Mobile Netw Appl (2007) 12:450–459
DOI 10.1007/s11036-008-0047-3

Coexistence of IEEE 802.11b and Bluetooth: An Integrated Performance Analysis

Min Song · Sachin Shetty · Deepthi Gopalpet

Published online: 26 April 2008

Abstract IEEE 802.11b wireless networks and Bluetooth networks provide complimentary services using the same unlicensed radio frequency band. As the benefits of utilizing these services become increasingly apparent, the likelihood of mutual interference also increases. The well-known frequency hopping algorithm and adaptive frequency hopping algorithm do not fully consider the interference level of the operating environment. In this paper an algorithm called interference-aware adaptive frequency hopping (IAFH) is presented and implemented on Bluetooth devices to mitigate the interference between IEEE 802.11b and Bluetooth wireless networks. An analytical model of IAFH is developed to evaluate the performance of 802.11b stations and Bluetooth devices in a mutual interference environment. The analysis comprises the collision probability, packet error rate, and throughput performance for both IEEE 802.11b and Bluetooth wireless networks. Simulation results confirm that 802.11b station and IAFH-enabled Bluetooth devices experience lower packet error rates and better throughput as compared to the frequency hopping and adaptive frequency hopping algorithms.

Keywords 802.11b · Bluetooth · frequency hopping · coexistence · interference

M. Song (✉) · S. Shetty · D. Gopalpet
Department of Electrical and Computer Engineering,
Old Dominion University,
Norfolk, VA 23529, USA
e-mail: msong@odu.edu

S. Shetty
e-mail: sshetty@odu.edu

D. Gopalpet
e-mail: deepthi.g7@gmail.com

1 Introduction

There has been an increasing proliferation of wireless communication networks sharing the unlicensed industrial, scientific, and medical (ISM) band. This has led to an investigation of mutual interference to determine the viability of these networks co-existing with each other. Two of the most important wireless networks designed to operate in the 2.4 GHz ISM band are IEEE 802.11b and Bluetooth.

An 802.11b network typically covers a small-sized area, up to 100 meters in diameter. The 802.11b standard uses a Direct Sequence Spread Spectrum (DSSS) signaling method and a complementary code keying (CCK) modulation scheme to obtain a maximum data rate of 11 Mbps [13]. The available bandwidth is divided into channels for the operation of DSSS. Each channel occupies approximately 20 MHz. There are 11 channels identified for DSSS system in the USA. Among them, there are only three non-overlapping channels.

Bluetooth uses frequency hopping spread spectrum (FHSS) to transmit radio signals. Bluetooth devices form a network, which is also termed as a piconet [4], to allow one *master* device to interconnect with up to seven active *slave* devices. A piconet typically has a range of about 10 m and a maximum data rate up to 1 Mbps. In Bluetooth, the 2.4 GHz band is segmented into 79 channels, each 1 MHz wide. Channels use a frequency-hop/time-division-duplex scheme. Each channel is divided into 625 μs interval, called slot, where a different hop frequency is used for each slot. The master transmission starts in even-numbered slots, while the slave transmission starts in odd-numbered slots. However, not the whole slot is used for packet transmission. To allow the transmitter and receiver devices to change from receiver to transmitter mode and ensure that the frequency synthesizer tune to the next channel frequency, a 259 μs turn around time is left at the end of the last slot [6]. Packets can be one, three and five slots long, namely DH1, DH3 and DH5 and are transmitted in consecutive slots [5]. Multiple piconets with overlapping coverage can co-exist provided their frequency hopping patterns are mutually orthogonal.

Performance degradation has been observed when IEEE 802.11b and Bluetooth operate in close vicinity [8, 9]. To mitigate the problem of interference between 2.4 GHz ISM band devices, the Bluetooth Special Interest Group (SIG) introduced adaptive frequency hopping (AFH) as a possible solution [14]. Further enhancements have been proposed to improve the performance of AFH-enabled Bluetooth devices [5, 10, 12]. In this paper, an interference-aware adaptive frequency hopping (IAFH) algorithm is presented. A mathematical model for the interference of 802.11b and IAFH-enabled Bluetooth devices is developed. The co-existence model is developed to model the performance impact on 802.11b and Bluetooth devices while operating in a mutual interference environment. The integrated analysis provides details on the following performance metrics: the collision probability, the packet error rate, and throughput.

The rest of the paper is organized as follows. Section 2 introduces the related work. Section 3 presents the new IAFH algorithm and the analytical models. Section 4 provides the simulations and analysis. Finally, the paper is concluded in Section 5.

2 Related work

Numerous attempts at quantifying the impact of interference on both 802.11b and Bluetooth performance have been made. Analytical results on the probability of collision were obtained by Shellhammer [23], Ennis [16], and Zyren [24]. Analytical study on 802.11b packet error rate was conducted by Conti [1]. The Bluetooth packet error rate is analytically studied in [18]. In all the above efforts, the probability of packet error is computed based on the probability of packet collision in both time and frequency domains. These analytical results give a first order approximation on the impact of interference and the resulting performance degradation. However, the analytical results ignored mutual interference that can change the traffic distribution for each system.

Experimental results have been gathered by Kamerman [21], Howitt et al. [20], and Fumolari [17] for a two-node 802.11b system and a two-node Bluetooth piconet. These results are more accurate at the cost of being too specific to the implementation testbed. Thus, a third alternative consisting of using modeling and simulation to evaluate the impact of interference gained prominence. Zurbes et al. [25] present simulation results for a number of Bluetooth devices located in a single large room. Results show that 100 concurrent web sessions degrade the performance by only 5%. Golmie et al. [19] use a detailed MAC and PHY simulation framework to evaluate the impact of interference for a pair of 802.11b nodes and a pair of Bluetooth nodes. Similar results have been obtained by Lansford et al. [22] for the case of coexisting 802.11b and Bluetooth devices on the same laptop. Their simulation models are based on a link budget analysis and a theoretical calculation of the BER.

