Soft real-time communications over Bluetooth under interferences from ISM devices

J. L. Sevillano, D. Cascado, F. Díaz del Río, S. Vicente, G. Jimenez
and A. Civit-Balcells

ETSII, Universidad de Sevilla, Av. Reina Mercedes, s/n. 41012, Sevilla, Spain

SUMMARY

Bluetooth is a suitable technology to support soft real-time applications like multimedia streams at the personal area network level. In this paper, we analytically evaluate the worst-case deadline failure probability of Bluetooth packets under co-channel interference as a way to provide statistical guarantees when transmitting soft real-time traffic using ACL links. We consider the interference from independent Bluetooth devices, as well as from other devices operating in the ISM band like 802.11b/g and Zigbee. Finally, we show as an example how to use our model to obtain some results for the transmission of a voice stream.

KEY WORDS: Bluetooth; co-channel interference; soft real-time; worst-case deadline failure probability

1. INTRODUCTION

Bluetooth is currently one of the most successful standards for low-power, short-range wireless personal area networks (WPAN). A wide range of devices are being introduced, with applications supporting asynchronous and/or isochronous data transfers. The motivation of this work is that more and more wireless devices are also introduced that operate in the same frequency bands than Bluetooth devices (the so-called ISM-Industrial, Scientific & Medical bands), including IEEE 802.11, IEEE 802.15.4, WirelessUSB, etc. Depending on the level of sensitivity to the interference from other devices, some types of traffic may suffer from unpredictable delays or packet losses due to these interferences. In this paper, we evaluate the real-time behaviour of Bluetooth communications, and show how interferences from other ISM devices can be taken into account.

A typical Bluetooth system is composed of a small number of devices that form a wireless network called a *piconet*. Connections are established *ad hoc* by a Bluetooth unit that becomes a

master so that the other units (*slaves*) synchronize with it. The Bluetooth channel is then divided into slots of length 625 μs so that time slots are alternatively used by master and slaves (time division duplex). Any unit may function as a master or as a slave (this role is maintained only for the duration of the piconet), but although it may participate as slave in multiple piconets, it can only be a master in one piconet.

Two types of connections can be established in Bluetooth [1]: synchronous connection-oriented (SCO) and asynchronous connectionless links (ACL). In ACL links, the master of the piconet performs a polling among all the slaves, although the polling policy is not specified by the current Bluetooth specification because the best policy depends on the application. Slave devices can only transmit if the master has requested them in the preceding slot. ACL packets require acknowledgement and they are retransmitted in case of errors using a fast automatic repeat request (ARQ) scheme until they are successfully delivered. Several types of ACL packets are defined depending on whether they use forward error correction (FEC) or not. Also, multi-slot packet transmissions are allowed (three or five slots packets).

On the other hand, SCO links are based on a fixed and periodic pre-allocation of slots (every two, four or six slots) for guaranteed transmission of continuous (audio–video) streams. SCO packets are always one slot length and they are never retransmitted. If errors occur during transmission (not corrected by FEC) they are ignored and the packet is delivered as it is received. Finally, the master does not have to poll the slaves in SCO links, so a slave may transmit an SCO packet without a previous request.

A typical use of Bluetooth would be the support of soft real-time applications like multimedia streams. In these applications, the use of SCO links monopolizes bandwidth and leaves very little room for other ACL links. Several authors [2, 3] have proposed that ACL links could be used to carry voice and other isochronous traffic. These applications require QoS guarantees in terms of delay, delay variation and loss rate [4], so several attempts have been made to provide these guarantees in ACL links [5, 6]. However, these works ignore the possibility of interferences from other devices. When using wireless connections we should always keep in mind that a system with several interfering devices is probabilistic in nature. In the case of ACL links, although the polling performed by the master provides determinism within a piconet, there is always the possibility of a collision with other Bluetooth devices connected to different logical channels, as well as with other interfering devices operating in the ISM band: WLANs like IEEE 802.11, IEEE 802.15.4 networks [7], WirelessUSB [8], microwave ovens, etc.

The problem of coexistence with other devices has received considerable attention in the literature [9–14]. However, these studies are limited to the computation of parameters like bit error rates (BER) or the probability that a given packet suffers a collision. To the best of our knowledge, they have not been used to provide real-time guarantees for Bluetooth connections taking into account co-channel interferences. Maybe the only exception is [15] where the authors compute the probability that a packet does not meet its deadline for two-way transactions under interferences from other Bluetooth devices. However, they do not take into account the effect of the polling algorithm or the interference between SCO and ACL packets. Furthermore, their sources of interference are only other independent Bluetooth devices.

