Comparing Bluetooth HDP and SPP for Mobile Health Devices

Jad Noueihed*, Robert Diemer*, Samarjit Chakraborty* and Stefanie Biala†
*Institute for Real-Time Computer Systems, TU Munich
†Vodafone Group R&D Germany, Munich
eMail: diemer@tum.de

Abstract—**The Bluetooth SIG recently released the Health Device Profile (HDP) in an effort to standardize health device communication using Bluetooth technology. HDP uses the IEEE 11073-20601 Data Exchange Protocol as the transport content. The same traffic can be sent using the popular Serial Port Profile (SPP), but this profile is not configured for health device communication. HDP's strict configurations and health device-specific requirements give the impression that it would have a reduced performance in comparison to SPP. In this paper we compare HDP with SPP by analyzing multi-rate data transmission in the context of a cardiovascular monitoring application. In particular, we model the expected number of transmissions and packet loss incurred by the two profiles. Our results show that in contrast to popular belief the transmission energy is similar for both profiles, but HDP offers the advantage of plug-and-play interoperability.**

I. Introduction

Mobile health devices capable of measuring health-related data, in association with mobile phones are enablers for telemonitoring applications, which can greatly benefit healthcare systems. However, these devices often suffer from interoperability problems, for which there are several ongoing standardization efforts. The IEEE is working on the 11073 family of standards aimed at tackling interoperability at the application layer and at other levels of the communication protocol stack. The Bluetooth SIG recently released the Health Device Profile (HDP), which standardizes the way health devices communicate using Bluetooth technology. Since HDP is a new profile, little is known about its performance overheads (i.e., the price to be paid for interoperability). The aim of this paper is to analytically compare HDP with the widely used Serial Port Profile (SPP) through a representative health monitoring case study.

Bluetooth works in an unlicensed portion of the communication spectrum, so it is susceptible to varying levels of interference. An increased interference would lead to an increased bit error rate and consequently packet error rate. For reliability, packets are retransmitted leading to an increased consumption of energy. In this paper we model the effect of interference on transmissions in HDP and SPP, and the effect of transmitting data at multiple rates as is typical with health monitoring applications. Our results indicate that in contrast to popular belief, the overheads of ensuring interoperability in HDP does not result in any performance degradation in comparison with SPP.

This paper is structured as follows. An overview of the relevant portions of the Bluetooth standard is given in Section II. Section III describes our setup and the data traffic model. In Section IV we analytically compare HDP and SPP. Finally, we outline some directions of future work in Section V.

II. Bluetooth

Bluetooth [1] is a low-power short-range wireless technology developed as a replacement for serial cables. It operates in the unlicensed Industrial, Scientific, and Medical (ISM) band from 2402 MHz to 2480 MHz. We assume that interference is generated by other Bluetooth devices, although it could also be due to devices like microwave ovens and 802.11 networks. Communicating Bluetooth devices form a piconet consisting of a single master and up to seven active slaves. Communication is done in a point-to-point fashion between the master and a slave, which can only transmit after being polled.

A. Controller

The controller is composed of the baseband and the radio. It carries out all the link management functionality, using the Link Manager Protocol. It is responsible for packetizing data, channel coding and decoding, and determining the operating frequency.

Channels: A channel is defined by a pseudorandom frequency hopping sequence, slot timing, and an access code. Logical transports represent a connection between the master and a slave. The transports can either be asynchronous (ACL) or synchronous (SCO). Logical links used on top of the transports represent a data connection between the master and the slave.

Packets: A Bluetooth baseband packet is composed of three parts: an access code, a payload, and a packet header. The payload consists of a payload header and a higher layer data unit. The header carries link control information and an error check. Several packet types are defined by the standard. Each type of logical transport uses a different packet type for data transmission. Common packets are used by all transports, e.g., for managing the hopping sequence or transmitting some connection status information. These packets will be referred to as control packets. The POLL packet is used by the master to poll the slaves. The NULL packet is used to acknowledge data packets. The control packet does not carry any user data. For each data carrying packet type there are versions that span 1, 3, or 5 time slots in combination with using different modulation schemes.