Recently two interference avoidance solutions have gained greater acceptance by the IEEE 802.15.2 Coexistence Task Group and the Bluetooth Special Interest Group. The first one is adaptive frequency hopping (AFH) mechanism targeted at modifying the Bluetooth frequency hopping sequence in the presence of 802.11b stations [2]. The second one is a Bluetooth interference aware scheduling (BIAS) strategy that postpones the transmission of packets on so-called *bad* frequencies [3, 7].

In AFH, the first step is to determine the frequency carrier for each packet, which is determined, based on the hop frequency selection scheme. The hop frequency scheme works as follows. Initially a pseudo random sequence is generated based on 27 bits of the master's clock value and the 28 bits of the Bluetooth masters address. Then the sequence is mapped with the desired hopping frequency. AFH uses pseudo random frequency hopping and channel conditions to change the hopping frequency dynamically thereby minimizing interference and enabling coexistence. By implementing this mechanism, the packet transmission is not delayed, but the throughput may decrease and some bad channels are used [5]. With changing channel conditions, the channels are periodically reevaluated and new adapted hop sequences are generated. The basic frequency hopping sequence has 79 channels while the adapted sequence may have minimum of 20 channels [5]. However, AFH would be ideal only in a slow changing environment where the same hopping sequence could be used for a long period of time.

The BIAS strategy allows the master device to avoid data transmission to a slave experiencing a *bad* frequency. Furthermore, since a slave transmission always follows a master transmission, using the same principle, the master avoids receiving data on a bad frequency, by avoiding a transmission on a frequency preceding a bad one in the hopping pattern. The BIAS strategy is illustrated in Fig. 1. In the figure, a packet is scheduled to be transmitted in the current Bluetooth slot. The slot is occupied for transmission for the first 366 μs. In the remaining 259 μs, the Bluetooth devices sense the channel for a good frequency to be used in the next slot. By referring to the frequency status table, the master transmits in a slot after it verifies that both the slave's receiving frequency, f_s, and its own receiving frequency, f_m, are good. Otherwise, the master skips the

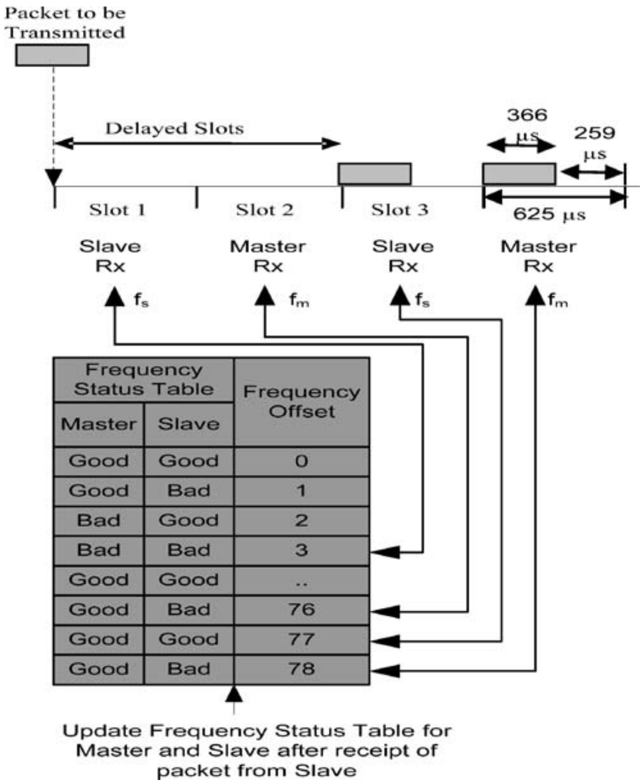

Figure 1 Bluetooth interference aware scheduling

current transmission slot and repeats the procedure over again in the next transmission opportunity. By implementing the delay policy, the channel efficiency and system throughput can be maximized considerably at the expense of delay in the packet reception. BIAS is suitable when the hopping pattern of the Bluetooth changes rapidly. This happens when interference levels change frequently.

3 Interference-aware adaptive frequency hopping (IAFH) algorithm

In a co-existence network, interference levels in the environment can vary slowly or rapidly. IAFH addresses issues related to both environments. The main premise of the approach is to withdraw the packet transmission if the interference levels vary more rapidly. By withdrawing the transmission, the number of collisions is reduced drastically. If the interference levels persist for a longer duration, AFH scheme is used to transmit the data using the available good channels. The algorithm can be incorporated into Bluetooth devices without any modification to the current slot structure.

The algorithm works as follows. A small period of time from the idle 259 μs is chosen to keep sensing the channel for a good frequency slot. If the slot is bad, the master implements a delay policy and it withdraws the transmission. The transmission is postponed until a slot associated with a good frequency becomes available. The master device uses the channel state information to avoid using a bad frequency to transmit data to a slave. Since a slave transmission always follows a master transmission, the master avoids receiving data on a bad frequency by avoiding a transmission on a frequency preceding a bad one in the hopping pattern. The delay policy is implemented until a certain amount of time. If the interference persists for a longer period, the frequency hopping sequence is then modified to replace the bad frequencies using the AFH scheme. The performance is improved, as no bad channels will be used till a certain point of time and only when the delay becomes unacceptable, the bad channels are used to comply with the Bluetooth transmission rate.

Before presenting the pseudocode, we provide the notations as follows.