In this paper, we present an analysis that allows us to obtain the probability that a Bluetooth packet exceeds its deadline, taking into account the effect of the polling within the piconet, the presence of SCO links and the interferences from other ISM devices, including 802.11b/g, Bluetooth and Zigbee. Our approach was outlined in a previous paper [16], where we proposed the use of schedulability tests like those that have been used to provide guarantees on message

delays in distributed real-time systems [17]. These analyses use the worst-case transmission times to bound message response times, and to assess the schedulability of the system. When packet transmission times cannot be upper-bounded, as is the case with Bluetooth piconets under co-channel interferences, then a probabilistic analysis is needed. One possibility is statistical rate monotonic scheduling (SRMS) [18]. A simpler approach is the probabilistic time-demand analysis used for single-processor systems with semi-periodic tasks (tasks released periodically but with random computation times) in Reference [19]. However, any of these approaches are difficult and computationally expensive. In this paper, we would rather evaluate the worst-case deadline failure probability (WCDFP), considering collisions from other ISM devices in a similar fashion to that used to take into account transmission errors in controller area network (CAN)-based systems in Reference [20]. To illustrate our methodology, we show some results for the transmission of a voice stream, '...one of the most sought-after applications for IEEE 802.15.1 [i.e. Bluetooth] devices, and they are most susceptible to interference' [9]. However, our approach is useful in any time-critical application.

The paper is organized as follows. In the next section, we first compute the worst-case response times ignoring interferences. Then in Section 3 we discuss how to include the effect of interferences, and in Section 4 we propose an unified model to include interferences from different ISM devices, particularly 802.11b/g, Bluetooth and Zigbee. In Section 5, we evaluate as an example the transmission of a voice stream through an ACL link under interferences. Finally, we present our conclusions.

2. WORST-CASE RESPONSE TIME

Consider a piconet composed of a master and up to seven slaves, with several SCO and ACL links. SCO packets have pre-assigned slots and always pre-empt ACL packets. Furthermore, they are never delayed, so their response times remain constant regardless of possible interferences. That is, the packet may be corrupted or not, but its response time is not affected by possible interferences. Although the behaviour of SCO links could also be evaluated using the probability that the packet is corrupted as a measure of performance, several reasons make ACL links more interesting. In coexistence scenarios, that is, situations where the main reason for packet drop is due to the interference from other devices and not due to noise or range, FEC protection will cause the Bluetooth unit to send more packets for a given amount of data, and this results in higher interferences [9]. DHx (Data High rate) ACL packets are therefore preferred since they are not FEC encoded. On the other hand, it has been shown that if ACL links are used for isochronous traffic (like voice) instead of SCO links, other ACL links like TCP behave better, with the voice quality only slightly affected [3]. For these reasons, in this paper we consider the use of ACL links to support real-time transmissions, and as a result we focus only on the computation of worst-case response times of ACL packets, while SCO packets are considered simply as a source of interference.

Therefore, consider N ($N \leqslant 7$) slaves with active ACL links. In this paper, we assume for simplicity that each ACL packet occupies a single slot. Only a single ACL link can exist between the master and every slave [21], and the polling is performed only with slaves with ACL links. Outgoing ACL packets are queued whenever they suffer delays from three possible sources:

- The polling mechanism. An ACL packet has to wait for the poll when it is the other nodes' turn.

- Pre-emption from SCO packets.
- Interferences from other ISM devices. These interferences may destroy the packets so they introduce additional delays.

In this section, we present an analysis to compute worst-case response times of ACL packets including the delays due to the first two sources of interferences, while in the following section we extend this analysis to include interferences from other ISM devices.

In our model, an SCO packet j is characterized by its period ($T_{\mathrm{SCO}j}$) and its constant transmission time (C_{SCO}). Although the Bluetooth channel is divided into slots of length $625\,\mu s$, due to the time division duplex mechanism we should count two slots per transmission. In other words, every transmission includes one slot for the packet and another one for the acknowledgement. Therefore, we consider for our discrete-time model a time unit d_slot $= 2*625\,\mu s$, so that $C_{\mathrm{SCO}} = 1$ d_slot.

On the other hand, an ACL packet i is characterized by its period (T_i),[‡] deadline (D_i) and transmission time (C_i). In what follows, we assume that ACL packets transmission times do not depend on the priority level (the Bluetooth MAC layer does not support priorities except for the priority of SCO over ACL packets). As a result, we write $C_i = C$, for all i. We also assume that packets must be received before the end of the period of the sending task (that is, $D_i < T_i$), to avoid packets from successive invocations of the sending task to delay each other.