ARQ: The Bluetooth baseband uses an automatic repeat request (ARQ) to retransmit erroneous baseband packets. The ARQ mechanism applies to both ACL packets types, Data–Medium Rate and Data–High Rate (DMx resp. DHx, x denotes number of used time slots). ARQ tries to transmit packets until it either succeeds or a timeout occurs, leading to the flushing of the entire L2CAP packet (described below).

B. L2CAP

According to the standard [2], the Logical Link Control and Adaptation Protocol (L2CAP) is a data link layer that provides connection-oriented and connectionless services — like flow control or retransmission — to upper layer protocols or applications.

The L2CAP layer can be configured to operate in more desired fashion by configuring some of the parameters. The flush timeout specifies the amount of time to transmit a packet before it is dropped by the controller. The retransmission timeout specifies the amount of time L2CAP will wait for the acknowledgement for an information frame before it retransmits the frame.

The mode of the L2CAP channel describes its configuration, and how it is expected to behave. There are different types of frames that are used by the modes. The *Basic mode* does not support any retransmissions, and thus does not offer any reliability. The *Streaming mode* is used with isochronous data. The frames are numbered but they are neither acknowledged nor retransmitted.

C. Profiles

A Bluetooth profile describes how devices communicating over Bluetooth interact, by specifying the configuration of the channel and the sequence of data exchange needed to establish the channel. It specifies the dependencies on other protocols and profiles, and the manner in which connection is established and configured. In this paper we compare the health device and the serial port profiles.

Health Device Profile (HDP) [3]: HDP is used to describe how health devices interact over Bluetooth. This profile uses the Multi-Channel Adaptation Protocol (MCAP) to establish communication channels. A control channel is used to establish and manage data channels. The data channels can be paused and restarted with minimal overhead and delay, by retaining the state of the connection before pausing it. This fast reconnection of the data channels allows power saving by allowing the controller to be placed longer in a low-power mode. Authentication and encryption of the channels are mandatory. HDP also specifies the L2CAP modes as either *Enhanced Retransmission* or *Streaming*. The data carried is IEEE 11073-20601 Optimized Exchange Protocol traffic.

Serial Port Profile (SPP) [4]: SPP is widely-used to replace wired serial ports. It makes use of the RFCOMM (Radio Frequency Communication) protocol [5], which is a channel multiplexing protocol running on top of L2CAP that can also provide RS-232 controls. SPP would emulate an asynchronous serial port. No reliability is offered by RFCOMM, so it has to be either offered by the lower layers or managed by the application.

III. Traffic Data Model

A. Setup

For our study, we consider a monitoring application, which keeps track of the heart rate and the walking speed of a patient/user. The goal is to allow patient monitoring over an extended time period. To establish the piconet, a health device searches for a mobile device to connect to and associate with. The mobile device assumes the role of the piconet master so that it can schedule the different health devices that connect to it. The aim is to have a general overview of the heart rate and speed measurements. It is still desired to be able to capture a more detailed view of periods during which the measurements are not within the normally expected range. In general, when measured values are stable, a mean value could be transmitted, otherwise all data is necessary when a more detailed view is desired.

B. IEEE 11073 and Measurement Data Objects

A major reason for using HDP is to have application-level interoperability and standardized usage. To ensure this, the data representation must adhere to the IEEE 11073 family of standards, specifically the IEEE 11073-20601 Optimized Exchange Protocol in conjunction with device specialization standards.

The IEEE 11073-20601 standard [6] can be broken down into three components: the domain information model for data representation, the service model for access definition, and the communication model. The device specializations are IEEE 11073-104xx standards that define the requirements for particular classes of health devices. These standards define the objects and access methods specific to a particular device. The IEEE 11073-10441 Device Specialization–Cardiovascular Fitness and Activity Monitor [7] is used for the monitoring of physical activity and the physiological response to it. Some special extensions necessary for the monitoring application have also been added to the standard.