3.1 Notations

d	Delay experienced when a Bluetooth packet transmission is withdrawn
d_{accept}	Acceptable delay defined by users. This value has been set to 10 ms
S_f	Bluetooth transmission slot using frequency f
l	Length of packet from master to slave
N_g^f	Set of slaves with good frequency f.
N_b	Union of sets of bad frequencies collected from all Bluetooth devices.
$N_b^{\min}$	Number of bad frequencies that must be used to comply with the Bluetooth transmission rate
W	Hopping window segment containing list of good and bad frequencies

3.2 Pseudocode

For every even slot {

if $d = d_{accept}$ {

Choose frequency f
// Master tentatively transmits on frequency f
If $S_f + l$ is good {

// Master can receive in next slot
N_g^f={Check frequency status table and pick set of slaves such that f is good }
If (N_g^f != NULL) {

Select slave i according to a priority criteria
Transmit packet of size l to slave i
}
Else

// Slave cannot receive in next slot

Withhold transmission by increasing d
}
Else

// Master cannot receive in next slot
Withhold transmission by increasing d
}

Else {

// Use AFH to transmit data
W= 32
// Initialize the hopping algorithm window to
// 32 "good" and "bad" frequencies
W=W+N_b
// Increase W by the number of *bad*
// frequencies $\mathbf{N_b}$
if (W > 79) {

W = 79
N_b = *min* (N_b, 79 – $N_b^{\min}$)
//use at least $N_b^{\min}$ different frequencies
}

}

}

3.3 Interference analysis and analytical model

The interference analysis is centered on deriving a closed form solution for the probability of collision P_c in terms of radio and network parameters. A collision occurs when one or more interfering signals overlap in both time and frequency domains with the desired signal. Based on the probability of collision, the packet error rate can be calculated. As mentioned earlier, 802.11b consists of three non-overlapping channels with a bandwidth of 22 MHz each. Thus, three possible scenarios can be considered for interference analysis.

In the first scenario, only one channel is occupied by 802.11b stations and thus Bluetooth devices can operate other 57 good channels. This leads to negligible probability of collision with 802.11b stations.

In the second scenario, two channels are occupied by 802.11b stations. Bluetooth devices are left with 35 channels to transmit. As Bluetooth devices need a minimum of 20 channels to transmit, the possibility of collision with 802.11b stations is reduced to a negligible value.

In the third scenario, 802.11b stations transmit on all the three non-overlapping channels, i.e., only 13 channels are left for Bluetooth devices. The probability of collision worsens. The Bluetooth devices have to transmit on at least seven bad channels. The rest of the analysis focuses on this scenario.

3.4 802.11b analysis

The probability of collision experienced by 802.11b stations due to the presence of Bluetooth devices is given by the collision in both time and frequency.

$$P_c = P_c(\text{time overlap}).P_c(\text{frequency overlap}) \tag{1}$$

In IAFH, collision occurs when the Bluetooth packet transmission is withdrawn and when the Bluetooth packet is transmitted. When the Bluetooth packet is withdrawn, collision occurs between the Bluetooth sensing operation and the 802.11b packet transmission. Therefore, the packet withdrawal probability can be written as:

$$P_{c-w} = \frac{\text{802.11b frame transmission time} + \text{Bluetooth sense time}}{\text{Total Bluetooth time interval}} \cdot \frac{N_b}{N_b^{\min}} \tag{2}$$

Similarly, when a Bluetooth packet occupies a slot, it collides with an 802.11b packet. This probability of collision can be written as:

$$P_{c-o} = \frac{\text{802.11b frame transmission time} + \text{Bluetooth occupied time}}{\text{Total Bluetooth time interval}} \cdot \frac{N_b}{N_b^{\min}} \tag{3}$$

The bit error rate for CCK modulation at the 802.11b receiver in AWGN in multi path channel using Rayleigh model is given as the following approximation [11],

$$\mathrm{BER} = \frac{M}{2M-1}\left(\sum_{m=1}^{M-1}\frac{(-1)^{m+1}\cdot\binom{M-1}{m}}{1+m+m\,8\,\Gamma_1}\right) \tag{4}$$

where M is the number of bits in the symbol and

$$\Gamma_1 = \sqrt{\frac{2E_b}{N_0}}, \text{for } \mathrm{CCK}_{11\mathrm{Mbps}} \tag{5}$$

where E_b is the 802.11b signal power, and N_0 is the noise spectral density.

The packet error rate (PER) of an 802.11b station is calculated based on the collision of the Bluetooth sensing operation with the 802.11b packet transmission and the collision of Bluetooth packet with an 802.11b packet. To find *PER*, we need to first find the probability of a good 802.11b packet $P(G)$.

$$P(G) = P(G|C).P_c + P(G|\sim C).(1-P_c) \tag{6}$$

where C represents a collision event.

Let BER_o represent the bit error rate of an 802.11b packet when its transmission overlaps the Bluetooth transmission in the first 366 µs of a slot. This is the length of the Bluetooth slot when a Bluetooth packet is being transmitted in presence of AWGN. Let BER_s represent the bit error rate of an 802.11b packet when its transmission overlaps with the Bluetooth sensing time in the remaining 259 μs. In the remaining 259 μs, only AWGN is present. Finally, let BER_n represent the bit error rate of an 802.11b packet when there is no Bluetooth packet being transmitted in the entire 625 μs time interval. We have,

$$P(G|\sim C) = (1 - \mathrm{BER}_n) \tag{7}$$

$$P(G|C) = (1 - \mathrm{BER}_o) \cdot (1 - \mathrm{BER}_s) \tag{8}$$

By combining Eqs. 2–8, we get

$$\mathrm{PER}_{c-w} = 1 - [(1 - \mathrm{BER}_n).(1 - P_{c-w}) + (1 - \mathrm{BER}_o).(1 - \mathrm{BER}_s).P_{c-w})]^k \tag{9}$$

$$\mathrm{PER}_{c-o} = 1 - [(1 - \mathrm{BER}_n).(1 - P_{c-o}) + (1 - \mathrm{BER}_o).(1 - \mathrm{BER}_s).P_{c-o})]^k \tag{10}$$

where PER_{c-w} is the *PER* due to collision during the packet withdrawal period, PER_{c-o} is the *PER* during the collision of Bluetooth packet with 802.11b packet, BER_o, BER_s, BER_n are calculated from Eq. 4 by setting the appropriate values for E_b and N_0 in Eq. 5, k=(802.11b packet transmission time)/(625 μs). For BER_o, the noise spectral density N_0 will be calculated by computing the signal strength for the Bluetooth packet being transmitted and AWGN. For BER_s, the noise spectral density N_0 will be calculated by computing the signal strength for the Bluetooth sensing operation and AWGN. For BER_o, the noise spectral density N_0 will be influenced by AWGN.