Several intra-piconet polling algorithms have been proposed [22]. In this paper, we consider a pure round-robin (one packet per visit), because it is the most used in current Bluetooth connections [5]. Furthermore, it has been shown that PRR has a good performance at high loads [22]. Anyway, other polling schemes could also be considered in our model.

We are interested in the computation of the worst-case response time of an ACL packet that may be delayed by other ACL/SCO packets. With the PRR algorithm, we can distinguish between two cases:

- Packets queued at a given node. Since only a single ACL link can exist between the master and every slave, there is no interference on a given ACL packet m due to local (within the node) higher priority packets except for SCO packets.
- Packets queued at other nodes (ACL links between the master and the other slaves). While our reference ACL packet m is waiting for the poll, these packets are being transmitted. They can be modelled as a single 'high priority packet' with periodicity equal to the polling period T_{POLL} [17].

The interference due to packets in other nodes is given by the number of times this 'higher priority packet' with period equal to T_{POLL} is scheduled for transmission during the worst-case packet response time:

$$\left\lceil \frac{Q_m}{T_{\mathrm{POLL}}} \right\rceil (T_{\mathrm{POLL}} - C)$$

where Q_m is the queuing time of our reference ACL packet m, that is, the time packet m spends in the queue (time from being queued to the time transmission begins). In the PRR case, nodes are only allowed to transmit a single packet per visit. Therefore, $T_{\mathrm{POLL}} = N*C$. Every time this

---

[‡] If the traffic in the ACL link is not isochronous, then the period should be interpreted as the minimum time between successive packets.

'packet' interfere a given packet m, we have to wait $T_{POLL} - C = (N - 1)*C$. Therefore, the total interference on a given packet m is

$$\left\lceil \frac{Q_m}{N*C} \right\rceil (N - 1)C$$

We also have to add the periodic and fixed interference from SCO packets, which does not depend on the node they may be generated. The master can support up to three simultaneous SCO links while slaves can support two or three SCO links. Therefore, the interference from these packets can be found by calculating the number of times successive instances of SCO packets could be scheduled for transmission in front of a given ACL packet m:

$$\sum_{\forall j \in SCO_S} \left\lceil \frac{Q_m}{T_{SCOj}} \right\rceil C_{SCO}$$

where SCO_S is the set of SCO links in the piconet. We assume that the jitter (the maximum time variability between subsequent packets of an SCO link) is null. The queuing time can now be found iteratively [17]:

$$Q_m^{n+1} = \left\lceil \frac{Q_m^n}{N*C} \right\rceil (N - 1)C + \sum_{\forall j \in SCO_S} \left\lceil \frac{Q_m^n}{T_{SCOj}} \right\rceil C_{SCO} \tag{1}$$

The total response time of a given ACL packet m in the worst case is then:

$$R_m = Q_m + C \tag{2}$$

If interferences from other ISM devices are ignored, then whatever polling scheme is used collisions do not occur within a piconet. Therefore, the packet transmission time is a constant. Since we are considering single-slot packets, then $C = 1d\_slot$ of $2*625\,\mu s$ (time division duplex).

3. CONSIDERING INTERFERENCES

If interferences are considered, we say that a collision occurs when the packet is received incorrectly due to these interferences. The sender is notified of this error in the slot directly following the unsuccessful transmission using a fast-ARQ scheme [1]. The packet is then retransmitted at the next opportunity (in alternate slots) until it is successfully received. Therefore, the transmission time C cannot be considered a constant anymore. Instead, it becomes a (discrete) random variable, so C and Q_m are random variables. Therefore, a probabilistic time-demand analysis should be used analogous to that used for single-processor systems with semi-periodic tasks (tasks released periodically but with random computation times) in Reference [19]. However, this approach is very difficult and computationally expensive because we have to compute the probability density function of Q_m from the probability density function of random variable C. Instead, since we are only interested in determining the probability that a packet meets its deadline, we will use a different approach.

Let us denote by P_S the probability of successful transmission. In a piconet with ACL links, the wasted time units due to collisions correspond to a sequence of Bernoulli trials with probability of success P_S. The overhead for every collision is just a packet retransmission C. Therefore, the effect of interferences can be included in our model of Section 2 by adding a

collision overhead $E(X)$ that is a function of the number of collisions during a time period X. If SCO packets are ignored, $E(X) = k*C$, with k being the number of collisions during any time period X. However, we have to take into account that SCO packets are not retransmitted in case of collisions (in that case, the packet is simply corrupted but delivered 'as is'). Therefore, k should be computed as the number of collisions occurring in any time period X *excluding the transmission period of SCO packets*. We will take this into account in due course.