C. Rate Controller

The rate controller we have used is an application that throttles the amount of information being transferred over the air by changing the level of information abstraction. Changing this level of detail is expected to save energy; this is done by reducing the volume of traffic, and by placing the transceiver in a low-power mode for a longer period of time while operating in the overview mode.

In this work, we use a simple rate controller. It is only capable of managing the data and not the configuration of the Bluetooth communication. It can choose between two levels of detail — an *overview* or *episodic mode* and a *detailed view* or *streaming mode*. The heart rate monitor outputs the inter-arrival time between every two consecutive heartbeats. The accelerometer produces a periodic stream of three mutually orthogonal accelerations sampled at 128 Hz. From these measurements, the rate controller can compute the mean heart rate and speed over a time period.The output of the rate controller is data adhering to the IEEE 11073 standards, plus the private/special extensions. This data is then transmitted using HDP. When in overview or episodic mode, a single episodic packet containing the average heart rate and the average speed is sent every $T_{episodic}$ time units. When in the detailed or streaming mode, a streaming packet containing the periodically available accelerometer readings is sent every $T_{streaming}$ time units. The heartbeat inter-arrival time is sent in the streaming packet when available in that mode. Here, the values of $T_{episodic}$ and $T_{streaming}$ are set at 5 sec and 0.125 sec respectively (more frequent packets in the streaming mode).

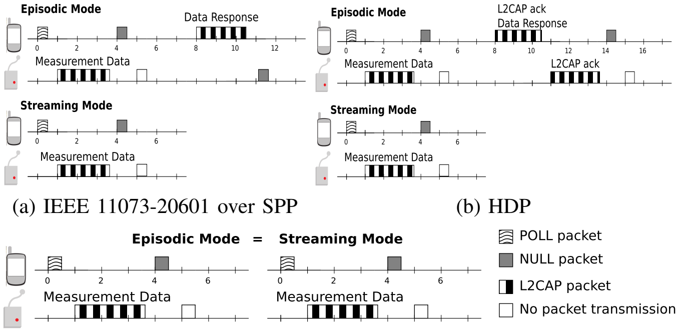
D. Assumptions

We assume the channel model to be a shared medium such that concurrent transmissions on the same channel lead to packet collision, which results in irrecoverable packets.

We compare two different setups. A setup describes the application traffic being communicated, in addition to the background traffic that constitutes the interference to the application traffic. Our setups are classified according to the background noise. In the first setup, the background traffic is the same as the application traffic. In the second, the background traffic is voice traffic. The application traffic is the output of the rate controller being transmitted over a Bluetooth channel. The data can be represented using the IEEE 11073-20601 protocol or a proprietary one [8], and it could be transported using HDP or SPP. Three combinations are used for analysis. The first is IEEE 11073-20601 data transported using HDP. The second is also IEEE 11073-20601 data, but it is transported using SPP. The third kind uses the proprietary protocol for representation, transported using SPP.

DH3 packets are the only asynchronous data packets that are used. DH3 packets do not use any forward error correction (FEC), which removes the effect that coding has on performance. The maximum frame size is 183 bytes, which can fit an application SDU (Service Data Unit) within a single frame, eliminating the effect of segmentation. These packets are also subject to the ARQ mechanism. The transmit power level is set at the maximum value for all piconets.

L2CAP retransmissions are limited to two attempts. Each L2CAP frame has to be acknowledged before another one can be transmitted. To avoid transmitting expired data, a flushout time is set at 100 ms, which is long enough to attempt several retransmissions by the ARQ, but is shorter than the packet arrival rate while the rate controller is in streaming mode. The retransmission timer is set at 500 ms, which is long enough for the packet and the acknowledgement to be transmitted. HDP is required to use the Enhanced Retransmission and Streaming modes. It is assumed that SPP uses the Basic mode.



