An Improved Packet Collision Analysis for Multi-Bluetooth Piconets Considering Frequency-Hopping Guard Time Effect

Ting-Yu Lin, *Member, IEEE*, Yen-Ku Liu, and Yu-Chee Tseng, *Senior Member, IEEE*

Abstract—Operating in the unlicensed 2.4-GHz ISM band, a Bluetooth piconet will inevitably encounter the interference problem from other piconets. With a special channel model and packet formats, one research issue is how to predict the packet collision effect in a multipiconet environment. In several earlier works, El-Hoiydi (2001), El-Hoiydi and Decotignie (2001), Lim *et al.* (2001), this problem is studied, but the results are still very limited in that packets are usually assumed to be uniform in lengths and in that time slots of each piconet are assumed to be fully occupied by packets. These assumptions have been successfully removed in the analytical results proposed in Lin and Tseng (2003). In this paper, we further improve the analytical results in Lin and Tseng (2003) by taking into account the frequency-hopping guard time effect in Bluetooth baseband. The result would offer a way to better estimate the network performance in a multipiconet environment.

Index Terms—Bluetooth, collision analysis, frequency hopping (FH), piconet, wireless communication, wireless personal-area network (WPAN).

I. INTRODUCTION

AS A promising wireless personal-area network (WPAN) technology, Bluetooth is expected to be used in many applications, such as wireless earphones, keyboards, and wireless access points [5], [7], [14]. Operating in the unlicensed 2.4-GHz ISM band, multiple Bluetooth piconets are likely to coexist in a physical environment. By interconnecting existing piconets, we can form a larger-scale network called scatternet. Several papers have addressed the formation algorithms and routing strategies for Bluetooth scatternets [9], [12], [13], [15], [16]. We do not intend to discuss those scatternet-related issues in this paper. Instead, we focus on the packet collision problem due to co-channel interference in a multi-Bluetooth piconets environment (whether the piconets are interconnected or not).

With a frequency-hopping (FH) radio and without coordination among piconets, transmissions from different piconets will inevitably encounter the collision problem. In a previous work [2], the author investigates the co-channel interference between Bluetooth piconets and derives an upper bound on packet error rate. The analysis in [2] has two limitations. First, all packets are assumed to be single-slot ones. Second, it is assumed that each piconet is fully loaded, in the sense that packets are sent in a back-to-back manner. These constraints greatly limit the applicability of the result in [2]. The work in [3] further extends the result in [2] by considering the physical location relationship between piconets (however, the above two limitations remain the same). The analysis in [10] does allow one-, three-, and five-slot packets, but packets of different lengths cannot mix together and each piconet is still assumed to be fully loaded.

The above limitations have been successfully removed in the analytical results proposed in [11]. A more general analysis model is proposed, where all packet types (one-slot, three-slot, and five-slot) can coexist in the network, and the system is not necessarily fully loaded. More specifically, the latter is achieved by modeling idle slots as individual single slots with *no* traffic load. In this paper, we further improve the analytical results in [11] by taking into account the FH guard time effect in Bluetooth baseband. The result would offer a way to better estimate the network performance in a multipiconet environment.

As to other references, the work in [4] considers the interference between Bluetooth and 802.11 wireless local area network (LAN) on the 2.4-GHz unlicensed band. Performance analysis of the Bluetooth physical layer is addressed in [8], but the location relation between piconets is modeled by a probabilistic model. Reference [1] further improves [8] by considering real location relation of piconets.

II. PROBLEM STATEMENT

Bluetooth is a master-driven, time-division duplex (TDD), FH wireless radio system [6]. The smallest networking unit is a *piconet*, which consists of one master and no more than seven active slaves. Each picocell channel is represented by a pseudo-random hopping sequence comprised of 79 or 23 frequencies. In Bluetooth, the hopping sequence is determined by the master's ID and clock value. For each piconet, its channel is divided into time slots, each corresponding to one random frequency. In the following discussion, we assume 79 frequencies.

T.-Y. Lin is with the Computer and Communications Research Laboratory, Industrial Technology Research Institute, Hsinchu 310, Taiwan, R.O.C. (e-mail: tylin@csie.nctu.edu.tw).

Y.-K. Liu and Y.-C. Tseng are with the Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: yklau@csie.nctu.edu.tw; yctseng@csie.nctu.edu.tw).

Fig. 1. Frequency-hopping guard time in Bluetooth.

Fig. 2. Classification of slot delimiters.

III. COLLISION ANALYSIS WITHOUT CONSIDERING GUARD TIME EFFECT

In this section, we review the collision analysis in [11], which does not consider the guard time effect. Specifically, when this effect is not considered, two packets are considered damaged if they are transmitted using the same frequency and they have nonempty overlapping in their slot time (including both transmission period and guard time period). The review would facilitate presenting our result in the next section, which considers guard time effect.

Let us consider a piconet X and another competitor piconet Y, which is regarded as the unique source of interference to X. We assume that the frequency hopping patterns of piconets are independent and random.[1] With the interference from Y, we first derive the success probability $P_S(i)$ of i-slot packets in X, where $i = 1, 3, 5$. We start by introducing the concept of "slot delimiter." A slot delimiter is the start of a slot. Consider any slot in X. One or two slot delimiters in Y may cross X's slot. However, since we are considering continuous probability, the possibility of two crossing slot delimiters can be ignored and, thus, we will deal with one crossing delimiter in the rest of the discussion. For example, for a one-slot packet in X, it succeeds only if there is no interference from the two slots before and after the delimiter, so the success probability of X's packet could be 1, 78/79, or $(78/79)^2$, depending on whether Y transmits or not (where 79 indicates the total number of possibly available frequencies for a 79-channel Bluetooth system). Below, we denote the constant factor 78/79 by P_0.