Anyway, considering co-channel interferences, Equation (1) becomes

$$Q_m^{n+1} = E(Q_m^n + C) + \left\lceil \frac{Q_m^n}{N*C} \right\rceil (N-1)C + \sum_{\forall j \in \text{SCO}_S} \left\lceil \frac{Q_m^n}{T_{\text{SCO}j}} \right\rceil C_{\text{SCO}}$$

$$= k*C + \left\lceil \frac{Q_m^n}{N*C} \right\rceil (N-1)C + \sum_{\forall j \in \text{SCO}_S} \left\lceil \frac{Q_m^n}{T_{\text{SCO}j}} \right\rceil C_{\text{SCO}} \tag{3}$$

Note that C is again a constant $C = 1$ d_slot $= 2*625\,\mu$s (for single slot packets) because all the effects of collisions are included in function $E(\ )$.

The response time of packet m with k collisions can now be computed iteratively. First we iterate Equation (3) with $k = 0$, obtaining a first result for Q_m, and through Equation (2) for R_m. Then we increment k and repeat the iterative computation to obtain new results for Q_m and R_m. We repeat this algorithm until $R_m > D_m$ (or the iteration does not converge). In this way we obtain K_m, which we define to be the maximum number of collisions for which $R_m \leqslant D_m$ (since the final value k_f makes R_m exceed the packet deadline, we have $K_m = k_f - 1$). Let Q_{MAX} be the last obtained value of Q_m, that is, the queuing time obtained through the following iteration:

$$Q_m^{n+1} = K_m*C + \left\lceil \frac{Q_m^n}{N*C} \right\rceil (N-1)C + \sum_{\forall j \in \text{SCO}_S} \left\lceil \frac{Q_m^n}{T_{\text{SCO}j}} \right\rceil C_{\text{SCO}} \tag{4}$$

and also let R_{MAX} be the last obtained value of R_m:

$$R_{\text{MAX}} = Q_{\text{MAX}} + C \tag{5}$$

so it represents the worst-case response time when the maximum tolerable number of collisions K_m occur.

Once K_m and R_{MAX} are obtained, we are now able to calculate the worst-case deadline failure probability (WCDFP) simply as the probability that more than K_m collisions occur during R_{MAX}. Since we assume that time is slotted, with a time unit d_slot $= 2*625\,\mu$s, then the probability that a collision occurs in a given time unit is $1 - P_S$ (i.e. the probability that the transmission is not successful). Therefore, we have that the probability that k collisions occur during a time period X is

$$\binom{X}{k} (1 - P_S)^k P_S^{X-k} \tag{6}$$

In our case, the time period is R_{MAX} but transmission of SCO packets should be excluded. Therefore,

$$X = R_{\text{MAX}} - \sum_{\forall j \in \text{SCO}_S} \left\lceil \frac{Q_{\text{MAX}}}{T_{\text{SCO}j}} \right\rceil C_{\text{SCO}} \tag{7}$$

Now, the WCDFP for an ACL packet m is given by

$$\text{WCDFP}_m = 1 - \sum_{k=0}^{K_m} \binom{X}{k} (1 - P_S)^k P_S^{X-k} \tag{8}$$

with X given by Equation (7).

4. PROBABILITY OF SUCCESSFUL TRANSMISSION

The analysis developed in Section 3 provides the WCDFP for an ACL packet as a function of the probability of successful transmission. In this section, we show how this probability can be obtained when there are different ISM devices interfering with our reference ACL packet.

We begin with the case of interferences due to other independent Bluetooth devices. In order to reduce interferences (among other reasons), Bluetooth uses Frequency Hopping, with a pseudo-random hopping sequence. We can then assume that a Bluetooth device transmits using randomly chosen frequencies. Therefore, there is a possibility that several independent (but interfering) Bluetooth devices coexist in the same area and that they choose the same hop carrier.

