Time Efficient Connection Formation Scheme in Bluetooth

Ishtiyaq Ahmad Wani and Roohie Naaz Mir

***Abstract*—Communications devices in a highly mobile environment need to minimize connection setup times in order to maximize useful data transfer. In this paper we analyze the device discovery process of Bluetooth, a technology that has potential in short-range, high mobility applications. In order to improve Bluetooth's performance in a mobile environment, it is desirable to reduce the amount of time it takes to set up connections between two devices. We suggest a simple change to the Bluetooth specification: using train A and train B (carrying of inquiry or paging messages) alternatively during inquiry and paging instead of using train A 256 times before using train B 256 times. This simple change can decrease connection setup time by a factor of approximately 3 without having deleterious effects on the overall system performance.**

***Index Terms*—Connection formation, discovery process, inquiry, paging.**

I. Introduction

One of the emerging short-range wireless networking technologies is the recent industry standard Bluetooth [1]. Bluetooth evolved from the need to replace wires in short-range communication, e.g., cables between mobile handsets and their headsets, or serial cables between computers and peripherals, with short-range wireless links. Derivable from the maximum transmission power and receiver sensitivity of the specification [1], a class-3 (most pervasive) BT has a transmission range of approximately 10 meters in a free propagation environment at a nominal ISM band frequency of 2.4 GHz. Bluetooth employs frequency hopping over 79 carrier frequencies, spaced 1MHz away, with a spectral efficiency of 1bps/Hz. The main communication structure of Bluetooth – called piconet – is a point-to-multipoint star topology, with a master node in the center and slave nodes at the perimeter of the star. Piconets can be interconnected to scatternets by nodes taking on master or slave roles in more than one piconet concurrently. The above-described characteristics make Bluetooth a viable candidate for establishing inexpensive personal area networks.

Before BT devices can exchange information among each other, they have to go through a three-phase link establishment procedure. In the first phase, the devices have to scan and search through a subset of the hopping frequencies to determine whether there are other devices in their transmission range. This first phase serves as the neighbor discovery process, and is referred to as inquiry in Bluetooth terminology. In the subsequent step, devices that are already aware of each other's proximity, initiate a handshake process by which they exchange crucial information for piconet formation; this phase is also referred to as paging. The third phase deals with the setting up of a virtual channel for further control information exchange and negotiating communication parameters relating to link management issues. It is implied, by the above three-phase process, that only because devices may be in each other's transmission range they do not necessarily have the means to communicate with each other (unlike Wi-Fi). This paper concentrates on the first two phases while proposing and describing a technique to speedup device discovery.

The rest of the paper is organized as follows: In Section 2 we analyze the mandatory inquiry and paging process of Bluetooth in more detail .In Section 3 we introduce our approach for speeding up the inquiry and paging processes. In Section 4 we compare the results of our approach and the existing connection procedure. Concluding remarks are drawn in Section 5 and references are listed in the end.

II. Analysis of Existing Connection Procedure

Connection formation in Bluetooth comprises of two procedures: 1. Inquiry 2. Paging

A. *Inquiry Procedure*

A unit that wishes to discover other Bluetooth units in range enters inquiry substate. It continuously transmits inquiry messages (inquiry access code packets - IAC) at different hop frequencies. Between inquiry transmissions the unit listens for responses. It transmits two inquiry messages in each TX slot and listens for a response in the preceding RX slot as shown in Fig. 1. If the response is received, it is not acknowledged and the probing unit continues with the inquiry transmissions. The unit leaves inquiry state either when it received a predetermined number of responses or when the Inquiry TX timer runs out. The inquiry message broadcast by the source does not contain any information about the source. However, it may indicate which class of devices should respond. There is one general inquiry access code (GIAC) to inquire for any Bluetooth device, and a number of dedicated inquiry access codes (DIAC) that only inquire for a certain type of devices. During an inquiry substate, the discovering unit collects the Bluetooth device addresses and clocks of all units that respond to the inquiry message [2]. It can then, if desired, make a connection to any one of them by means of the page procedure. A hopping sequence consists of two groups of frequencies: train A and train B (each of which are 16 frequencies long). According to

Manuscript received April 17, 2012; revised June 3, 2012.