Depending on what packet(s) is divided by it, a delimiter is classified into ten types (refer to Fig. 2).

- B_1, B_2, B_5: the beginning of a one-, three-, and five-slot packet, respectively.
- B_3, B_4: the beginnings of the second and third slots of a three-slot packet, respectively.
- B_6, B_7, B_8, B_9: the beginnings of the second, third, fourth, and fifth slots of a five-slot packet, respectively.
- B_{10}: the beginning of a dummy slot.

The rate of B_1 is λ_1 per slot; the rate of each of B_2, B_3, and B_4 is λ_3, the rate of each of B_5, B_6, B_7, B_8, and B_9 is λ_5, and the rate of B_{10} is λ_0. We denote the arrival rate of B_j by $\lambda(B_j), j = 1 \ldots 10$. Given any B_j, we also define $g(j)$ to be the number of slots that follows delimiter B_j and belong to the same packet. For example, $g(1) = 1$, $g(3) = 2$, $g(7) = 3$, and $g(10) = 1$.

Intuitively, when a packet in X is crossed by a delimiter of type $B_1/B_2/B_5$ in Y, there may exist two packets (of different frequencies) in both sides of the delimiter in Y, which are potential sources of interference to X's packet. On the other hand,

In each piconet, the master and slaves take turns to exchange packets. While the master only transmits in even-numbered slots, slaves must reply in odd-numbered slots. Three packet sizes are available: one-slot, three-slot, and five-slot. For a multislot packet, its frequency is fully determined by the first slot and remains unchanged throughout.

Since Bluetooth takes a FH channel model, each packet has a guard time at the packet end. As Fig. 1 illustrates, an i-slot packet actually does not fully occupy all the i slot(s). Let T_S be the length of one time slot. For a one-slot packet, the data duration is T_{R1}. For three-slot and five-slot packets, the data durations at the last time slot are T_{R3} and T_{R5}, respectively. Those vacant periods without data transmission activities are designed mainly for radio transceiver turnover, preparing for stabilizing at the next frequency hop. Define $r_i, i = 1, 3, 5$, to be the corresponding *data occupancy ratio* for an i-slot packet at the ith time slot

$$r_1 = \frac{T_{R1}}{T_S}, \quad r_3 = \frac{T_{R3}}{T_S}, \quad r_5 = \frac{T_{R5}}{T_S}. \tag{1}$$

According to the Bluetooth specification, $r_1 = 366/625 = 0.586$, $r_3 = 372/625 = 0.595$, and $r_5 = 370/625 = 0.592$ for one-slot, three-slot, and five-slot data packets (without error correction capability), respectively. Since the differences between r_i's are very small in Bluetooth, below we will approximate them by a single value r for simplicity, i.e., $r = r_1 = r_3 = r_5$.

In this paper, we consider N piconets coexisting in a physically closed environment. Since no coordination is possible between piconets, each piconet has $N - 1$ potential competitors. In any time instance, if two piconets transmit with the same frequency, the corresponding two packets are considered damaged (note that during the guard time periods, a host is not considered transmitting). Our goal is to derive an analytic model to evaluate the impact of collisions in such a multipiconet environment.

We assume a Poisson traffic in each piconet, and let λ_1, λ_3, and λ_5 be the arrival rates of one-, three-, and five-slot packets per slot, respectively, to a piconet. Note that we do not model the link-layer automatic repeat requests (ARQs) mechanism. The given arrival rates account for both original and retransmitted packets. For a multislot packet, only the header slot counts as arrival. It is easy to see that $\lambda_1 + 3\lambda_3 + 5\lambda_5 \leq 1$. Further, we regard the remaining vacant slots as "dummy" single-slot packets. Thus, the arrival rate of such dummy (one-slot) packets is $\lambda_0 = 1 - (\lambda_1 + 3\lambda_3 + 5\lambda_5)$.
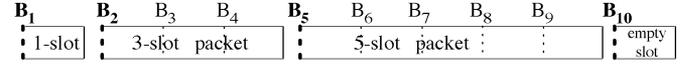
[1]In the Bluetooth specification, the hopping patterns are pseudorandom. However, without the randomness assumption, the analysis would be difficult. We will validate this assumption by generating pseudorandom hopping patterns in our simulations to verify the impact of such an assumption.
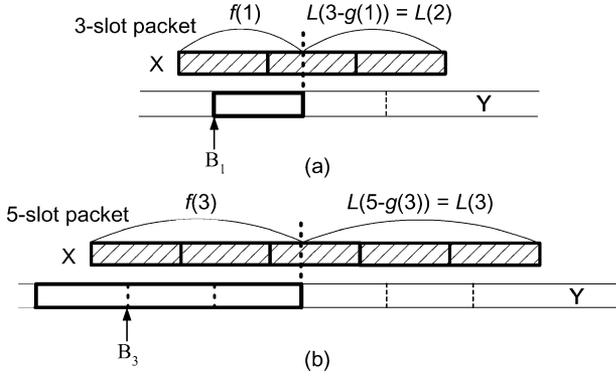
Fig. 3. Analysis of success probabilities for (a) three-slot and (b) five-slot packets.

when the delimiter is of the other types, the interference source reduces to one.

Definition 1: Given any i-slot packet in piconet X and any interference source piconet Y, define $L(k)$, $k < i$, to be the probability that the packet of X experiences no interference from Y starting from the delimiter of Y crossing the $(i-k+1)$th slot of the packet to the end of the packet, under the condition that the aforementioned delimiter is of type $B_1/B_2/B_5/B_{10}$. For $k \le 0$ (in which case the above definition is not applicable), $L(k) = 1$.

