analysis given above, however, is sufficient to show that a high PER can be caused by multiple BT piconets and using a predefined PER threshold to classify channels can be erroneous. When we use the measured PERs to classify channels, both BT and WLAN traffic loads should be considered and the threshold should be automatically adjusted to adapt to changes of the network environments. However, in the existing literature, a fixed threshold value is used, which is predefined and can only be changed manually after it has been set. Different papers considered different threshold values without a common selection rule; e.g., in [4], the PER threshold is 0.5 while in [3] the BER threshold is 0.001 (equivalent to a PER of about 0.3 if packet size is 366 bits). In [5] and [6], a channel is considered "bad" if a single packet error has occurred.

In addition to the observation that the PER should be dynamically adjusted, we also note that the bandwidth of an IEEE 802.11b channel is 22 MHz and therefore a WLAN transmission may interfere with about 22 BT channels. The existing channel classification methods ignore this available information and therefore are not very efficient. This property tells us that the "bad" channels in general span a range of consecutive frequencies. Also, according to [1], a WLAN channel is centered at one of 11 fixed frequencies (e.g., 2412 MHz, 2437 MHz, 2462 MHz, …). All these features can help us in locating the "bad" channels more accurately.

B. Problem with Channel Visiting

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

In the existing coexistence mechanisms, once a channel has been marked as "bad", it will not be visited at all. For example, in the existing scheduling mechanism [2][5], 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. In AFH,

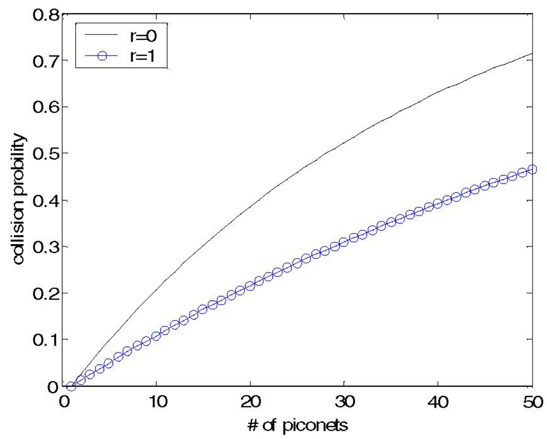

Figure 1. Collision probability between multiple piconets

only the "good" channels are included in the hop set [2][4][1]. This rigid channel visiting scheme may cause some problems. If the BT traffic load is heavy or the number of BT piconets is large, scheduling may cause the BT packets to experience unacceptable backlogs, and AFH may cause the BT packets to experience lots of collisions with other BT transmissions in other piconets as given by (1).

In this paper we mainly investigate the scheduling mechanism. The advantages of scheduling over other non-collaborative mechanisms are that: 1) the algorithm can be implemented entirely in the master node; 2) the method does not require changes to the BT physical specification. In the following we derive some analytical results for the scheduling mechanism.

Assume there are multiple unsynchronized BT piconets that are subject to interference from WLANs. These BT piconets follow the same assumptions on packet length and duration as stated in Section II-A. Assume there are m_g "good" channels and $m_b = m - m_g$ "bad" channels. Assume the probability is a constant p_w that a BT packet transmission in a "bad" channel collides with a WLAN packet.

When the scheduling mechanism is not applied, the average probability that a BT packet transmission is unsuccessful is then given by

$$p_u = p_{bt}\frac{m_g}{m} + (p_{bt} + p_w - p_{bt} \times p_w)\frac{m_b}{m}. \qquad (2)$$

where p_{bt}, the probability of a BT packet transmission colliding with other BT packets in a channel, is given by (1).

When scheduling [2][4] is applied, the probability that a BT packet transmission is unsuccessful (collided or delayed), p_u, becomes

$$p_u = p_{bt}\left(m_g/m\right)^2 + 1 - \left(m_g/m\right)^2. \qquad (3)$$

Define the normalized throughput S of a BT piconet as the fraction of packets transmitted successfully, and it is given by

$$S = G(1 - p_u). \qquad (4)$$

[1] If the number of "good" channels is less than 15, some "bad" channels are included in the hop set to increase its size to 15 to satisfy FCC part 15 [10].