Ishtiyaq Ahmad Wani and Roohie Naaz Mir are with the NIT Srinagar, Hazratbal, 190006, JandK, India (e-mail: naaz310@yahoo.co.in).

DOI: 10.7763/IJFCC.2012.V1.91

the Bluetooth standard [1] a single train must be repeated 256 times before another train is used. At least three train switches are needed (4 train sequences). Since each train is 10ms long, the inquiry procedure can take up to 10.24s

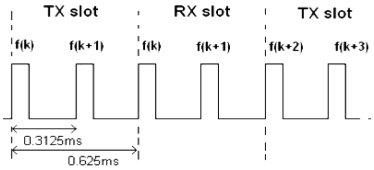

Fig. 1. RX/TX cycle of Bluetooth transceiver in Inquiry mode.

Inquiry Scan (IS)

A unit that wishes to be discovered by other Bluetooth devices in range enters inquiry scan substate. It continuously listens for inquiry messages at different hop frequencies. The period of inquiry scan can be 0s (continuous scan) – R0, 1.28s – R1 mode, or 2.56s – R2 mode. A scanning unit listens for an IAC for Tw_inquiry_scan seconds. Tw_inquiry_scan should be long enough to scan 16 frequencies (1 train). Because one train lasts for 10ms, Tw_inquiry_scan should also be 10ms. During these 10ms the receiver of the scanning device listens on a single frequency determined by the inquiry scan hopping sequence and the current value of the device's clock. The scanning device changes its listening frequency according to inquiry hopping sequence every 1.28s. But, because several units might respond to an inquiry at the same time, a protocol for slave inquiry response in order to avoid or minimize the probability of collisions is needed. Thus, when the inquiry scanning unit receives an inquiry message it generates a random number, RAND, uniformly selected between 0 and 1023 slots, returns to Connection or Standby state for duration of RAND slots. After this random backoff time, it returns to the inquiry response substate and on the first inquiry message received it will answer with an FHS packet. The FHS packet contains its Bluetooth device address (BD_ADDR) and clock value among other information. After sending the FHS packet, the inquiry scanning unit enters Page scan (PS) substate and waits to be paged for a certain time [3].

B. Paging Procedures

Page Scan (PS)

Page scan works similar to inquiry scan, however, in page scan a unit listens for its own unique DAC (Device access code) and it alone can respond. There are 32 paging frequencies, which comprise a page hopping sequence, determined by the paged unit's BD_ADDR. Every 1.28s, a different listening frequency is selected as in inquiry scan. During a scan window the unit listens on one frequency.

Page (P)

When a Bluetooth unit wants to make a connection to another unit, it pages that unit. Paging means sending an ID packet with a certain DAC in it over and over again until a response is received. The master does not know exactly when the slave wakes up and on which hop frequency, therefore it transmits a train of identical DACs at different hop frequencies and listens in between for responses. The master uses the slave's BD_ADDR and an estimate of the slave's clock to determine the page hopping sequence. To compensate for the uncertainty in the knowledge of a slave's clock, the master will send its page message during a short time interval on a number of wake-up frequencies. During each transmission slot the master sequentially transmits on 2 different hopping frequencies. The page hopping sequence of 32 frequencies is divided into two trains of 16 frequencies each and each train is repeated for Npage times or until a response is received. Paging ends until a response is received or timeout value Page to is exceeded.

Npage depends on the mode of page scan. In R0 mode we have to repeat one train before trying another one a number of times and Npage = 1. Since the other unit is listening continuously it doesn't matter when we start paging. In general, Npage is such that paging is as long as Tpage_scan. Thus, in *R*1 and *R*2 modes Npage is 128 and 256, respectively.