The above probability function is introduced in [11]. Intuitively, $L(k)$ is the success probability of the last k slots of X's packet excluding the part before the first delimiter of Y crossing these k slots, given the above delimiter type constraint. With this definition, we can find $P_S(i)$ by repeatedly cutting off some slots from the head of X's packet, until there is no remaining slot

$$P_S(i) = \sum_{j=1}^{10} \lambda(B_j) \cdot f(j) \cdot L(i - g(j)) \qquad (2)$$

where

$$f(j) = \begin{cases} (1 - \lambda_0) \cdot P_0^2 + \lambda_0 \cdot P_0, & \text{if } j = 1, 2, 5 \\ (1 - \lambda_0) \cdot P_0 + \lambda_0, & \text{if } j = 10 \\ P_0, & \text{otherwise} \end{cases}.$$

In the equation, we consider each type B_j, $j = 1, \ldots, 10$, of the first delimiter in Y crossing X's packet. The corresponding probability is $\lambda(B_j)$. Function $f(j)$ gives the probability that the packet(s) of Y on both sides of the first delimiter B_j does (do) not interfere with X's packet. It remains to consider the success probability of the last $i - g(j)$ slots of X's packet, excluding the part before the first delimiter of Y crossing these $i - g(j)$ slots (which must be of delimiter type $B_1/B_2/B_5/B_{10}$). This is reflected by the last factor $L(i - g(j))$.

For example, Fig. 3(a) illustrates a three-slot packet in X. The first delimiter in Y crossing the three-slot packet is of type B_1. The success probability of the first part in X is $f(1) = (1 - \lambda_0) \cdot P_0^2 + \lambda_0 \cdot P_0$. Intuitively, if the packet of Y before the delimiter B_1 is a dummy packet (of probability λ_0), the success probability is simply P_0; otherwise, there are two packets which

are potential interference sources, and the success probability is P_0^2. Then, we can move on to consider the success probability of the remaining part of X after the second delimiter in Y, which is given by $L(2)$. $L(2)$ computes the success probability of the last two slots excluding the part before the dotted line. By multiplying $f(1)$ with $L(2)$, we obtain the success probability of the whole packet. Another example of a five-slot packet is shown in Fig. 3(b). The first delimiter in Y crossing the five-slot packet is of type B_3. So the success probability from the beginning of the packet up to the third delimiter in Y crossing the packet is $f(3)$. For the remaining part, the success probability is $L(3)$. So the success probability of the five-slot packet is $f(3) \cdot L(3)$.

The remaining part of X's packet covered by $L(k)$ must start with a delimiter in Y of a restricted type of $B_1/B_2/B_5/B_{10}$. Since the packet in Y after the delimiter must be a complete packet, it can be solved recursively as follows ($k > 0$):

$$\begin{aligned} L(k) = {} & \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot L(k - g(10)) \\ & + \frac{\lambda_1}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot L(k - g(1)) \\ & + \frac{\lambda_3}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot L(k - g(2)) \\ & + \frac{\lambda_5}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot L(k - g(5)). \quad (3) \end{aligned}$$

In each term, the first part is the probability of the corresponding packet type in Y. As to the boundary conditions, $L(k) = 1$, for $k \le 0$.

Next, we consider an N-piconet environment. For each piconet X, there are $N - 1$ piconets each serving as an interference source. Since these interferences are uncoordinated and independent, the success probability of an i-slot packet in X can be written as $P_S(i)^{N-1}$. So the network throughput of X is

$$\begin{aligned} T = {} & \lambda_1 \cdot P_S(1)^{N-1} \cdot R_1 + 3 \cdot \lambda_3 \cdot P_S(3)^{N-1} \cdot R_3 \\ & + 5 \cdot \lambda_5 \cdot P_S(5)^{N-1} \cdot R_5 \quad (4) \end{aligned}$$

where R_1, R_3, and R_5 are the per-slot data rates of one-, three-, and five-slot packets, respectively (for example, if DH1/DH3/DH5 are used, $R_1 = 216/1 = 216$, $R_3 = 1464/3 = 488$, and $R_5 = 2712/5 = 542.4$ bits per slot, where 216, 1464, and 2712 are the numbers of bits contained in DH1, DH3, and DH5 packets, respectively). Note that we will use "bits per slot" as our metric to calculate network throughput. This also explains why factors of three and five are multiplied to the second term and third term in (4) when calculating T. The aggregate network throughput of N piconets is $N \times T$.

IV. ENHANCED COLLISION ANALYSIS CONSIDERING GUARD TIME EFFECT

Next, we improve the collision analysis in [11] by considering the guard time effect. During guard time periods, hosts are considered not transmitting. Thus, collisions may only occur in real transmission periods. Again, we consider a piconet X and another competitor piconet Y, which is the interference source of

X. With guard time effect, the probability $P_S(i)$ should be reformulated as follows:

$$P_S(i) = (1 - r) \cdot \sum_{j=1}^{10} \lambda(B_j) \cdot \tilde{f}(j) \cdot L(i - g(j)) + (2r - 1)$$
$$\cdot \sum_{j=1}^{10} \lambda(B_j) \cdot f(j) \cdot L(i - g(j)) + (1 - r)$$
$$\cdot \sum_{j=1}^{10} \lambda(B_j) \cdot f(i,j) \cdot \tilde{L}(i - g(j)) \qquad (5)$$

where

$$\tilde{f}(j) = \begin{cases} 1, & \text{if } j = 10 \\ P_0, & \text{otherwise} \end{cases}$$

and

$$f(i,j) = \begin{cases} (1 - \lambda_0) \cdot P_0 + \lambda_0, & \text{if } i = 1 \text{ and } j = 1, 2, 5 \\ (1 - \lambda_0) \cdot P_0^2 + \lambda_0 \cdot P_0, & \text{if } i = 3, 5 \text{ and } j = 1, 2, 5 \\ (1 - \lambda_0) \cdot P_0 + \lambda_0, & \text{if } j = 10 \\ P_0, & \text{otherwise} \end{cases}.$$