For multiple piconets in the vicinity of an 802.11b system, the PER for IAFH can be written as [4]:

$$\mathrm{PER}\{N(\gamma)\} = 1 - (1 - \mathrm{PER})^{N(\gamma)} \tag{11}$$

where $N(\gamma)$ represents the number of piconets is given by

$$N(\gamma) = D_{BT}\left(\pi d_s^2\right) \exp\left[\frac{2\left(\sigma_{I/S}^2 - 10n\Gamma \log_{10}(e)\right)}{(10n \log_{10}(e))}\right] \tag{12}$$

where D_{BT} is the Bluetooth piconet density, d_s is the separation between 802.11 device and its access points, $\sigma_{I/S}$ is the interference to signal shadowing standard deviation, and n is the path loss exponent.

The throughput of the 802.11b station in the presence of N (γ piconets can be written as:

$$\mathrm{Th} = \frac{\mathrm{Data}_{80211b}}{T_{80211b}} (1 - \mathrm{PER})^{N(\gamma)} \tag{13}$$

Data_{80211b} is the number of 802.11b bytes transmitted and T_{80211b} is the total time taken by the packet to reach the receiver.

To compute the total throughput for IAFH, Eq. 9 is substituted in Eq. 13 to calculate the throughput when the packets are withdrawn and Eq. 10 is substituted in Eq. 13 to calculate the throughput after the delay exceeds d_{accept}. This will improve the performance of the 802.11b system as no bad channels will be used till a certain point of time and only when the delay becomes unacceptable, the bad channels are used.

3.5 Bluetooth analysis

We now look at the performance analysis of Bluetooth piconet in presence of multiple 802.11 stations. We will first compute the bit error probability for GFSK modulation at the Bluetooth receiver in the presence of AWGN. Let P_e denote the bit error rate experienced for the transmission rate of 1 Mbps with a SNR $\gamma = E_s/N_0$. E_s is the signal power for a Bluetooth device, and N_0 is the noise spectral density. According to [15], the bit error rate for a Bluetooth device transmitting at 1 Mbps and using GFSK modulation is given by

$$P_e = Q_1(a,b) - \frac{1}{2}\exp\left(-\frac{a^2+b^2}{2}\right) I_0(a,b); \tag{14}$$

where the constants a and b are

$$a = \sqrt{\frac{\gamma}{2}\left(1 - \sqrt{1 - \left(\frac{\sin(2\pi h)}{2\pi h}\right)^2}\right)}$$

$$b = \sqrt{\frac{\gamma}{2}\left(1 + \sqrt{1 - \left(\frac{\sin(2\pi h)}{2\pi h}\right)^2}\right)}$$

$Q_1(\cdot)$ is the first-order Marcum Q-function, and $I_0(\cdot)$ is the 0-order modified Bessel function.

The analysis for packet error rate in a Bluetooth device is similar to the corresponding analysis performed for an 802.11b station. The packet error rate due to withdrawal of packets is given by:

$$\mathrm{PER}_{c-w} = 1 - [(1 - \mathrm{BER}_n).(1 - P_{c-w}) + (1 - \mathrm{BER}_0).P_{c-w})]^k \tag{15}$$

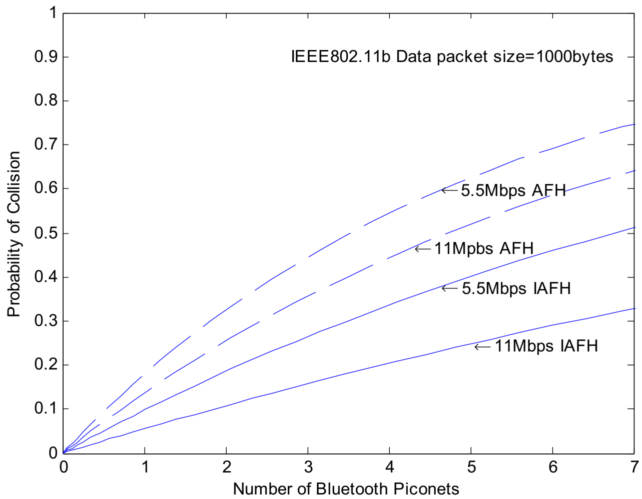

Figure 2 Probability of collision with increasing number of piconets

The packet error rate due to collisions can be derived as:

$$PER_{c-o} = 1 - [(1 - BER_n).(1 - P_{c-o}) + (1 - BER_0).P_{c-o})]^k \quad (16)$$

where BER_n and BER_o are calculated from Eq. 14 by setting the appropriate values for E_b and N_0, and k= (802.11b packet transmission time)/(625 μs)

The throughput of a Bluetooth piconet in the presence of j 802.11b stations is given by:

$$\mathrm{Th} = \frac{s}{(n+1) \times 625\ \mu s}(1 - \mathrm{PER})^j \quad (17)$$

where s is the length of a Bluetooth DH type packet in bits, and n is the length of the Bluetooth packet in slots,