III. Proposed Approach

The unit that wants to discover other units in its range should send inquiry messages (IAC) on train A and Train B alternatively (instead of sending each train 256 times before using next one) as illustrated in Fig. 2

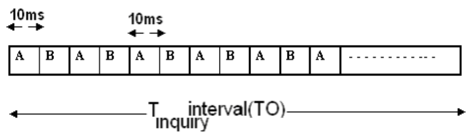

Fig. 2. Inquiry train.

The Tinquiry Interval (TO) should be long enough to get responses from all the devices in the range of inquirer (master). As will be seen TO = 3.18s is sufficient to gather responses from all the devices in the range of inquirer (Master).

The other devices (slaves) enters inquiry scan once in every 2.56s (R2 mode) or 1.28s (R1 mode) for a duration of 20ms (instead of 10ms in existing scheme) to scan both trains AandB as illustrated in Fig. 3.

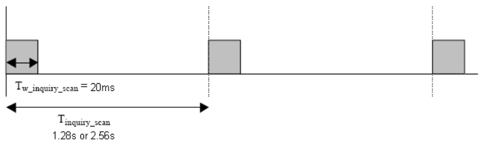

Fig. 3. Inquiry scan.

Using this approach the maximum time which the inquirer needs to discover other Bluetooth units will be only 3.22s in R2 mode and 1.94s in R1 mode (instead of 10.24s in the existing scheme) as illustrated in Fig. 4.

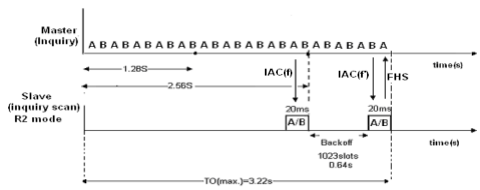

Fig. 4. Inquiry procedure.

Irrespective of at what time the slave wakes up in the inquiry scan window, it will receive the inquiry message in the first scanning window and then waits for a random amount of time between 0-1023 slots (in order to avoid collisions). After random backoff is finished the slave again scans the channel for inquiry messages and gives response (FHS) to master after receiving the inquiry message. Since the slave can wakeup at any time to scan for inquiry messages with equal probability, the average time it takes to respond to the inquiry message is (2.56+0.64+0.02)/2)s = 1.61s in R2 mode and 0.97 in R1 mode.

The paging will also be performed in a similar manner and the average time which it takes to page a slave will be 1.28s (2.56s maximum) in R2 mode and 0.64 (1.28s maximum) in R1 mode of page scanning as seen in Fig. 5.

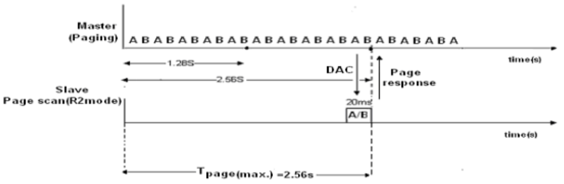

Fig. 5. Paging procedure.

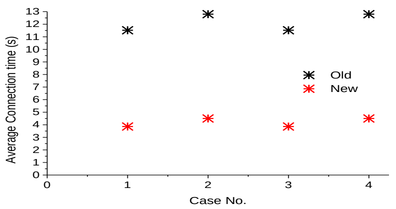

Fig. 6. Case No. Vs average connection time (Inquiry + paging).

A. Analytical Results

When we try to evaluate the performance of the connection formation procedure (old or new) we are interested in how much time it takes for the connection to be established from the time an application requested it. We can start by assuming that every time the master wants to make connection with some other device (slave) it will go for both inquiry and paging and we also assume that inquiry is not finished until inquiry timer TO expires. Secondly we must decide which mode of inquiry scan and page scan is being used –*R*1 or *R*2. A particular unit will use one particular mode. Therefore we will consider each case separately.

Let us consider some common combinations of scenarios in table I.