$\tilde{L}(k)$ can be derived in two cases. When $k > 1$

$$\tilde{L}(k) = \frac{\lambda_0}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot \tilde{L}(k - g(10))$$
$$+ \frac{\lambda_1}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot \tilde{L}(k - g(1))$$
$$+ \frac{\lambda_3}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot \tilde{L}(k - g(2))$$
$$+ \frac{\lambda_5}{\lambda_0 + \lambda_1 + \lambda_3 + \lambda_5} \cdot P_0 \cdot \tilde{L}(k - g(5)) \qquad (6)$$

and when $k \leq 1$, $\tilde{L}(k) = 1$.

In (5), the definitions of $\lambda(B_j)$, $f(j)$, and $g(j)$ remain the same as those in (2). Equation (6) differs from (3) in its boundary conditions. To explain the above formulations, we introduce the concept of *critical section* (CS) within a single time slot. Since guard time periods are interleaving real data packets, the position (or offset) of the first slot delimiter B_j in Y crossing X's packet does affect the packet success probability $P_S()$. So, we partition the first slot of any X's packet into three critical sections, CS1, CS2, and CS3, which occupy $1 - r$, $2r - 1$, and $1 - r$ proportions of the first slot in this packet, respectively (recall that r is the approximated data occupancy ratio). An illustration is shown in Fig. 4.

Fig. 4 shows how the slot delimiter of Y may cross the above critical sections. In the first case, the delimiter B_j falls in X's CS1. If fortunately B_j equals B_1, B_2, B_5, or B_{10} (i.e., the beginning of a new packet), the darkened area before B_j in Y belongs to guard time. So the CS1 (before B_j) of X would experience no interference from Y. In this case, the packet success probability $P_S()$ will increase, and this effect is reflected in the new function $\tilde{f}(j)$ [compared to the function $f(j)$ in (2)].

In the second case of Fig. 4, the delimiter B_j falls in X's CS2. The aforementioned benefit would disappear because certain part in the beginning of CS1 will fall out of the range of
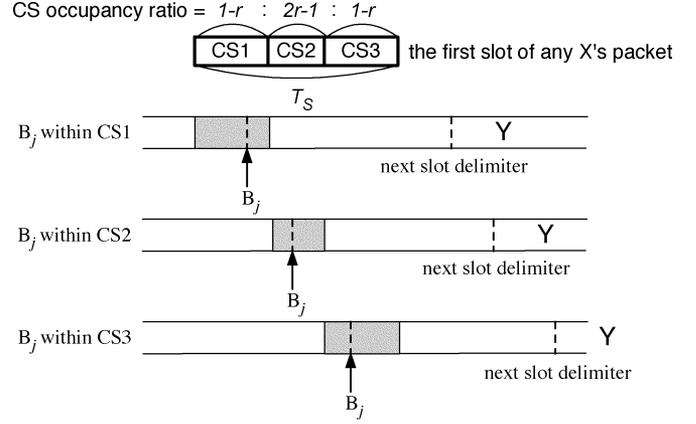

Fig. 4. Critical sections and collision analysis by considering guard time effect.

Y's guard time period. So $P_S()$ remains the same in this case [compared with (2)].

In the third case of Fig. 4, the delimiter B_j resides in X's CS3. If fortunately the packet in X is a single-slot packet, then CS3 of X would be guard time. If so, the darkened area after B_j in Y would not pose any interference to X's packet. Even if the packet in X is not a single-slot packet, this also means that the last slot delimiter of Y crossing X's packet would fall in a CS3 of X, a guard time period. If this last delimiter is the beginning of a packet, $P_S()$ can also benefit in this case. This is reflected by defining the new function $f(i,j)$, and by ending the recursive formula $\tilde{L}()$ at an earlier time (whenever k decreases to 1 or less).

Here, we give some examples for our analysis.

- **Case I—B_j within CS1 (with probability $1 - r$).**

  Consider the example in Fig. 5(a). Benefiting from the guard time of Y's packet before B_1, the packet success probability of X's one-slot packet is P_0. In comparison, without considering the guard time effect, the formulation in [11] would suggest a lower success probability of P_0^2.

- **Case II—B_j within CS2 (with probability $2r - 1$).**

  Consider the example in Fig. 5(b). The packet success probability of X's one-slot packet is P_0^2, which remains the same as that in the analysis of [11]. Guard time does not reduce the probability of interference from Y's packet(s) in this case.