(a) IEEE 11073-20601 over SPP

(b) HDP

(c) Proprietary Protocol

Fig. 1. Traffic model for the HDP, IEEE 11073-20601 data over SPP and the proprietary protocol. The transmission of only one packet is shown.

E. The Traffic Models

The traffic model describes the packets that are sent, including the POLL, the NULL, and all acknowledgements. The traffic is classified into four categories according to how it is generated. Three of them are the different options for the data traffic output of the rate controller: HDP traffic, IEEE 11073-20601 over SPP, and the proprietary protocol over SPP. The fourth is the voice traffic. Each traffic generator takes into account the requirements of the communication standards, and generates the appropriate packets to fulfill this purpose. Fig. 1b, 1a, and 1c show the packets that are transmitted between the health devices and a mobile phone. The major difference between the three protocols lies in the manner in which the transmission of measured data is acknowledged. Packet generation is deterministic with the arrival rate and length determined by the rate controller. Voice packets are assumed to be HV2 packets. These packets use a FEC with rate 2/3, and occupy half the total number of slots.

IV. Comparison and Results

A. Interference

Interference is experienced by a piconet when another piconet simultaneously transmits data on the same frequency channel used by the first. It is important to note that the different devices of the same piconet are treated as a single source of interference. The slaves are scheduled by the master, and thus the different devices of the piconet do not transmit at the same time. It is the frequency hopping pattern that determines the interference a piconet causes to another. The interference analyzed applies to the reference piconet as a whole and not the individual devices.

Related work: There has been previous work on analyzing the effect of piconet interference, but none considers a traffic model specific to health device monitoring. The analysis in [9] gives an upper bound on the packet error rate, but it assumes only single-slot packets, with packets being transmitted back to back. The work in [10] uses models of the wireless channel to find the signal to interference and noise ratio. This is then used to find the outage probability due to interference from other piconets. The analysis in [11] considers the availability of multi-slot packets and computes the probability of successful transmission using a Markov-chain model based on the operation of the ARQ scheme. The probability of successful transmission was also computed in [12]. This work considers the guard time between the slots and uses a model based on slot delimiters to compute the probability.

B. Collision Analysis for Background Traffic

In the presence of interfering Bluetooth piconets, collisions occur, leading to an increased number of retransmissions and consequently more energy consumption. Analytic expressions are derived for determining the probability that a device successfully transmits a baseband packet based on the number of interferers and the traffic they transmit. These measures deal with the number of transmissions and packet loss.

Baseband Packet Transmitting Probability: To successfully transmit a baseband packet it should not collide with traffic of interfering piconets. Two cases can be distinguished. First, if the device chooses to transmit on the same frequency f_i an interferer is using at the time, no collision would occur if the time offset is right. This means that if the device starts transmitting during the silent periods of the interferer, the packet will be successfully transmitted. When the interferer decides to transmit it would hop to a different frequency f_j while the device continues transmission on f_i. Second, if the device chooses a frequency f_i different from the frequency f_j an interferer is operating on, a collision would occur if the interferer hops to frequency f_i to transmit. A successful transmission occurs if the interferer continues to operate on f_j or hops to a frequency f_k different from f_i. To obtain this probability, the following terms are defined:

- $P_{epi} = Pr\{\text{being in episodic mode}\}$
- $P_{str} = Pr\{\text{being in streaming mode}\} = 1 - P_{epi}$
- $P_{fs} = Pr\{\text{choosing the same freq.}\} = \frac{1}{79}$
- $P_{fd} = Pr\{\text{choosing a different freq.}\} = \frac{78}{79}$
- $P'_{fd} = Pr\{\text{choosing a diff. freq. given the prev. freq.}\} = \frac{77}{78}$
- $R_{epi} = Pr\{\text{interferer transmitting a packet while in epi. mode}\}$
- $R_{str} = Pr\{\text{interferer transmitting a packet while in str. mode}\}$
- $L_{slot} = \text{length of a slot} = 625$
- $L_{meas\_epi} =$ length of a measurement packet during epi. mode
- $L_{meas\_str} =$ length of a measurement packet during str. mode
- $L_{resp} =$ length of a response packet during epi. mode
- $L_{ack} =$ length of an L2CAP ack. $= 238$
- $L_{control} =$ length of a control packet$= 126$
- $m =$ number of interfering piconets

The length of a packet is measured in symbol periods. R_{epi} and R_{str} are computed based on the arrival rate of packets from the controller and the expected number of transmissions in worst case settings. They are used to show whether an interfering piconet has packets to transmit.