The total normalized throughput of the N piconets is given by

$$S_t = NG(1 - p_u) . \quad (5)$$

Assuming $m = 79$, $G = 1$, $p_w = 0.6$, and $r = 2(1 - 366/625) = 0.8288$, Figure 2 gives the throughput per piconet with and without scheduling.

These numerical results show that using the original scheduling method, avoiding WLAN interference may sacrifice the BT throughput when BT piconets have a high throughput requirement and the WLAN does not occupy its channel 100% of the time. When there are multiple WLANs, or when the BT systems use only 49 channels, the problem can be even worse.

The above analysis uses a simplified BT model. We shall examine more realistic cases by simulations in Section VI.

The problem experienced by the scheduling mechanisms may exist in other non-collaborative mechanisms as well, e.g., AFH. With a reduced hop set in AFH and a number of BT piconets, the collision probabilities between these piconets can become very high and the throughput can be significantly reduced.

The discussions above motivate our proposal of new channel classification and channel visiting techniques.

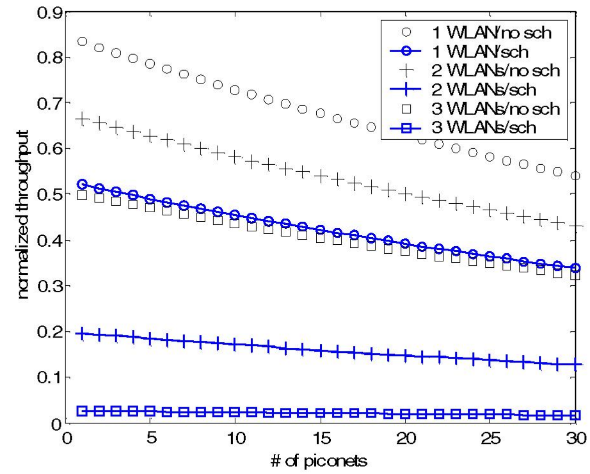

Figure 2. Throughput comparison between w/o and with scheduling when there are 1, 2 and 3 WLANs

III. Channel Clustering and Probabilistic Channel Visiting

A. Channel Clustering and Classification

Consider that each WLAN channel interferes with a cluster of 22 BT channels. When all the BT channels have been visited a sufficiently large number of times by a BT piconet, PERs can be collected and used to classify the channels.

We observe that the channel clustering and classification problem has some similarity to other well-studied topics, e.g., segmentation or edge detection in pattern recognition [11]. 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. It is possible to cluster channels more accurately using more sophisticated signal processing techniques; however, considering that BT devices must be implemented at very low cost and may have low processing capability, in this paper we use the simple procedure described below.

Let $p_{e,i}$ be the measured PER of the i^{th} channel and m be the number of channels. The average PER over all channels is $\theta_1 = \sum_{i=0}^{m-1} p_{e,i} \Big/ m$. Each channel, i, from Channel 0 to m-1, is checked. If $p_{e,i} > \theta_1$, it is possible that Channel i is the starting edge of a bad channel cluster and the cluster spans from i to i+W-1. For 802.11b WLANs, W = 22. However, due to fluctuations of PERs, it is also possible that Channel i is a good channel. So we must obtain more information before making the decision. We let $\theta_{2,i} = \left(\sum_{j=1}^{K} p_{e,i-j} + \sum_{j=0}^{K-1} p_{e,i+W+j} \right) \Big/ 2K$. $\theta_{2,i}$ is the average PER of the channels adjacent to the two edges of the potential "bad" channel cluster starting from Channel i. K is the width of the potential edge area. If most of the channels from Channel i to Channel i+W-1 (e.g., more than 80% of the 22 channels) collectively experience PERs higher than $\theta_{2,i}$, very likely Channel i to i+W-1 is a "bad" channel cluster.