- **Case III—B_j within CS3 (with probability $1 - r$).**

  In Fig. 5(c), the packet in X under consideration is a three-slot packet. The CS3 in X's first slot does not represent a guarding period in this example. Hence, the two packets of Y right before and after the first delimiter B_1 still pose as a potential interference source to X's packet. However, since B_1 falls in CS3, the last slot delimiter of Y crossing X's packet (denoted by SD_{last} in the figure) must reside in a guarding period. As a result, no interference needs to be taken into account after SD_{last}. In this example, the success probability of X's packet is P_0^3 (compared with a success probability of P_0^4 in the analysis of [11] without considering guard time effect). These are reflected in the function $f(i,j)$ and in the boundary conditions for $\tilde{L}(k)$.
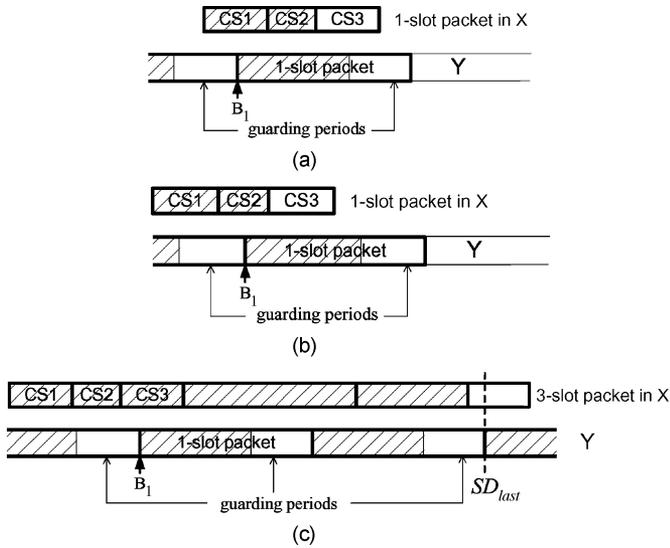
Fig. 5. Collision analysis examples. B_j falling in (a) CS1, (b) CS2, and (c) CS3.

Finally, for an N-piconet environment, the network throughput T of piconet X can be derived by the same equation (4). The aggregate network throughput of N piconets is, therefore, $N \times T$.

We remark that we use the uniform data occupancy ratio r to approximate r_1, r_3, and r_5. One concern is that the proposed analysis may have a certain level of bias. We examine this concern through simulation experiments in Section V.

V. SIMULATION AND EXPERIMENTAL RESULTS

This section presents our simulation and experimental results based on C++ programs. We test different numbers of piconets. Each simulation run lasts for 10 000 time slots. Frequency-hopping sequences are simulated by pseudorandom sequences as defined in Bluetooth (recall that in our analysis, the sequences are assumed to be random). For simulation results, we use the exact values of r_1, r_3, and r_5. For analytical results, we use the approximated r. Packets being simulated are DH1/3/5. We do not model the physical environment of the networks (such as relative locations of Bluetooth devices and their transmission power), so collisions are modeled only at the logical level.

Assuming $\lambda_1 = \lambda_3 = \lambda_5$, we inject traffic loads of 100% and 70% to each piconet (the load reflects the percentage of busy slots in a piconet, i.e., $\lambda_1 + 3\lambda_3 + 5\lambda_5 = 1$ and 0.7). Also, note that as mentioned earlier, the traffic loads are aggregated loads, which include both original and retransmitted traffics. Fig. 6(a) and (b) plots the error probabilities of DH1/3/5 packets from both analysis and simulation results under different N's. To verify our analysis results, Fig. 6(c) and (d) further plots the relative errors between simulation and analytical results. The relative error is calculated by [(packet error probability from analysis)-(packet error probability from simulation)]/(packet error probability from simulation). The packet error probability increases as the traffic load or the number of pi-
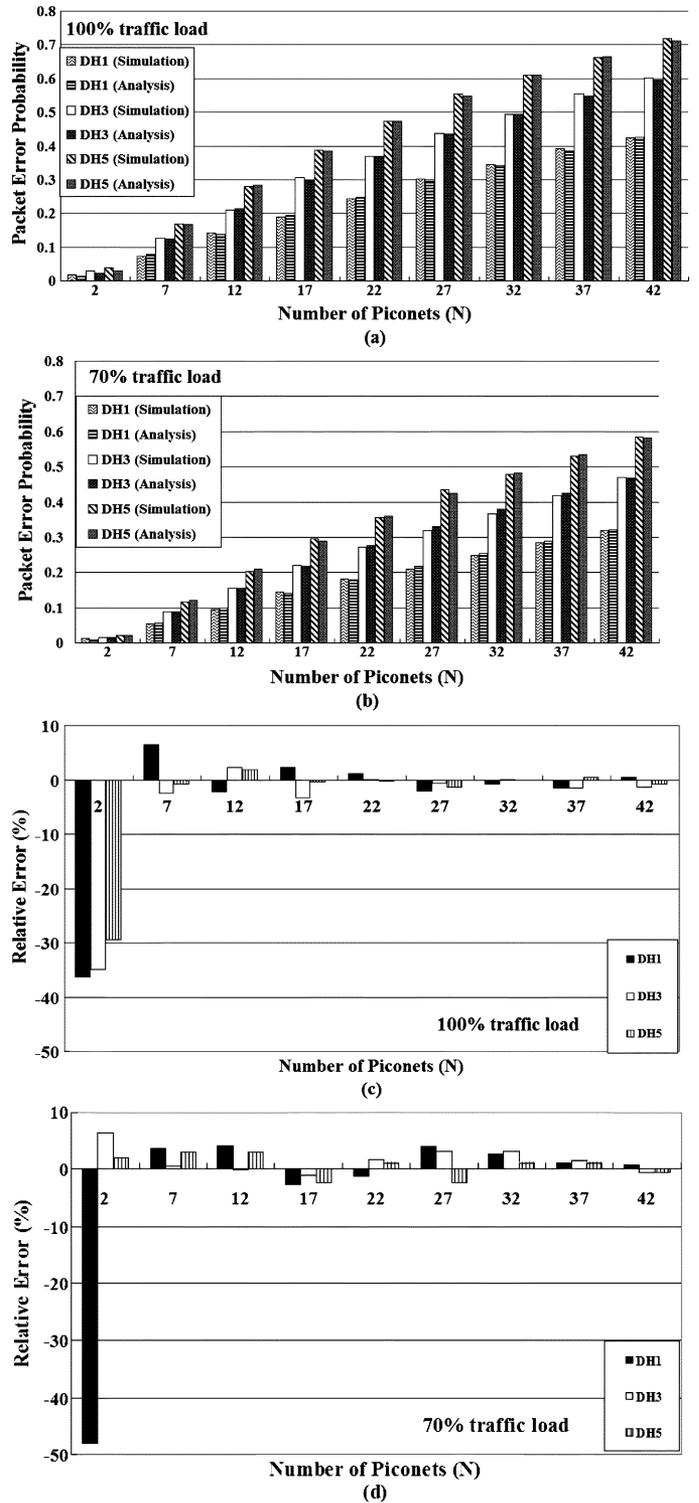
Fig. 6. Packet error probabilities under (a) 100% traffic load and (b) 70% traffic load, and relative errors between simulation and analytical results under (c) 100% traffic load and (d) 70% traffic load.

