

## Research Article

# Sensor Protocol for Roaming Bluetooth Multiagent Systems

Neungsoo Park,<sup>1</sup> Bijoy Kumar Mandal,<sup>2</sup> and Young-Ho Park<sup>3</sup>

<sup>1</sup> Department of Computer Science and Engineering, Konkuk University, Seoul 143-701, Republic of Korea

<sup>2</sup> Computer Science & Engineering Department, NFET, NSHM Knowledge Campus—Durgapur, Durgapur 713212, India

<sup>3</sup> Department of Multimedia Science, Sookmyung Women's University, Seoul 140-742, Republic of Korea

Correspondence should be addressed to Young-Ho Park; [yhpark@sm.ac.kr](mailto:yhpark@sm.ac.kr)

Received 29 January 2013; Accepted 4 April 2013

Academic Editor: Sabah Mohammed

Copyright © 2013 Neungsoo Park et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Bluetooth is a low-cost, short-range wireless technology capable of providing many communication functionalities. However, Bluetooth does not support any sensor protocol which is related to a roaming and in which handoff occurs dynamically when a Bluetooth device is moving away from coverage of the network. If a device is losing its connection to the master device, there is no provision which transfers it to another master. Handoff is not possible in a piconet, as in order to stay within the network, a slave would have to keep the same master. Thus, by definition intrahandoff is not possible within a piconet. This research mainly focuses on Bluetooth roaming sensor technology and designs a sensor protocol which works in a roaming for Bluetooth multiagent system technology. The advantage of designing a roaming protocol is to ensure the Bluetooth enabled roaming devices can freely move inside the network coverage without losing its connection or break of service, in case of changing the base stations.

## 1. Introduction

Bluetooth is a low-cost, short-range wireless technology capable of providing many communication functionalities, ranging from wire replacement to simple personal area networking [1]. The range of Bluetooth network can be 0 to 100 meters. It is used to transmit both synchronous and asynchronous data. The bandwidth of the Bluetooth network, at physical layer, is 2.1 Mbit/s. Compared with other systems in the same frequency band, the Bluetooth radio hops faster and uses shorter packets. With respect to other wireless communication devices, Bluetooth connection can support both data and voice communications.

Bluetooth has two types of networks, piconets and scatternets. Piconet (PAN) consists of a master and one to seven active slaves. In a piconet, the devices share the same frequency-hopping spread spectrum (FHSS) channel, which is a transmission technology used in a local area wireless network. A scatternet is a collection of piconets and a connection node which links two piconets. This node can be simultaneously a slave of multiple piconets or a master in only one piconet. On the other hand, a Bluetooth device can

be master in one and slave in other piconets, or slave in all piconets it is connected to [2].

An agent refers to an entity that functions continuously and autonomously in an environment in which other processes take place and other agents exist [3]. It can be a physical or software entity. Multiagent systems (MAS) involve a team of intelligent agents working together to accomplish a given task. The accomplishment of a task depends on the coordination of actions and behaviour of the agents. To achieve coordination between agents, communication can be used. Usually, multiagents operate in a close proximity. Thus, Bluetooth technology can be employed to set up the communication among agents.

This research mainly focuses on Bluetooth roaming sensor technology and designs a sensor protocol which works in roaming for Bluetooth multisystems technology. The advantage of designing a roaming protocol is to ensure that the Bluetooth enabled roaming devices can freely move inside the network coverage without losing its connection or break of service in case of changing the base stations.

The remainder of this paper is organized as follows. Section 2 explains Bluetooth protocol stack. Section 3

describes our methodology. Section 4 discusses how to connect Bluetooth devices. Section 6 illustrates data transfer operation. Section 7 explains how to connect slave which goes out of the piconet. Section 8 demonstrates handover old access point to new access points. Section 9 highlights conclusions and future work.

## 2. Bluetooth Protocol Stack

The Bluetooth protocol allows the authentication and encryption of a link at the same time. The security is controlled by the lower layers of the Bluetooth protocol. First, security level is assured by the fact that each Bluetooth module has a unique MAC (48-bits) address. In this way, when a scanner connects itself to a terminal with which it is twinned, the bar code is transmitted to this equipment. This process is shown in Figure 1.

The baseband layer is responsible for controlling and sending data packets over the radio link. It provides transmission channels for both data and voice. The baseband layer maintains Synchronous Connection-Oriented (SCO) links for voice and Asynchronous Connectionless (ACL) links for data. The SCO link is a symmetric point-to-point link between master and single slave in the piconet. SCO packets are not retransmitted, but ACL packets are, to ensure data integrity [5]. The Link Manager Protocol (LMP) uses the links set up by the baseband to establish connections and manage piconets. Responsibilities of the LMP also include authentication and security services and monitoring of service quality [5].