The probability of a device successfully transmitting a baseband packet is given by the probability that no collisions occur with the traffic of any of the other piconets:

$$\begin{aligned}
&Pr\{\text{Packet successfully transmitted}\}\\
&= Pr\{\text{Packet does not collide with traffic of other piconets}\}\\
&= \sum_{i=0}^{m} \binom{m}{i} \Big[Pr\{\text{no collision when choosing same freq}\}\Big]^{i}\\
&\quad \times \Big[Pr\{\text{no collision when choosing a different freq}\}\Big]^{m-i}
\end{aligned}$$

The probability of having no collisions according to the operating frequency is a function of the packet type, the packet length, the probability of being in episodic mode, and the probability of the interferer transmitting a packet. We can express the probability of not having a collision while using the same frequency f_s as

$$\begin{aligned}
&Pr\{\text{no collisions when choosing the same frequency}\}\\
&= Pr\{packet, f = f_s\} = Pr\{freq = f_s\}Pr\{packet|f = f_s\}
\end{aligned}$$

where $packet$ indicates the probability of successfully transmitting a baseband packet. We can expand the conditional probability and show, that the previous expression results in:

$$\begin{aligned}
&Pr\{packet|f = f_s\}\\
&= P_{fs}P_{epi}R_{epi}Pr\{packet|f = f_s, state = R_{epi}\}\\
&+ P_{fs}P_{epi}(1 - R_{epi})\\
&+ P_{fs}P_{str}R_{str}Pr\{packet|f = f_s, state = R_{str}\}\\
&+ P_{fs}P_{str}(1 - R_{str})
\end{aligned}$$

Here $state$ indicates the probability that a packet is being transmitted in the interfering piconet. A similar expression can be derived when the device chooses a frequency f_d different from the one the interfering piconet is using, by substituting P_{fd} with P_{fs} and f_d with f_s. The four conditional probabilities $Pr\{packet|f_{s,d}, state = R_{epi}, R_{str}\}$ depend on the packet being transmitted and the associated conditions. It is determined for each packet type that is used in data exchange. These probabilities are given below for HDP episodic measurement packets generated with application-type background traffic:

$$\begin{aligned}
&Pr\{packet|f = f_s, state = R_{epi}\}\\
&= \left[1 - \frac{L_{meas\_epi} + L_{resp} + L_{ack} + 3L_{control}}{14L_{slot}}\right]\\
&Pr\{packet|f = f_s, state = R_{str}\}\\
&= \left[1 - \frac{L_{meas\_str} + 2L_{control}}{6L_{slot}}\right]\\
&Pr\{packet|f = f_d, state = R_{epi}\}\\
&= \left[\frac{3(3L_{slot} - L_{meas\_epi})}{14L_{slot}}\right.\\
&\quad \left. + \left(1 - \frac{3(3L_{slot} - L_{meas\_epi})}{14L_{slot}}\right) P'_{fd}\right]\\
&Pr\{packet|f = f_d, state = R_{str}\}\\
&= \left[\frac{3L_{slot} - L_{meas\_epi}}{6L_{slot}} + \left(1 - \frac{3L_{slot} - L_{meas\_epi}}{6L_{slot}}\right) P'_{fd}\right]
\end{aligned}$$

Expected number of transmissions: The ARQ mechanism handles the retransmission of ACL packets when there is a transmission error. This would apply to episodic and streaming measurement packets, response packets, and L2CAP acknowledgements. This is shown as a function of the number of interfering piconets only for episodic measurement packets, but the same method may be used in all the other cases. Fig. 1b shows that the transmission of an episodic measurement packet requires the successful transmission of a POLL packet, the DH3 packet containing the data, and a NULL packet. Failure with any of these packets would trigger the process to start all over again.