conets grows. Smaller packets suffer less collisions than larger ones due to the formers' shorter transmission durations. However, larger packets are much more bandwidth-efficient than smaller ones (e.g., a DH5 carries 542.4/216 times more bits per slot than a DH1 does). This observation leads us to conduct the next experiment by using network throughput as the metric.
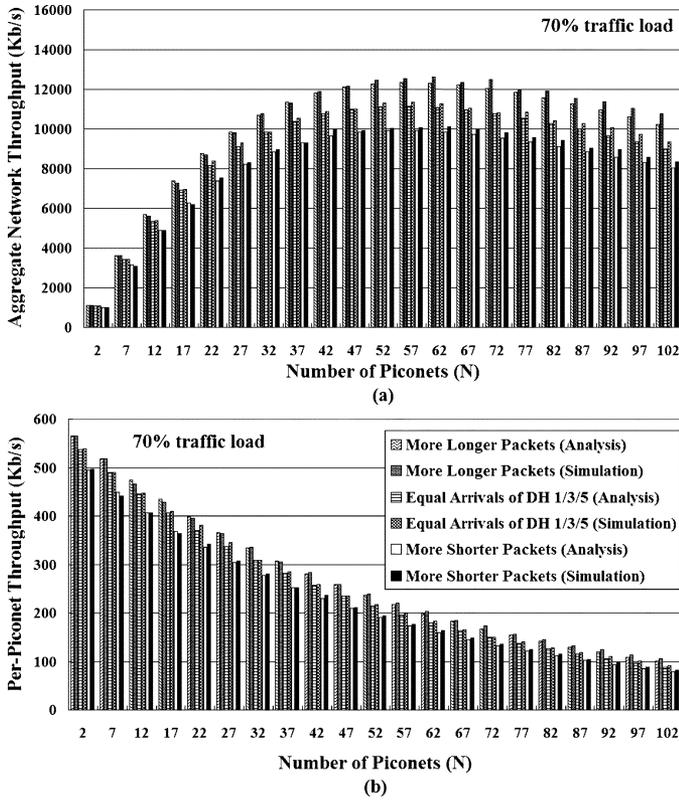
Fig. 7. (a) Aggregate network throughput under 70% traffic load. (b) Per-piconet throughput under 70% traffic load.
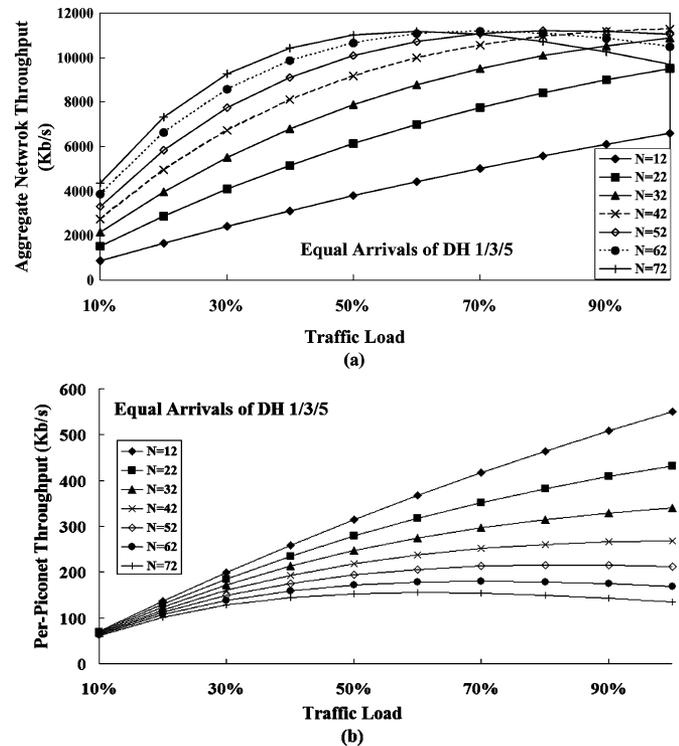