If Channel i to i+W-1 is not a "bad" cluster, the probability of misunderstanding it to be a bad cluster is very low. In this case, $\theta_{2,i}$ is close to the average PER of the channels not being occupied by a WLAN. For each channel not being occupied by a WLAN, the probability that its PER is higher than the average PER is only 1/2. Therefore, for example, among 22 channels, the probability that the PERs of more than $80\% \times 22 = 17.6$ channels are higher than the average PER (the 1st channel must be higher) is only $\frac{1}{2} \times \sum_{i=17}^{21} \binom{21}{i} \Big/ 2^{21} = 0.2\%$.

Once the clusters have been detected, we can then utilize other known information of WLAN channels to refine them. According to [1], WLAN channels must be located at some certain frequencies and between two WLAN channels there must be a "guard" band. Therefore in the refinement procedure, the detected cluster edges are compared with these positions and adjusted to the nearest ones accordingly.

B. Probabilistic Channel Visiting

We have shown in Section II-B that when the existing coexistence mechanisms, such as scheduling and AFH, are enabled, the performance of BT systems can be degraded. In order to seek a balance between interference mitigation between different wireless technologies and within the same wireless technology, we introduce a new channel visiting technique - probabilistic channel visiting. A channel marked as "bad" can be visited with a probability (p_v). This visiting probability can be dynamically adjusted according to the variation of the BT traffic load and the environmental conditions. The original scheduling coexistence mechanism can be modified to be a probabilistic scheduling mechanism. 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 WLAN is less than 1, a higher p_v can lead to a higher BT throughput in the probabilistic scheduling.

It can be seen that the original scheduling mechanism is a special case of the probabilistic scheduling mechanism, i.e., the visiting probability is always 0. When the visiting probability is always 1, it is equivalent to the case when the coexistence mechanism is disabled.

The p_v adjustment mechanism can be built upon different quality of service and fairness objectives. Considering the low processing ability of BT devices and the non-collaborative nature, in this paper we propose the simple buffer content based probabilistic channel visiting scheme described below.

The p_v value is adjusted periodically according to the measured buffer depth, $\hat{B}$. If $\hat{B}$ is bigger than a threshold B^*, p_v is increased with a value Δ. Otherwise, it is reduced by the same amount. In implementation, 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 this scheme, the top priority for a BT device to consider is to satisfy its own throughput and delay requirements. In reality, this consideration is reasonable because the BT and WLAN devices likely belong to different users. As the coexistence mechanism is implemented in the BT devices only, it is unreasonable to allow such 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 the WLAN can be reduced.

C. Other Improvements to the Existing Coexistence Mechanisms

The common procedure of the new non-collaborative coexistence mechanisms based on channel clustering and probabilistic channel visiting is shown in Figure 3. There are two types of channel visiting phases: all channel visiting and selective channel visiting. The all channel visiting phase is used for channel classification and the selective one for interference avoidance.

Different from the existing timer based method [4][5], we use a counter based method to delimit the phases, because timers generally cost more resources to implement, and the timer based method is dependant on incoming BT traffic density.

Three types of counters (C1, C2, and C3) are used to record the numbers of packet transmissions. When the counters are accumulated to their respective upper limits, corresponding actions are taken.

C1: when C1 reaches its limit, an all channel visiting phase starts.

C2: when C2 reaches its limit, a p_v adjustment is performed.

C3[i]: is used to record the number of packet transmissions for Channel i ($0 \le i < m$); when all C3 counters reach their limit, the all channel visiting phase finishes and a channel classification is performed.