Let P_{succ} be the probability of successfully transmitting an application packet in a single trial without triggering the ARQ mechanism. For the measurement packet we have:

$$\begin{aligned} P_{succ} =& Pr\{\text{POLL transmission success}\} \\ &\times Pr\{\text{measurement packet transmission success}\} \\ &\times Pr\{\text{NULL transmission success}\} \end{aligned}$$

Transmission trials are considered statistically independent of one another. This is a reasonable assumption because the piconets are independent of each other, with each generating its own hopping sequence. Thus, collision events are independent of one another. The probability distribution of the random variable N_{TX} (the number of transmissions) is given by

$$p_{N_{Tx}}(n) = (1 - P_{succ})^{n-1} P_{succ}$$

The expected value of the number of transmissions, N_{Tx}, is

$$E\{N_{Tx}\} = \sum_{n=1}^{\infty} n \cdot p_{N_{Tx}}(n) = \frac{1}{P_{succ}}$$

Probability of packet flush: The flush timer is started when a packet enters the transmit buffer of the controller. If a timeout occurs, the packet is flushed. Each data packet requires a certain number of slots for a successful transmission. ARQ retransmissions can occur a finite number of times, denoted by N_{Tx_thresh}, before a timeout occurs. N_{Tx_thresh} is determined by

$$N_{Tx\_thresh} = \left\lfloor \frac{\text{flush timeout}}{\text{successful packet transmission duration}} \right\rfloor$$

The packet flush probability is given by

$$\begin{aligned} Pr\{\text{packet flush}\} &= Pr\{N_{Tx} > N_{Tx\_thresh}\} \\ &= 1 - \sum_{n=1}^{N_{Tx\_thresh}} p_{N_{Tx}}(n) \\ &= (1 - P_{succ})^{N_{Tx\_thresh}} \end{aligned}$$

If we consider an episodic measurement packet which requires 6 slots to successfully transmit,

$$N_{Tx\_thresh} = \left\lfloor \frac{100ms}{625\mu s/slot \times 6slots} \right\rfloor = 26$$

$$Pr\{\text{packet flush}\} = (1 - P_{succ})^{26}$$

Probability of L2CAP retransmission: An L2CAP retransmission occurs when the retransmission timer has a timeout. This can occur if the original packet is flushed or if the acknowledgement for it is flushed. Thus, no retransmission occurs when both packets are successfully transmitted.

$$\begin{aligned} P_{RTX} =& Pr\{\text{L2CAP retransmission}\} \\ =& 1 - Pr\{\text{successful packet delivery}\} \\ & \times Pr\{\text{successful ack delivery}\} \\ =& 1 - \left[1 - (1 - P_{succ\_packet})^{N_{Tx\_packet}}\right] \\ & \times \left[1 - (1 - P_{succ\_ack})^{N_{Tx\_ack}}\right] \end{aligned}$$

where N_{Tx_packet} and N_{Tx_ack} are the threshold values for the number of retransmissions of measurement packets and acknowledgments respectively. The probability distribution $p_{N_{RTX}}$ of the random variable N_{RTX}, the number of L2CAP retransmissions is given by

$$p_{N_{RTX}}(n) = (P_{RTX})^n (1 - P_{RTX})$$

Probability of packet loss: Packets that are not retransmitted by L2CAP are lost when they are flushed. This applies to all streaming packets and packets that use SPP. For these packets

$$Pr\{\text{packet loss}\} = Pr\{\text{packet flush}\}$$

Episodic measurement packets and the responses, which use HDP, are retransmitted by L2CAP. These packets are lost when the L2CAP retransmissions fail. The probability of packet loss is given by

$$Pr\{\text{packet loss}\} = 1 - Pr\{N_{RTX} \leq 2\} = 1 - \sum_{n=0}^{2} p_{N_{RTX}}(n)$$

For SPP the loss probabilities are the flushing probabilities. For HDP, the loss probabilities are the L2CAP retransmission failure probabilities.