Fig. 8. (a) Aggregate network throughput versus traffic load. (b) Per-piconet throughput versus traffic load.
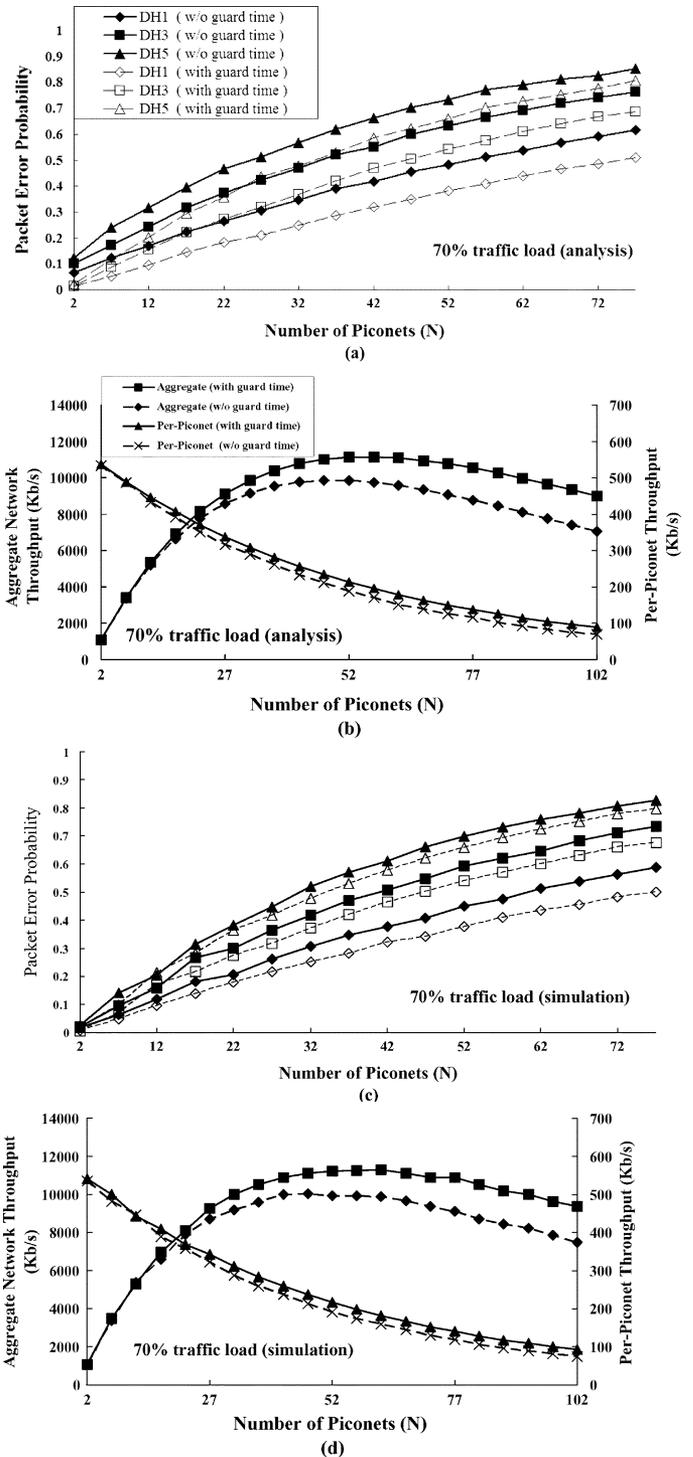

Fig. 9. Analytical results to evaluate the guard time effect on (a) packet error probability and (b) network throughput, and simulation results to evaluate the guard time effect on (c) packet error probability and (d) network throughput.

Next, we evaluate the aggregate network throughput $(N \times T)$ and per-piconet throughput. We show the case of 70% traffic load. We consider three arrival models: one with equal arrivals of DH1/3/5 packets $(\lambda_1 = \lambda_3 = \lambda_5)$, one with more shorter packets $(\lambda_1 : \lambda_3 : \lambda_5 = 3 : 2 : 1)$, and one with more longer packets $(\lambda_1 : \lambda_3 : \lambda_5 = 1 : 2 : 3)$. Both analytical and simulation results are shown in Fig. 7. The aggregate throughput saturates at a certain point as the number of piconets increases, and then drops. Different from the earlier observation, the result indicates that longer packets are more preferable in terms of

throughput because the collision problem can be compensated by the benefit of higher bandwidth efficiency. Also, in terms of per-piconet throughput, the performance consistently degrades as N increases, which is reasonable due to the impact of increased packet error probability.

Fig. 8 plots the network throughputs vs. traffic loads under fixed values of N. It indicates that the throughput goes up steadily as traffic load increases when $N \leq 42$. However, for larger N's, throughputs saturate at certain points, due to more serious collisions. The results can be used to estimate the proper number of piconets to be deployed in a physical area, in terms of the aggregate network throughput and per-piconet throughput under our packet collision model.

Finally, we observe the guard time effect. Fig. 9 compares the packet error probability and network throughput when the guard time effect is considered and when the guard time effect is not considered (i.e., the case in [11]) under 70% traffic load with equal arrivals of DH1/3/5 packets. Both analytical and simulation results are shown, which do not exhibit much difference. From the figure, we observe that due to less stronger conditions for packet collision, at $N = 42$, when considering the guard time effect, the packet error probabilities for DH1/3/5 packets are reduced by about 0.1, 0.09, and 0.075, respectively, and the aggregate network throughput is improved by about 1000 kb/s. The difference of packet error probability for DH1 is more significant because guard time takes a relatively larger part in such packets. As to throughput, the improvement when considering guard time effect tends to increase as N increases.

VI. Conclusion

We have presented new collision analysis results for Bluetooth piconets by taking into account the guard time effect. The result improves earlier works by removing several unrealistic limitations. Our simulation results match quite well with our analytical results, which justify the correctness. In addition, the guard time effect, which influences system throughput as the number of piconets grows, has been proven to be noticeable. This further validates the usefulness of the proposed enhanced analysis in this work.

References

[1] C. de Morais Cordeiro, D. Sadok, and D. P. Agrawal, "Piconet interference modeling and performance evaluation of Bluetooth MAC protocol," in *Proc. IEEE Int. Conf. Global Telecommunications (GLOBECOM)*, 2001, pp. 2870–2874.

[2] A. El-Hoiydi, "Interference between Bluetooth networks—Upper bound on the packet error rate," *IEEE Commun. Lett.*, vol. 5, pp. 245–247, June 2001.

[3] A. El-Hoiydi and J. D. Decotignie, "Soft deadline bounds for two-way transactions in Bluetooth piconets under co-channel interference," in *Proc. IEEE Int. Conf. Emerging Technologies Factory Automation*, Oct. 2001, pp. 144–151.