We can assume that hops are evenly distributed over 79 different frequency bands [23]. It can be shown that if M_B is the number of interfering piconets (including the reference piconet) and r is the normalized load over every piconet (assuming a homogeneous traffic, so r is the same for all the piconets) then the probability of successful transmission is [15]

$$P_S|_{B-B} = \left[2(1 - \sigma)\left(1 - \frac{r}{79}\right) + (2\sigma - 1)\left(1 - \frac{r}{79}\right)^2 \right]^{2(M_B - 1)} \tag{9}$$

where $\sigma = 366/625 = 0.5856$ is the ratio between the actual transmission time of a single slot packet and the Bluetooth slot duration. Recall that this expression is based on the fact that a successful transmission occurs if there are no collisions in two consecutive (and independent) slots: one for the packet and the other one for the acknowledgement (ACK). If the ACK is not received, the packet is retransmitted as if a collision had occurred in the packet itself. This expression can be simplified if we neglect the term $(r/79)^2 < 0.00016$:

$$P_S|_{B-B} \cong \left(1 - 2\sigma \frac{r}{79}\right)^{2(M_B - 1)} \tag{10}$$

Since we are concerned with the worst-case behaviour (in order to provide QoS or real-time guarantees), then this pessimistic approximation can be considered a lower bound on the probability of successful transmission, and also the case of fully loaded piconets should be used, that is, $r = 1$. Furthermore, although a time and frequency coincidence does not always destroy the packet, depending on the strength of the interference signal arriving to the receiver [24, 25], we also assume the worst-case: interference of just one bit is enough to destroy the whole packet.

This previous model can be easily interpreted as the probability that none of the other $M_B - 1$ piconets transmit with the same carrier than our reference piconet in two consecutive slots. The facts that the actual packet transmission time is only of duration σ slots and that independent piconets are not synchronized are taken into account through the factor 2σ. We will use this model to show how to include interferences from other independent Bluetooth devices in the computation of the worst-case response time of ACL packets. However, many other more

complex (and not so pessimistic) interference models could also be used [25], including the multi-slot case discussed in Reference [11].

Other non-Bluetooth devices may also produce interferences. The most important source of interference are 802.11 devices, and their effect on Bluetooth transmissions are well studied in the literature [10,12–14]. In this paper, we use the model described in Reference [14] which assumes the worst case that a time and frequency coincidence always destroys the packet. Let H and W be the transmission times of a single slot Bluetooth packet ($H = 366\,\mu s$) and of an IEEE 802.11 packet, respectively, and also let L be the dwell period of an IEEE 802.11 packet. Finally, if we define G as

$$G = \lceil H/L \rceil * L - W - H$$

then it can be shown that the probability of a successful Bluetooth transmission under interferences from 802.11 devices is [14][§]

$$P_S|_{B-W} = \left[\left(1 - \frac{|G|}{L} \right) \left(1 - \frac{B}{79} \right)^{\lceil H/L \rceil} + \left(\frac{|G|}{L} \right) \left(1 - \frac{B}{79} \right)^{\lceil H/L \rceil - G/|G|} \right]^{2M_W} \tag{11}$$

M_W is the number of interfering 802.11 devices, and B is the channel width of the 802.11 system, that is, $B = 22\,MHz$ for 802.11b and $B = 16.5\,MHz$ for 802.11g (we only consider direct sequence spread spectrum—DSSS systems). Different from Reference [14], we take again into account that a successful transmission occurs if there are no collisions in two consecutive (and independent) Bluetooth slots. Also note that we ignore coexistence mechanisms like adaptive frequency hopping (AFH), alternating wireless medium access (AWMA), etc. [10].

An interesting observation is that Equations (9) and (11) are essentially based on the same reasoning. Particularly, although Equation (11) was obtained in Reference [14] only for the case of interferences from 802.11 devices, in fact Equation (9) can be obtained as a particular case of Equation (11). Indeed, if we take $W = H$ (the interfering device is also a Bluetooth packet) and $\sigma = H/L < 1$ in Equation (11), we have $G = (L - 2H) < 0$. Finally, since for a Frequency Hopping system like Bluetooth $B = 1$, we obtain Equation (9) for the case $r = 1$.

The same approach could also be used for other ISM devices. For instance, an IEEE 802.15.4 (Zigbee) transmission using the ISM band will interfere with Bluetooth in approximately 3 out of the 79 hops [7, Annex E]. Usually, in a sensor network the probability of activity of every node is very low, and also the transmission time of an 802.15.4 packet is much larger than the duration of a Bluetooth slot, so we could use Equation (11) with $B = 3$ and $L \gg W \gg H$. In this case, $G = (L - W - H) > 0$. Therefore, the probability of a successful Bluetooth transmission under interferences from 802.15.4 devices could be expressed as

$$P_S|_{B-Z} = \left(1 - \frac{3}{79} \upsilon \right)^{2M_Z} \tag{12}$$

where M_Z is the number of interfering 802.15.4 devices and $\upsilon = (W + H)/L \cong W/L$ can be interpreted as the probability of activity of an IEEE 802.15.4 node. Recall that in a typical sensor network $M_Z \gg$ and $\upsilon \ll$.