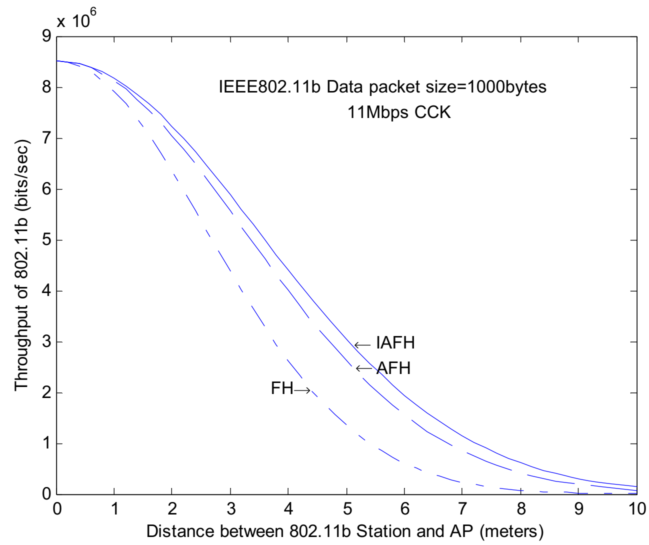

Figure 4 Throughput of the 802.11b system at 11 Mbps in the presence of four BT piconets

4 Numerical results and analysis

4.1 Performance analysis of 802.11b

We now present the simulations for the performance analysis of 802.11 networks in presence of Bluetooth piconets. The simulations were conducted in MATLAB. This section includes the results of all the experiments conducted with IAFH, FH, and AFH algorithms. Due to space constraints, the simulations results are only presented for scenario three. Fig. 2 illustrates the probability of collision of an 802.11b station with respect to number of Bluetooth piconets. The Bluetooth devices transmit DH3 packets at 1 Mbps. The 802.11b station transmits 1,000

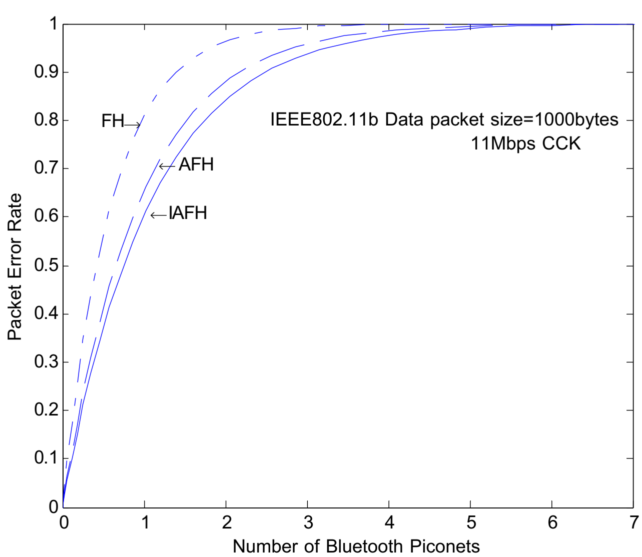

Figure 3 Packet error rate of 802.11b at 11 Mbps

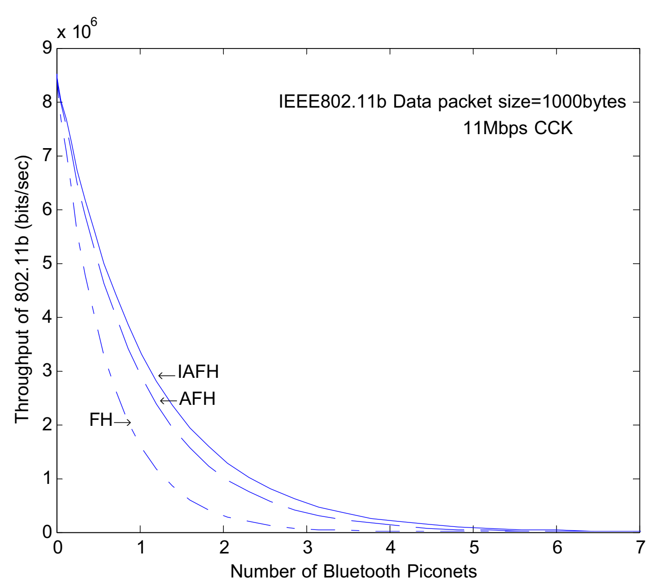

Figure 5 Throughput of 802.11b network using 11 Mbps CCK

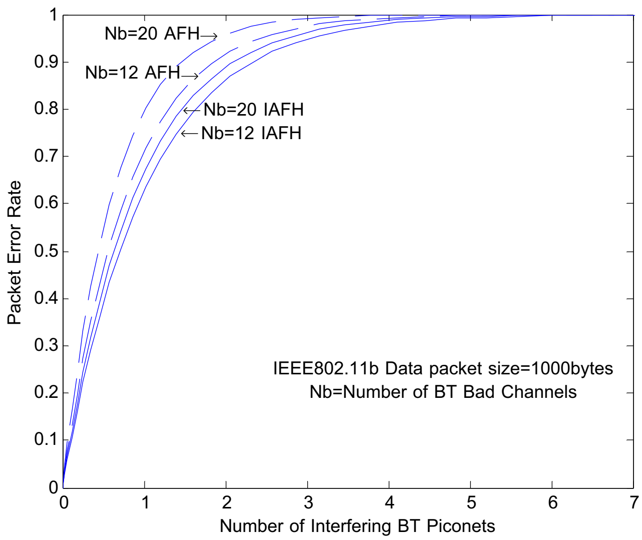

Figure 6 Packet error rate with increased number of bad channels

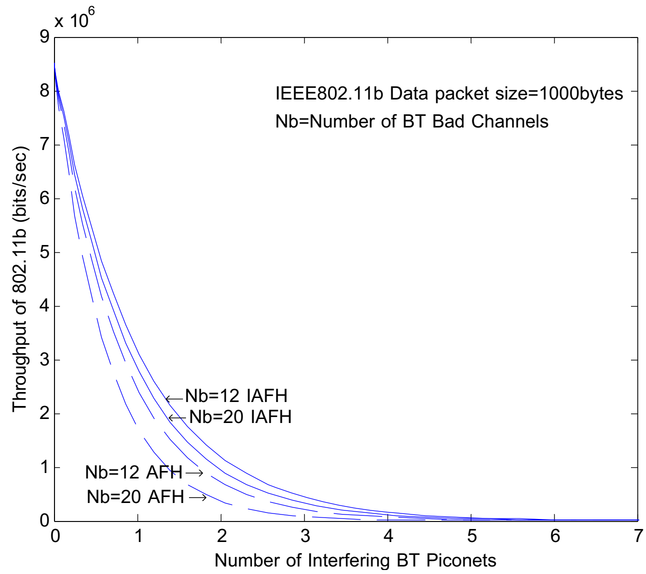

Figure 8 Throughput vs. number of interfering piconets

byte packets at 5.5 and 11 Mbps data transfer rates. The sense time for a Bluetooth device is 50 μs.