TABLE I: COMPARISON OF VARIOUS POSSIBLE SCENARIOS

Case No.	Case Description	Results using existing scheme	Results using proposed scheme
Case 1	Connection Establishment with inquiry in R1 mode and paging in R1 mode	11.52s	3.86s
Case 2	Connection Establishment with inquiry in R1 mode and paging in R2 mode	12.80s	4.5s
Case 3	Connection Establishment with inquiry in R2 mode and paging in R1 mode	11.52s	3.86s
Case 4	Connection Establishment with inquiry in R2 mode and paging in R2 mode	12.80s	4.5s

B. Existing Scheme:

In case 1, the average time it takes a unit to respond to an inquiry is 1.28s. However, paging doesn't begin right after a slave responds to an inquiry message. The master will continue inquiry procedure until Inquiry TO expires. According to Bluetooth Specification, Inquiry TO can be anywhere from 1.28s to 61.44s. However, 10.24s is sufficient to gather all possible responses from 7 slaves. Thus it is reasonable to assume that inquiry will last on average 10.24s. The average duration of paging is 1.28s. Thus connection formation will take on average 11.52s (table I) in an error free environment. In case 2, inquiry still takes about 10.24s. Paging will now take on an average of 2.56 s. So connection formation will take 12.80s. Case 3 and 4 are similar to case 1 and 2 respectively because inquiry time is independent of inquiry scan mode.

C. Proposed Scheme:

In the proposed technique the average time it takes a unit to respond to an inquiry is 0.97s in case 1. However, paging doesn't begin right after a slave responds to an inquiry message. The master will continue inquiry procedure until inquiry to expires. In the new approach TO = 3.22s is sufficient to gather all possible responses from 7 slaves. Thus it is seen that inquiry will last an average of 3.22s. The average duration of paging is 0.64s. Thus connection formation will take an average 3.86s in an error free environment. In case 2, inquiry still takes about 3.22s. Paging will now take an average 1.28s. So connection establishment will take 4.5s. Case 3 and 4 are similar to case 1 and 2 respectively because inquiry time is independent of inquiry scan mode.

IV. COMPARING THE RESULTS

The exact amount of improvement of new procedure over the existing one depends on the relative frequency of each case. However it is difficult to say which case is common and which isn't since we cannot predict the behavior of users. Assuming all cases are equally frequent the average connection establishment time for the proposed procedure is 12.61 s. The average connection establishment time for the new procedure is 4.18s. There is an improvement in the new approach over the existing one by a factor of 3.

V. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a new procedure to connect Bluetooth devices .We showed analytically that this proposal is more efficient than the existing one .The proposed procedure for connection formation uses trains *A* and *B* alternatively .The results show that the new proposal takes less time than the existing one. The results presented in this paper were derived analytically. Future work includes a computer simulation of the new procedure to uncover the advantages or disadvantages of the scheme. The exact power expenditure is also of interest, since in order to support new connection scheme the inquiry scan and page scan times doubles but at the same time inquiry and paging times reduces.

REFERENCES

[1] *Bluetooth Special Interest Group*, "Specification of the Bluetooth System, Core Version 1.1," Specification vol. 1, [Online]. Available: http://www.bluetooth.com, February 2001.

[2] J. Haartsen, "Bluetooth-The universal radio interface for ad hoc, wireless connectivity," Ericsson Review no. 3, 1998.

[3] J. C. Haartsen, "The Bluetooth Radio System," *IEEE Personnel Communications*, February, 2000.

[4] A. Busboom, I. Herwono, M. Schuba, and G. Zavagli. "Unambiguous Device Identification and Fast Connection Setup in Bluetooth," *Proc. of the European Wireless 2002*, vol. 0, Florence, Italy, Feb, 2002.

[5] K. Cheolgi, M. Joongsoo, and L. Joonwon, "A Random Inquiry Procedure using Bluetooth," *Proc. of International Conference on Communication in Computing (CIC)*, Las Vegas, USA, Jun, 2001.

[6] P. Murphy, E. Welsh, and J. P. Frantz, "Using Bluetooth for Short-Term Ad-Hoc Connections Between Moving Vehicles: A Fesability Study," *IEEE Vehicular Technology Conference*, vol. 1, no. 55, Birmingham, AL, May, 2002, pp. 414-418.