[4] N. Golmie and F. Mouveaux, "Interference in the 2.4 GHz ISM band: Impact on the Bluetooth access control performance," in *Proc. IEEE Int. Conf. Communications (ICC)*, 2001, pp. 2540–2545.

[5] D. Groten and J. Schmidt, "Bluetooth-based mobile ad hoc networks: Opportunities and challenges for a telecommunications operator," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, 2001, pp. 1134–1138.

[6] (2001, Feb.) Bluetooth SIG Specification v1.1. [Online]. Available: http://www.bluetooth.com

[7] P. Johansson, M. Kazantzidis, R. Kapoor, and M. Gerla, "Bluetooth: An enabler for personal area networking," *IEEE Network*, vol. 15, pp. 28–37, Sept./Oct. 2001.

[8] A. Karnik and A. Kumar, "Performance analysis of the Bluetooth physical layer," in *Proc. IEEE Int. Conf. Personal Wireless Communications (ICPWC)*, Dec. 2000, pp. 70–74.

[9] C. Law, A. K. Mehta, and K.-Y. Siu, "Performance of a new Bluetooth scatternet formation protocol," in *Proc. ACM Int. Symp. Mobile Ad Hoc Networking Computing (MobiHoc)*, 2001, pp. 183–192.

[10] Y. Lim, J. Kim, S. L. Min, and J. S. Ma, "Performance evaluation of the Bluetooth-based public Internet access point," in *Proc. IEEE Int. Conf. Information Networking*, 2001, pp. 643–648.

[11] T.-Y. Lin and Y.-C. Tseng, "Collision analysis for a multi-Bluetooth picocells environment," *IEEE Commun. Lett.*, vol. 7, pp. 475–477, Oct. 2003.

[12] T.-Y. Lin, Y.-C. Tseng, and K.-M. Chang, "Formation, routing, and maintenance protocols for the BlueRing scatternet of Bluetooths," *Wireless Commun. Mobile Comput.*, vol. 3, no. 4, pp. 517–537, June 2003.

[13] G. Miklos, A. Racz, Z. Turanyi, A. Valko, and P. Johansson, "Performance aspects of Bluetooth scatternet formation," in *Proc. ACM Int. Symp. Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2000, pp. 147–148.

[14] K. Sairam, N. Gunasekaran, and S. Reddy, "Bluetooth in wireless communication," *IEEE Commun. Mag.*, pp. 90–96, June 2002.

[15] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. LaMaire, "Distributed topology construction of Bluetooth personal area networks," in *Proc. IEEE INFOCOM*, 2001, pp. 1577–1586.

[16] G. V. Zaruba, S. Basagni, and I. Chlamtac, "Bluetrees—Scatternet formation to enable Bluetooth-based ad hoc networks," in *IEEE Int. Conf. Communications (ICC)*, 2001, pp. 273–277.

Ting-Yu Lin (M'04) received the Ph.D. degree in electrical engineering and computer science from the National Chiao-Tung University, Hsinchu, Taiwan, R.O.C., in 2002.

She is currently a Software Engineer with the Computer and Communications Research Laboratory, Industrial Technology Research Institute, Hsinchu, Taiwan, R.O.C. Her research interests include wireless communication, mobile computing, WLANs/WPANs, and energy conservation designs (software-level).

Dr. Lin was a recipient of the Phi Tau Phi Scholastic Honor Award.

Yen-Ku Liu received the B.S. and M.S. degrees in computer science from the National Chiao-Tung University, Hinshu, Taiwan, R.O.C., in 2002 and 2004, respectively.

His research interests include wireless networks, sensor networks, and Bluetooth technology.

Yu-Chee Tseng (S'91–M'95–SM'03) received the B.S. and M.S. degrees in computer science from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1985, and the National Tsing-Hua University, Hinshu, Taiwan, R.O.C., in 1987, and the Ph.D. degree in computer and information science from The Ohio State University, Columbus, in 1994.

He was with D-LINK Inc., as an Engineer in 1990. He was an Associate Professor with Chung-Hua University, Hsinchu, Taiwan, R.O.C. (1994–1996) and the National Central University, Taoyuan, Taiwan, R.O.C. (1996–1999), and a Full Professor with the National Central University (1999–2000). Since 2000, he has been a Full Professor in the Department of Computer Science and Information Engineering, National Chiao-Tung University, Hsinchu, Taiwan, R.O.C. He has guided students to participate in several national programming contests and received several awards. His research interests include mobile computing, wireless communication, network security, and parallel and distributed computing.

Dr. Tseng is a two-time recipient of the Outstanding Research Award, National Science Council, R.O.C., in 2001–2002 and 2003–2005, and a recipient of the Best Paper Award at the International Conference on Parallel Processing (2003). Several of his papers have been chosen as Selected/Distinguished Papers in international conferences. He served as Program Chair of the Wireless Networks and Mobile Computing Workshop (2000–2001), a Vice Program Chair in the International Conference on Distributed Computing Systems (ICDCS) (2004), and as Vice Program Chair of the IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2004). He was an Associate Editor for *The Computer Journal*, a Guest Editor for the *ACM Wireless Networks* Special Issue on Advances in Mobile and Wireless Systems, the IEEE TRANSACTIONS ON COMPUTERS Special Issue on Wireless Internet, the *Journal of Internet Technology* Special Issue on Wireless Internet: Applications and Systems, *Wireless Communications and Mobile Computing* Special Issue on Research in Ad Hoc Networking, Smart Sensing, and Pervasive Computing, *Telecommunication Systems* Special Issue on Wireless Sensor Networks, and the *Journal of Information Science and Engineering* Special Issue on Mobile Computing, and as an Editor for the *Journal of Information Science and Engineering*. He is a member of the Association for Computing Machinery (ACM).