C. Analysis - Numerical Results

Expected number of transmissions by ARQ: The influencing factors on the required number of transmissions by the ARQ mechanism is the length of the packet being sent, the total number of packets (including the POLL and NULL packets) required to transmit it, and the number of interfering piconets.

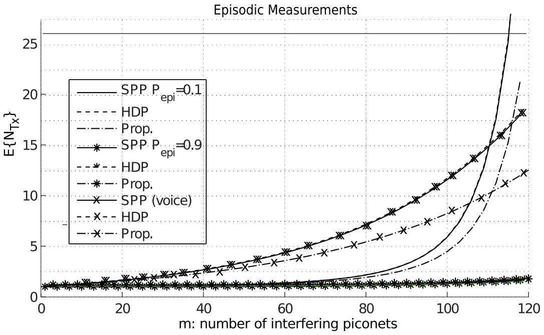
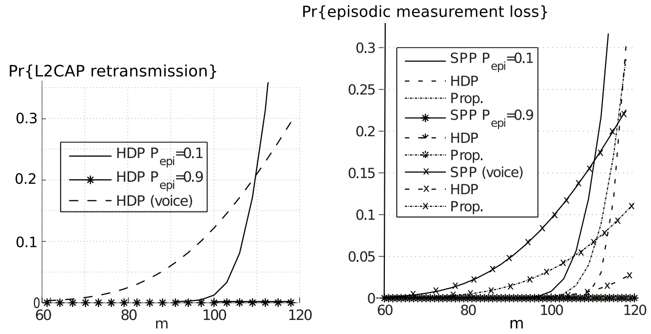


Fig. 2. $E\{N_{Tx}\}$ for episodic measurement packets for values of P_{epi} equal to 0.1 and 0.9. The horizontal line shows N_{Tx_thresh} beyond which packets are flushed.

For voice-type interference, the number of transmission attempts increase with the number of interferers; Fig. 2 shows, that it does not exhibit the constant linear portion found in application-type background traffic. This is because HV2 voice packets are not covered by the ARQ scheme. The background voice traffic is determined by the number of the interferers only. Application traffic is retransmitted by the ARQ scheme. If the background traffic is of application-type, as the interference increases, the number of retransmissions also increase. This positive feedback cycle causes a steep exponential increase in the transmission attempts, which would cause the link to break down as Fig. 2 illustrates. Finally, IEEE 11073-20601 performs roughly the same irrespective of whether it uses HDP or SPP. However, a difference is seen between using IEEE 11073-20601 and the proprietary protocol used to represent data. The proprietary protocol has the advantage of requiring smaller packet sizes. A smaller packet size increases the probability of successful transmission, thus decreasing the need for retransmissions. The influence of the packet size is also seen between episodic packets and streaming ones. The smaller episodic packets require a lower average number of transmissions.

Probability of L2CAP retransmission: Fig. 3a shows the probability of having an L2CAP retransmission for HDP episodic measurements, determined by the flush probabilities of the measurement packet and the response packet. When traffic is oriented towards episodic, the probability of requiring an L2CAP retransmission is roughly 0. As the operation leans towards streaming with $P_{epi} = 0.1$ and voice-type interference, the probability of a retransmission starts to increase at 60 interferers when the flush probabilities increase. This is much less than for application-type interference where the increase is at around 100 piconets.