[§] In case the M_W 802.11 devices have different values for W and L, that is, the traffic is not homogeneous, then obviously Equation (11) becomes a product of M_W terms.

To sum up, we have shown that Equation (11), originally developed only for the case of interferences from 802.11 devices, provides us with a generic model that can be used to model interferences from other ISM devices. Redefining parameters like B, H, L or W, we can obtain the probability of a successful Bluetooth transmission under interferences from different ISM devices. In the general case, the product of the $P_{S|B-X}$ terms should be used. For instance, if 802.11, Zigbee, and independent Bluetooth devices are present, then:

$$P_S = \prod_{X=\{B,W,Z\}} (P_S|_{B-X})$$

An important consideration is that these models assume the worst case that a time and frequency coincidence always destroys the packet. Although we think they are still useful for the computation of worst-case, real-time guarantees, we should be careful because many of their results may be too pessimistic. For instance, IEEE 802.15.4 networks are tailored for applications with low power, so the probability of interferences due to these devices is relatively low even in case of time and frequency coincidence. In other words, parameters M_X should not be interpreted as nodes operating in the same area, but as nodes that have sufficient power to cause a collision.

5. EXAMPLE: VOICE OVER ACL LINKS UNDER CO-CHANNEL INTERFERENCE

The analysis developed in this paper is useful for any time-critical application. Only for illustration purposes, let us consider as an example the transmission of a voice stream in several configurations. First, consider a configuration composed of a piconet of 3 slaves, where 2 slaves use ACL links. We also assume that the other node is using an SCO link in order to show how our model is able to include this additional source of interference. However, in order not to monopolize the bandwidth we assume that this SCO link uses HV3 packets, that is, one-slot packets that are sent every six slots.

We assume that one of the ACL links is used to carry a voice stream. Voice can be implemented as 8 kbps coded speech, with voice packets of 20 bytes every 20 ms [3]. DH1 packets (28 bytes maximum) can be used when carrying this traffic over ACL links. We assume that the deadline of these packets is equal to the time gap between arrival of two packets, that is, $T_i = D_i = 20$ ms.

Now, let us apply the model described in Sections 2–4 to compute the WCDFP of these voice packets in the presence of interferences from the other ACL and SCO links of the piconet, as well as from other independent Bluetooth devices. In this configuration, 2 slaves use ACL links so $N = 2$. Also, $C = C_{SCO} = 1$ d_slot because only DH1 and HV3 packets are used. Finally, $T_{SCO} = 3$ d_slots for HV3 packets, so Equation (3) becomes:

$$Q_m^{n+1} = k + \left\lceil \frac{Q_m^n}{2} \right\rceil + \left\lceil \frac{Q_m^n}{3} \right\rceil$$

We now iterate with $k = 0$, and obtain a final value $Q_m = 2$ d_slots, and using Equation (2) $R_m = 3$ d_slots $= 3.75$ ms, which is less than the deadline $D_m = 20$ ms. We then increase k, and repeat the process. This way, we obtain a final value for $K_m = 2$, $Q_{MAX} = 12$ d_slots and $R_{MAX} = 13$ d_slots $= 16.25$ ms, which meets the deadline $D_m = 20$ ms. If more than two collisions occur then this timing requirement is not met.

Now, X is given by Equation (7):

$$X = R_{\text{MAX}} - \left\lceil \frac{Q_{\text{MAX}}}{3} \right\rceil = 9$$

and P_S is given by Equation (10) (with $r = 1$ in order to consider the worst case) since in this example we only consider the interference of Bluetooth devices (note that if other ISM devices are present, the product of P_S given by Equations (10)–(12) should be used):

$$P_S = \left(1 - 2\sigma \frac{r}{79} \right)^{2(M_B - 1)} = (0.985)^{2(M_B - 1)}$$

Then, the WCDFP of the ACL link carrying voice can be obtained through Equation (8) as a function of the number of interfering piconets M_B:

$$\text{WCDFP}_{2\text{ACL}+\text{SCO}} = 1 - \sum_{k=0}^{2} \binom{9}{k} (1 - (0.985)^{2(M_B - 1)})^k (0.985)^{2(M_B - 1)(9 - k)} \tag{13}$$