It can be seen that with the IAFH algorithm, the probability of collision is reduced considerably when compared to AFH. This is because the transmission of the Bluetooth packets is withdrawn initially and then the hopping sequence is modified. It can also be concluded that the probability of collision is low when the data rate is 11 Mbps. At higher data transmission rates, the bit error rates reduce due to an increase in bits which escape collision with the Bluetooth sense window and the 802.11b packet transmission.

Figure 3 demonstrates the packet error rate for an 802.11b station in the presence of N_i interfering Bluetooth piconets. The plot compares the results between FH, AFH and IAFH algorithms. It is assumed that the probability of packet error was only caused by the channel noise and interference collision. The packet error rate is very high even if a single Bluetooth piconet adopts FH algorithm. This is because the packet error rate is dependent on the probability of collision. As the number of collisions increase, the packet error rate also goes higher. By adopting IAFH, packet error rates can be kept to a minimal.

The effect of distance between the 802.11b transmitting station and the access point is extremely important in the interference analysis. This is because as the distance between the 802.11b station and the access point (AP) increases, more number of Bluetooth piconets are likely to cause interference.

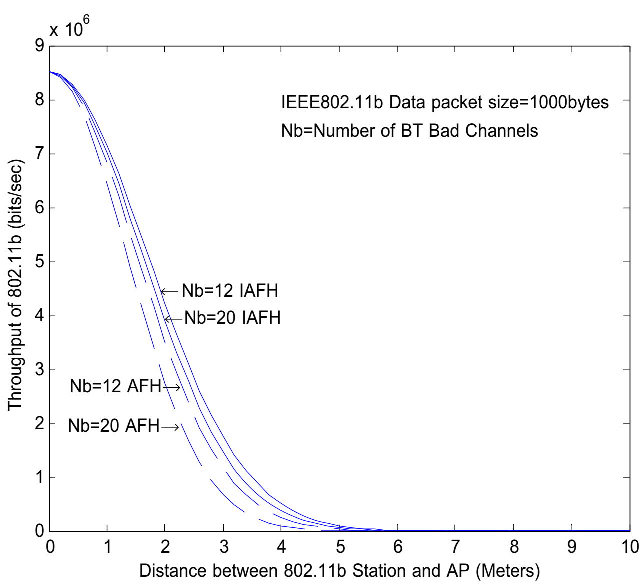

Figure 7 Throughput vs. distance

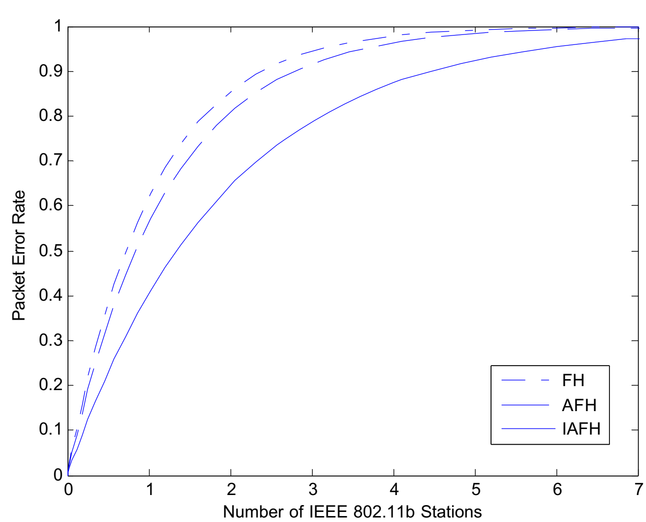

Figure 9 Packet error rate in presence of multiple 802.11 stations

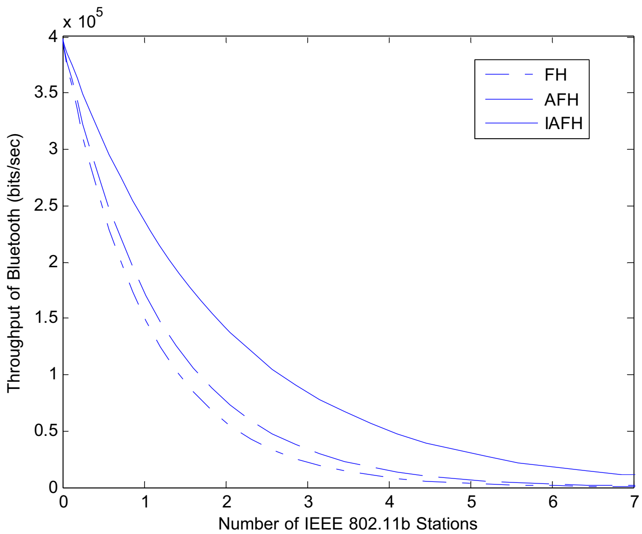

Figure 10 Throughput of Bluetooth device with increasing 802.11 stations

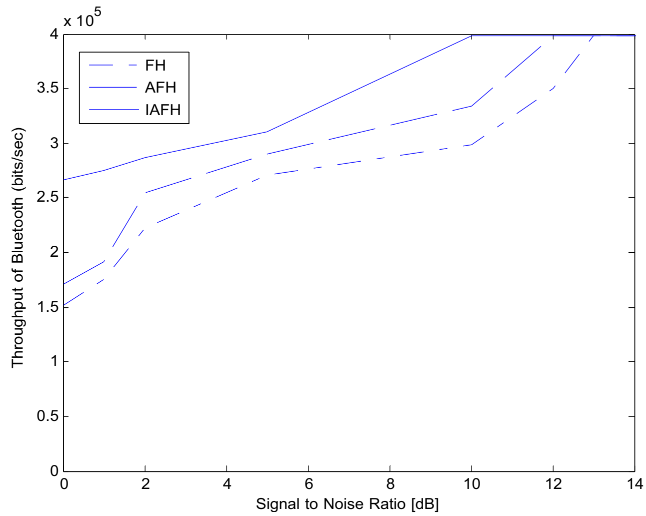

Figure 12 Throughput with different signal to noise ratios

Figure 4 shows how the throughput is affected when the distance between an 802.11b station and its AP is increasing. From Fig. 4 one can observe that as the distance between the 802.11b transmitting station and the access point is increasing, the throughput is decreasing very rapidly. With IAFH, the throughput is better when compared to FH and AFH. Figure 5 shows the throughput of 802.11b when the number of Bluetooth piconets are increasing. The increase of piconets leads to increase in packet error probability, which eventually causes the reduction in throughput. The throughput is reduced more rapidly in the case of FH and it improves with AFH and IAFH.

When all 802.11b stations are transmitting, Bluetooth devices use the minimum seven bad channels to transmit without causing significant delay. But if Bluetooth devices have more data to transmit or more number of Bluetooth piconets have to transmit, the number of bad channels will increase. Figure 6 illustrates that with the increase in the number of bad channels, the packet error rate goes higher.

Figure 7 shows the plot of the throughput vs. distance between the 802.11b station and its AP with the increasing number of bad channels. It can be seen that with the proposed algorithm, the decay in the throughput is not as fast as AFH.