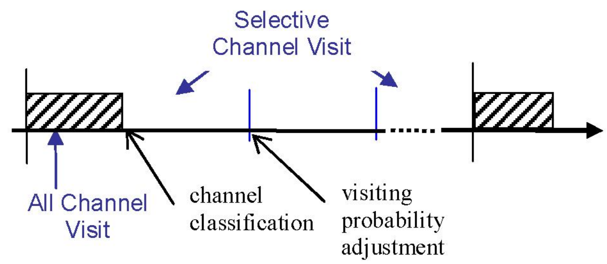

Figure 3. Common procedure of the new non-collaborative coexistence mechanisms

We also note that when a channel is extremely bad (say PER > 0.9), the payback for a BT device to use this channel is very limited. In this case we set the visiting probability to 0.

IV. Performance Evaluation by Simulations

In this section we demonstrate the effectiveness of the new channel classification scheme and the performance improvement of the proposed probabilistic scheduling method.

The simulations were based on OPNET [12]. As there is no BT model in the standard OPNET library, the SuiteTooth BT model from Highland Systems [13] was used. On the basis of OPNET and SuiteTooth, the coexistence mechanisms were implemented and evaluated.

The simulated network topology is explained as follows. Each WLAN AP and its stations are separated by 10m. The WLAN APs are located at (0, 0) and the stations are at (10, 0). The BT piconets are randomly placed in a rectangular area bounded by (0,-2), (0,2), (10,-2) and (10,2). The distance between a BT master and its slave is 0.5m. A WLAN station sends data packets to the AP. The physical layer data rate in WLAN is 11 Mbps. The packet size follows the NIST Internet packet size distribution [13] given in TABLE I. The inter-arrival time is exponentially distributed. The data traffic between BT master and slave is bidirectional. The higher layer BT packets are all 100 bytes with Poisson arrivals. At the BT baseband layer, DH1 packet format is used.

By taking into consideration of both measurement accuracy and control promptness, the values of the control parameters used in the proposed schemes are given heuristically as follows. The upper limits for the counters are respectively: 32000 for C1, 1600 for C2, 20 for C3. The Δ value for p_v adjustment is 0.15. Buffer content threshold B^* is 1000 bytes. The K value for channel clustering is 5.

Our first objective is to examine the effectiveness of the proposed channel classification method. Define the channel identification ratio (idr) as

$$idr = \frac{\sum_{i=1}^{N}\sum_{j=0}^{1}\sum_{k=0}^{m-1} b_{i,j,k}}{N \times 2 \times m}. \qquad (6)$$

where

$$b_{i,j,k} = \begin{cases} 1, & \text{if the } k\text{th channel in direction } j \text{ of the} \\ & i\text{th piconet is correctly identified} \\ 0, & \text{if the } k\text{th channel in direction } j \text{ of the} \\ & i\text{th piconet is wronly identified} \end{cases} \quad (7)$$

Two WLANs occupying 44 BT channels were simulated. The mean packet inter-arrival time over each WLAN is 0.001 sec. The mean packet inter-arrival time at the BT higher layer is 0.01 sec. For each simulation, a new random seed is used. 10 simulations were run for each scenario.

When there is one piconet, the average *idr* before refinement is 99.6%. When there are 10 piconets, the *idr* value before refinement is 98.5%. After refinement it is 99.4%, which is only slightly worse than the one piconet case. These values show that the channel clustering/classification technique is effective even when there are multiple piconets.

Our second objective is to evaluate the performance of probabilistic scheduling. Since the performance benefits of using scheduling under light BT load condition have been intensively studied in [2][4][5], here we mainly examine performance under a heavy BT load condition.

We consider the case where there is only one WLAN. The WLAN mean packet inter-arrival time is 0.002 sec. The BT mean packet inter-arrival time is varied to impose different BT traffic load. In scenario 1, a single piconet was simulated and in scenario 2, 5 piconets were simulated. Figure 4 and Figure 5 present the simulation results for the two scenarios, respectively, where "sch" represents the scheduling method in [2][4][5], and "p-sch" represents our proposed probabilistic scheduling method. It can be seen that when the offered load is heavy, the original scheduling method fails to satisfy the BT throughput requirement (given by the offered BT traffic load), while the new scheduling method avoids this drawback. With the new scheduling method, the WLAN throughput is reduced only slightly but the BT throughput is substantially increased.