Probability of packet loss: SPP episodic packets are lost when they are flushed, but the HDP episodic packets are lost when the L2CAP retransmission attempts fail. Fig. 3b shows that HDP episodic packets suffer less loss than SPP ones. When compared to the proprietary protocol, the L2CAP retransmissions allow HDP to provide higher reliability even with a larger packet size. Increasing the retransmission attempts would make the exponential increase to occur at a higher number of interferers, but it would not eliminate it. IEEE 11073-20601 streaming packets have a similar loss rate under SPP and HDP. The proprietary protocol outperforms them both in this case because of the smaller packet size.



(a) $Pr\{$L2CAP retransmission$\}$ using HDP

(b) $Pr\{$packet loss$\}$

Fig. 3. Probabilities for L2CAP retransmissions and packet loss in episodic mode

V. Concluding Remarks

We have compared the recently released HDP with the well-known SPP in the context of a real-life health monitoring application. In particular, two interference types were considered. The first is due to application data, and the second due to voice data. Traffic models, created for the different data sources, were used in our analysis. When using IEEE 11073-20601 data, HDP and SPP turn out to be similar. However, HDP offers more reliability for episodic traffic due to the L2CAP retransmissions. The proprietary protocol has better performance than IEEE 11073-20601 data, because the packet size gets smaller due to more optimized data representation. When the interference is of voice-type, not much can be done to improve performance. The interference levels can only be lowered by reducing the number of interfering piconets. This is because voice traffic has a constant load on the channel. However, when the interference is of application-type, the rate controller can have significant impact on the performance. With a high number of interfering piconets, the rate controller can be used to reduce the interference level, thereby improving the performance. Operating more towards episodic mode reduces the number of transmissions and consequently the probability of having collisions. This interference reduction comes at the price of providing the service application with less data than it requires.

If we relax the assumption that SPP uses the Basic L2CAP mode, and allow it to use the Enhanced Retransmission and the Streaming modes, it would offer the same reliability as HDP. The difference between these profiles would be in the way channels are established. The main advantage of HDP is the fact that all of its different aspects — from connection establishment to data representation and exchange — are standardized, thereby resulting in better interoperability. More work can be done to compare HDP with SPP. These profiles require the MCAP and RFCOMM protocols, which run on the host. The operation of these protocols can be analyzed. This would give further information concerning memory requirements, processing workload, power consumption of the host, and delays.

References

[1] Bluetooth SIG, “Bluetooth Core Specification v2.1 + EDR,” July 2007.

[2] Bluetooth SIG, “Core Specification Addendum 1,” June 2008.

[3] Bluetooth SIG, “Health Device Profile V1.0,” June 2008.

[4] Bluetooth SIG, “Serial Port Profile V1.1,” February 2001.

[5] Bluetooth SIG, “RFCOMM,” June 2003.

[6] “Health informatics-personal health device communication part 20601: Application profile- optimized exchange protocol,” *IEEE Std 11073-20601-2008*.

[7] “Health Informatics-Personal health device communication Part 10441: Device Specialization–Cardiovascular Fitness and Activity Monitor,” *IEEE Std 11073-10441-2008*, January 2009.

[8] R. Diemer, “Documentation: Datenuebertragungsprotokoll für InPriMo_activity über Bluetooth.” http://www.rcs.ei.tum.de/pub/InPriMo/Datenuebertragungsprotokoll_InPriMo_Activity.pdf, May 2008.

[9] A. El-Hoiydi, “Interference between Bluetooth networks-upper bound on the packet error rate,” *IEEE Communications Letters*, vol. 5, June 2001.

[10] A. Karnik and A. Kumar, “Performance analysis of the Bluetooth physical layer,” in *IEEE International Conference on Personal Wireless Communications*, 2000.

[11] S. Baatz, M. Frank, P. Martini, and C. Scholz, “A worst-case model for co-channel interference in the Bluetooth wireless system,” in *IEEE 28th International Conference on Local Computer Networks*, 2003.

[12] T.-Y. Lin, Y.-K. Liu, and Y.-C. Tseng, “An improved packet collision analysis for multi-Bluetooth piconets considering frequency-hopping guard time effect,” in *IEEE 58th Vehicular Technology Conference*, 2003.