Let us now consider another configuration with the only difference of using 3 ACL links instead of 2 ACL links and 1 SCO link. Note that the comparison is fair since the bandwidth used by the SCO link in the previous example and that used by our third ACL link in this case is the same, that is, one packet every six Bluetooth slots. This comparison is interesting because several authors have reported that SCO links are redundant since voice can be carried more efficiently using ACL links [3, 5]. In this case, $K_m = 5$, $Q_{\text{MAX}} = 15\text{d\_slots}$ and $R_{\text{MAX}} = X = 16\text{d\_slots} = 20\,\text{ms}$, which exactly meets the deadline. Therefore:

$$\text{WCDFP}_{3\text{ACL}} = 1 - \sum_{k=0}^{5} \binom{16}{k} (1 - (0.985)^{2(M_B - 1)})^k (0.985)^{2(M_B - 1)(16 - k)} \tag{14}$$

Figure 1 compares the analytical results for the two cases (Equations (13) and (14)). As expected, it can be observed that the probability of deadline failure of the ACL link carrying voice increases as the number of independent but interfering piconets grows. If we assume that an acceptable missed deadline ratio for voice traffic is 10% [26], then the maximum number of
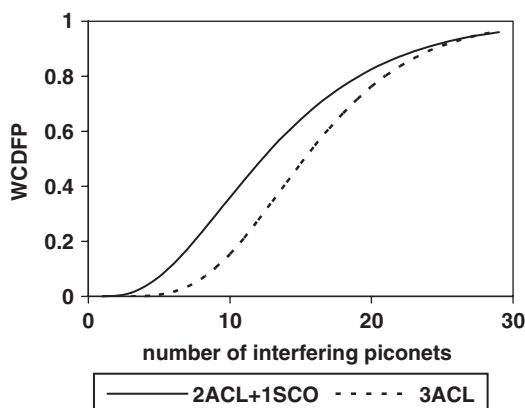

Figure 1. WCDFP of an ACL link carrying voice in a piconet composed of 3 slaves. Two configurations are shown: 2 ACL links and 1 SCO link (solid) and 3 ACL links (dashed).
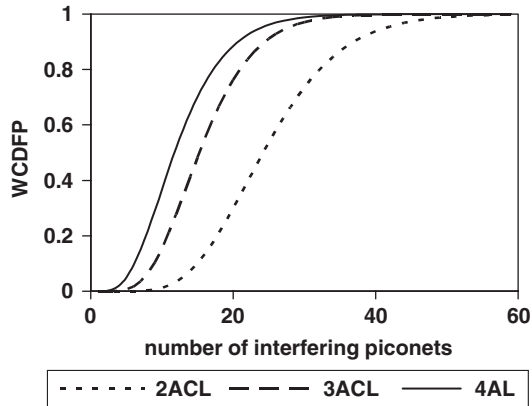
Figure 2. WCDFP of an ACL link carrying voice in the presence of 2, 3 and 4 ACL links.

Table I. Data for Figure 2.

N	K_m	R_{MAX}
2	7	15d_slots
3	5	16d_slots
4	3	13d_slots

independent piconets coexisting in the same area should be lower than 6 (in case 2 ACL links and 1 SCO link are used) or 9 (in the case of 3 ACL links). Note that the WCDFP of the ACL link carrying voice degrades if another slave in the piconet uses an SCO link instead of an ACL link, so that the maximum allowable number of interfering piconets decreases. This result analytically confirms other results reported in the literature, obtained through simulation, in the sense that ACL links behave better if SCO links are not used [3].

As a final example, we present in Figure 2 how the WCDFP of an ACL link carrying voice behaves as the number of other ACL links in the piconet increases. In this case, no SCO links are used. The configurations presented in Figure 2 and their corresponding data are shown in Table I (N is the total number of ACL links).

It can be observed that when the number of active ACL links in the piconet increases, the WCDFP of the ACL link carrying voice increases very rapidly. If we again assume an acceptable missed deadline ratio of 10%, then the maximum allowable number of independent piconets coexisting in the same area is 14 for $N = 2, 8$ for $N = 3$ and 6 for $N = 4$.

6. CONCLUSIONS

In this paper, the worst-case delay of ACL packets is analytically derived including preemption from SCO packets, the polling mechanism and the effect of interferences from other devices. The probability that this delay exceeds the packet deadline (the so-called *worst-case deadline failure probability—WCDFP*) is obtained as a function of the probability of successful transmission.

We then obtain a generic model for the computation of this probability under interferences from different ISM devices. This approach allows us to evaluate the WCDFP as a function of the number and type of interfering devices. Our model may be useful to analytically predict how soft real-time applications behave when several active ISM devices are used in a common area, including other nodes in the piconet, active SCO/ACL links and independent ISM devices (Bluetooth, 802.11b/g and Zigbee devices).