V. Conclusion

In this paper, we have presented two new techniques, channel clustering and probabilistic channel visiting, to improve the existing non-collaborative BT/WLAN coexistence mechanisms. Simulation results show that the new channel classification is effective and the new probabilistic scheduling method can overcome the low BT throughput problem in the existing scheduling mechanism with only a slight degradation to the WLAN throughout.

References

[1] IEEE Std 802.11b-1999, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-speed Physical Layer Extension In The 2.4 GHz Band," Sep. 1999.

[2] IEEE Std 802.15.2, "Coexistence of wireless personal area networks with other wireless devices operating in unlicensed frequency bands," Aug. 2003.

[3] N. Golmie, "Performance evaluation of a Bluetooth channel estimation algorithm," *IEEE PIMRC 2002*, pp.156-160, 2002.

[4] N. Golmie, N. Chevrollier, and O. Rebala, "Bluetooth and WLAN Coexistence: Challenges and Solutions," *IEEE Wireless Communications Magazine*, pp. 22-29, December 2003.

[5] N. Golmie, "Bluetooth dynamic scheduling and interference mitigation," *Mobile Networks and Applications*, vol. 9, pp. 21-31, Feb. 2004.

[6] 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," *Proc. 7th International Symposium on Parallel Architectures, Algorithms and Networks*, pp.1-6, 2004.

[7] Z. Jiang, Q. Pang, V.C.M. Leung, and V. Wong, "High Capacity Bluetooth Access Point Design for Interference Elimination," *Wireless LANs and Bluetooth,* Nova Science Publishers, 2005.

[8] A. El-Hoiydi, "Interference between Bluetooth Networks - Upper Bound on the Packet Error Rate," *IEEE Communications Letters,* vol. 5, pp. 245-247, June 2001.

[9] G. Pasolini, "Analytical Investigation on the Coexistence of Bluetooth Piconets," *IEEE Communications Letters*, vol. 8, no. 3, pp. 144-146, March 2004.

[10] FCC Parts 15 Regulations, http://www.fcc.gov/oet/info/rules/part15.

[11] A.R. Webb, *Statistical Pattern Recognition*, Wiley, 2002.

[12] OPNET, http://www.opnet.com.

[13] Highland Systems, "SuiteTooth: Bluetooth simulation model suite for OPNET," Nov. 2004.

TABLE I. NIST Packet Size Distribution

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

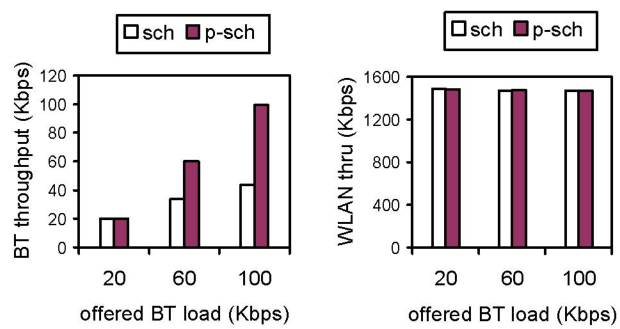

Figure 4. BT throughput per piconet and WLAN throughput comparison for scenario 1

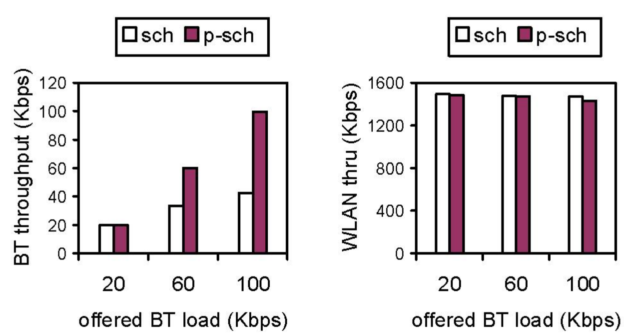

Figure 5. BT throughput per piconet and WLAN throughput comparison for scenario 2