Figure 8 shows the throughput of 802.11b network in the presence of Bluetooth piconets with AFH and IAFH implemented. With the increase in the number of Bluetooth piconets, due to the increase in packet error probability in the channel model, the system throughput reduces when Bluetooth devices have to select channels allocated to 802.11b stations. Further, with the increase in the number of Bluetooth bad channels the throughput reduces rapidly.

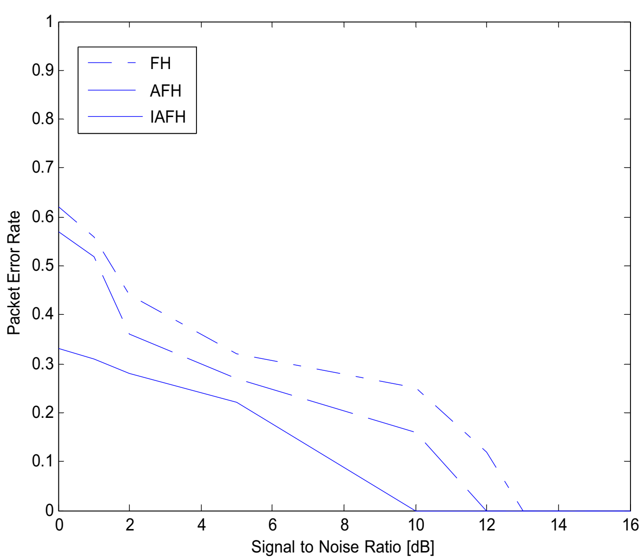

Figure 11 Packet error rate in presence of increasing signal to noise ratio

4.2 Performance analysis of Bluetooth

In this section, we present the performance analysis of Bluetooth devices in the presence of 802.11b stations. Figure 9 shows the packet error rates experienced by a Bluetooth device. The IAFH algorithm provides the minimal packet losses as compared to FH and AFH. This is because during the withdrawal of packets, packets are not lost as they do not collide with 802.11b transmission. The Bluetooth packets are withdrawn until the delay exceeds d_{accept}. The probability of collisions increases as more 802.11b stations co-exist with the Bluetooth piconet. When the total number of 802.11b stations exceeds 7, all the three algorithms suffer significant packet losses. But IAFH exhibits minimal packet losses when the number of 802.11b stations is fewer.

Figure 10 shows the throughput experienced by a Bluetooth device in the presence of multiple 802.11

stations. Though the throughput decreases with increasing interfering 802.11b stations, IAFH provides better throughput as compared to the other two algorithms.

In Fig. 11, we analyze the impact of SNR on the packet error rates. The Bluetooth data packet is of DH3 type and the 802.11b station is transmitting 1,000 byte packets at 11 Mbps. Again due to more packet withdrawals than packet collisions, the packet error rates for IAFH are lesser than that compared to FH and AFH algorithms.

Similarly, Fig. 12 depicts the throughput of Bluetooth device while transmitting DH3 packets with increasing SNR values in presence of one 802.11 station. The IAFH algorithm enabled Bluetooth piconet exhibits better throughput as compared to the FH and AFH algorithms.

5 Conclusions

In this paper, we have presented an interference-aware adaptive frequency hopping (IAFH) algorithm for Bluetooth devices to mitigate the interference between IEEE 802.11b and Bluetooth wireless networks. The main idea is that Bluetooth devices withdraw the packet transmission if the interference levels vary rapidly; otherwise transmit data using the available good channels. The algorithm can be incorporated into Bluetooth devices without any modification to the current slot structure. An analytical model of IAFH as been developed to evaluate the performance of 802.11b stations and Bluetooth devices in a mutual interference environment. Extensive simulations are conducted to evaluate the performance improvement of IAFH. Results confirm that 802.11b station and IAFH-enabled Bluetooth devices experience lower packet error rates and better throughput as compared to the frequency hopping and adaptive frequency hopping algorithms. Hence, it can be concluded that by applying the IAFH algorithm, more IEEE 802.11b stations and Bluetooth devices can coexist in a close proximity.