As an example, we then show how to use our model to evaluate the behaviour of an ACL link carrying voice in the presence of other active SCO/ACL links in the piconet and of a number of independent but interfering Bluetooth devices. The maximum allowable number of interfering devices is obtained for various configurations.

REFERENCES

1. Haartsen JC. The Bluetooth radio system. *IEEE Personal Communications* 2000; **7**:28–36.
2. Famolari D, Anjum F. Improving simultaneous voice and data performance in Bluetooth systems. *Proceedings of Globecom 2002*, 2002; 1810–1814.
3. Kapoor R, Chen L, Lee Y, Gerla M. Bluetooth: carrying voice over ACL links. *Proceedings of the IEEE MWCN 2002*, 2002; 379–383.
4. Hamdi M, Lee FL. Providing deterministic packet delays and packet losses in multimedia wireless networks. *Wireless Communications and Mobile Computing* 2000; **3**:3–22.
5. Chawla S, Saran H, Singh M. QoS based scheduling for incorporating variable rate coded voice in Bluetooth. *Proceedings of the IEEE International Conference on Communications*, 2001; 1232–1237.
6. Chen WP, Hou JC. Provisioning of temporal QoS in Bluetooth networks. *Proceedings of the IEEE MWCN'02*, Stockholm, Sweden, 2002; 389–393.
7. IEEE Std. 802.15.4. Part 15.4: wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs), October, 2003.
8. http://www.usb.org/wusb/home
9. Cascado D *et al*. Performance analysis of single-slave Bluetooth piconets under cochannel interference. *Proceedings of IEEE PIMRC'2004*, Barcelona, Spain, 2004; 954–958.
10. IEEE Std. 802.15.2. Part 15.2: coexistence of wireless personal area networks with other wireless devices operating in unlicensed frequency bands, August 2003.
11. Lin TY, Tseng YC. Collision analysis for a multi-Bluetooth picocells environment. *IEEE Communications Letters* 2003; **7**–**10**:475–477.
12. Howitt I. Bluetooth performance in the presence of 802.11b WLAN. *IEEE Transactions on Vehicular Technology* 2002; **51**:1640–1651.
13. Golmie N, Mouveaux F. Interference in the 2.4 GHz ISM band: impact on the Bluetooth access control performance. *Proceedings of the International Conference on Communication ICC-2001*, June 2001; 2540–2545.
14. Feng W, Arumugam N, Krishna GH. Performance of a Bluetooth piconet in the presence of IEEE 802.11 WLANs. *Proceedings of the 13th IEEE PIMRC*, September 2002; 1742–1746.
15. El-Hoiydi A, Decotignie JD. Soft-deadline bounds for two-way transactions in Bluetooth piconets under co-channel interference. *8th IEEE International Conference on Emerging Technologies and Factory Automation*, October 2001; 143–150.
16. Sevillano JL *et al*. Statistical Qos guarantees in Bluetooth under co-channel interference. *10th IFIP International Conference on Personal Wireless Communications, PWC'05*, Colmar, France, August 2005.
17. Tindell KW, Burns A, Wellings AJ. Guaranteeing hard real-time end-to-end communications deadlines. *Technical Report RTRG/91/107*, University of York, U.K., December 1991.
18. Atlas A, Bestavros A. Statistical rate monotonic scheduling. *IEEE Real-Time System Symposium*, Madrid, 1998.
19. Tia TS *et al*. Probabilistic performance guarantee for real-time tasks with varying computation times. *IEEE Real-Time Technology and Applications Symposium*, Chicago, U.S.A., May 1995; 164–173.

20. Navet N, Song YQ, Simonot F. Worst-case deadline failure probability in real-time applications distributed over controller area network. *Journal of Systems Architecture* 2000; **46**:607–617.
21. Bluetooth Special Interest Group. Specification of the Bluetooth system—Core vol. 1 V1.1, February 2001; 1084.
22. Capone A, Gerla M, Kapoor R. Efficient polling schemes for Bluetooth picocells. *Proceedings of the IEEE International Conference on Communication ICC-2001*, June 2001; 1990–1994.
23. Haartsen JC. The Bluetooth radio system. *IEEE Personal Communications* 2000; **7**:28–36.
24. Cascado D *et al*. Including propagation effects on packet error rate of Bluetooth packets. *International Workshop on Wireless Circuits and Systems*: *WoWCAS 2004*, Vancouver, Canada, May 2004.
25. Howitt I. Mutual interference between independent Bluetooth piconets. *IEEE Transactions on Vehicular Technology* 2003; **52**:708–718.
26. Kostas TJ *et al*. Real-time voice over packet-switched networks. *IEEE Network* 1998; **12**:18–27.