Acknowledgements Min Song's research is supported by NSF CAREER award CNS-0644247.

References

1. Conti A, Dardari D, Pasolini G, Andrisano O (2003) Bluetooth and IEEE 802.11b coexistence: analytical performance evaluation in fading channels.. IEEE J Sel Areas Commun 21(2):259–269, Feb
2. Golmie N, Rebala O, Chevrollier N (2003) Bluetooth adaptive frequency hopping and scheduling. IEEE MILCOM-2003 2:1138–1142, Oct
3. Golmie N (2004) Bluetooth dynamic scheduling and interference mitigation. ACM Mobile Network, MONET
4. Howitt I (2001) IEEE 802.11 and Bluetooth coexistence analysis methodology. IEEE Trans Veh Technol 2:1114–1118
5. Wadhwa M, Song M (2006) Performance of IEEE 802.11b Devices in the presence of adaptive frequency hopping enabled Bluetooth devices. Proc. of the 4th IEEE International Conference on Information Technology: Research and Education. Tel Aviv, Israel, Oct
6. Cordeiro CM, Abhyankar S, Toshiwal R, Agarwal DP (2003) A novel architecture and coexistence method to provide global access to/from Bluetooth WPANs by IEEE 802.11 WLANs. Proc. IEEE Int Perform Comput Commun, pp 23–30
7. Golmie N, Chevrollier N, Rebala O (2003) Bluetooth and WLAN coexistence: challenges and solutions. IEEE Wireless Commun 10 (6):22–29, Dec
8. Howitt I (2001) WLAN and WPAN coexistence in UL band. Trans Veh Technol 50(4):1114–1124, July
9. Feng W, Nallanathan A, Krishna GN (2003) Performance of physical (PHY) and medium access control (MAC) layers of IEEE 802.11b in the presence of Bluetooth piconets. IEEE Trans Veh Technol 2:1489–1492, Apr
10. Popovski P, Yomo H, Prasad R (2006) Dynamic adaptive frequency hopping for mutually interfering wireless personal area networks. IEEE Trans Mobile Comput 5(8):991–1003, Aug
11. Jo J-H, Jayant N (2003) Performance evaluation of multiple IEEE 802.11b WLAN stations in the presence of Bluetooth radio interference. IEEE J Commun 2:1163–1168, May
12. Bamahdi OA, Zummo SA (2006) An adaptive frequency hopping technique with application to Bluetooth–WLAN coexistence. Proc. of International Conference on Mobile Communications and Learning Technologies, Mauritius, Apr
13. IEEE (1999) Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. *IEEE Standard 802.11* [Online]. Available: http://standards.ieee.org
14. Specification of the Bluetooth system. Core, Version 2.0 [Online]. Available: http://www.bluetooth.com, Nov. 2004
15. Roh JS (2000) Performance analysis and evaluation of Bluetooth networks in wireless channel environment. Proc. IEEE Int Conf Syst Netw Commun, pp 61–65
16. Ennis G (1998) Impact of Bluetooth on 802.11 direct sequence. IEEE P802.11 Working Group Contribution, IEEE P802.11-98/319, Sep
17. Fumolari D (2001) Link performance of an embedded Bluetooth personal area network. Proc. of IEEE ICC'01, Helsinki, Finland, June
18. Golmie N, Mouveaux F (2001) Interference in the 2.4 GHz ISM band: Impact on the Bluetooth access control performance. Proc IEEE ICC'01, Helsinki, Finland, June
19. Golmie N, Van Dyck RE, Soltanian A (2001) Interference of Bluetooth and IEEE 802.11: Simulation modeling and performance evaluation. Proc. of the 4th ACM International Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems, July
20. Howitt I, Mitter V, Gutierrez J (2001) Empirical study for IEEE 802.11 and Bluetooth interoperability. Proc. of IEEE Vehicular Technology Conference (VTC), Spring
21. Kamerman A (2000) Coexistence between Bluetooth and IEEE 802.11 CCK: solutions to avoid mutual interference. IEEE P802.11 Working Group Contribution, July
22. Lansford J, Stephens A, Nevo R (2001) Wi-Fi (802.11b) and Bluetooth: enabling coexistence. IEEE Netw Mag, Sep
23. Shellhammer S (2000) Packet error rate of an IEEE 802.11 WLAN in the presence of Bluetooth. IEEE P802.15 Working Group Contribution, May
24. Zyren J (1999) Reliability of IEEE 802.11 WLANs in presence of Bluetooth radios. IEEE P802.11 Working Group Contribution, Sep.
25. Zurbes S, Stahl W, Matheus K, Haartsen J (2000) Radio network performance of Bluetooth. Proc. of IEEE ICC 2000, June

Min Song is an Associate Professor in the Department of Electrical and Computer Engineering at Old Dominion University. He received his Ph.D. in Computer Science from the University of Toledo in 2001. His research interests include protocols design and performance analysis of mobile ad hoc networks and wireless sensor networks, computer networks security, wireless communications, packet switch architecture, and distributed data mining. Dr. Song is the recipient of NSF CAREER award. He received early Tenure and Promotion in June 2007. He is the Founder and Director of Wireless Communications and Networking Laboratory. Dr. Song is an IEEE Senior member.

Sachin Shetty received Bachelor of Science in Computer Engineering from Mumbai University, India in 1998. He received his Master of Science in Computer Science from University of Toledo in 2002. He then earned his PhD degree from Old Dominion University in 2007. He has authored and co-authored 15 international refereed conference publications and book chapters. His main research areas are wireless network security, sensor networks and distributed data mining.

Deepthi Gopalpet received her Master of Science in Computer Engineering from Old Dominion University in 2007. Her research interests are network performance analysis and software engineering.