The Host Controller Interface (HCI) is the dividing line between software and hardware. The Logical Link Control and Adaptation Protocol (L2CAP) and layers above are currently implemented in software, and the LMP and lower layers are in hardware. The HCI is the driver interface for the physical bus that connects these two components. The L2CAP can be accessed directly by the application, or through certain support protocols provided to ease the burden on application programmers. Quality of Service (QoS) parameters are exchanged at this layer as well.

## 3. Methodology

The design process of a sensor protocol for roaming Bluetooth, equipped with MAS, will be carried out in two steps. The first step is to define roaming, when it should occur, for example, when an object will realize that it is going out of the range. And the second step is how to design a roaming sensor protocol. The main focus of this paper is to design a sensor protocol for roaming Bluetooth equipped with MAS.

*3.1. Connection and Data Transfer on Roaming Devices.* Here we assume that master is fixed unit connected to a fixed network, while slaves are mobile units. If a mobile unit loses the connection to the fixed unit it currently communicates with, a new unit is chosen to relay all the packets to the communication partner. For connection and data transfer between master and roaming slave, we need a sensor slave

between them. For example, slave is a common unit between both piconets in Figure 2.

*3.2. Maintain a Connection If Slave Is out of the Reach of Master.* To know when slave will lose connection, RSSI (Received Signal Strength Indication) will be checked if it is in acceptable level. This means the roaming object is within the range of the network coverage. If the RSSI value of the moving object is less than the acceptable level, then it means the object is going out of range of the network coverage. That is the point where the object will start roaming and find the value of RSSI. There is a command used in Host Controller Interface (HCI) layer of the Bluetooth protocol stack `get_link_quality` [2]. This command enables us to measure the quality of RSSI on the basis of which the roaming occurs.

## 4. Mathematic Modelling

Consider an object which wants to move in  $x$ - and  $y$ -coordinating system using Bluetooth technology for communication. The position of the object with reference to  $x$ ,  $y$  is  $A(x, y)$  as shown in Figure 3.

Let  $r$  be the distance object from its signal source. Thus, after resolving  $r$  into the rectangular components, the horizontal and vertical component becomes

$$x = r \cos \theta, \quad (1)$$

$$y = r \sin \theta. \quad (2)$$

Now, if we square and add (1) and (2), we get the following formula:

$$\therefore r = \sqrt{(x^2 + y^2)}. \quad (3)$$

The position of the object with respect to  $x$ - and  $y$ -coordinating system is as follows:

$$\tan \theta = \frac{y}{x} \longrightarrow \theta = \tan^{-1} \frac{y}{x}. \quad (4)$$

Velocity can be defined as

$$\text{velocity} = \frac{\text{displacement}}{\text{time}}. \quad (5)$$

Thus, (2), (3), and (4) for  $X$ -Axis become (with respect to time)

$$v_x = \dot{x} = \frac{d(r \cos \theta)}{dt}, \quad (6)$$

$$\text{or, } v_x = \dot{x} = -r \sin \theta.$$

Similarly, for  $Y$ -Axis,

$$v_y = \dot{y} = r \cos \theta. \quad (7)$$

Differentiating (4) with respect to time,

$$\theta = \frac{\dot{x}y - y\dot{x}}{x^2 + y^2}, \quad (8)$$

$$\text{angular velocity} = \frac{\text{angular position}}{\text{time}}.$$

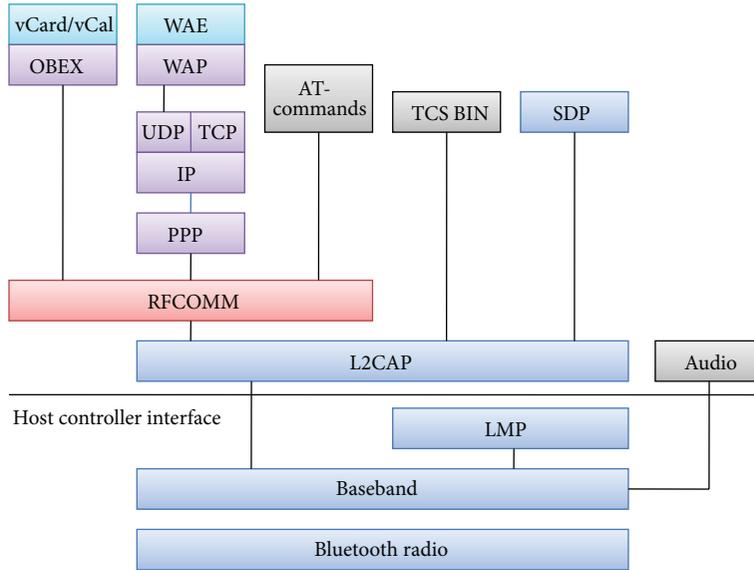


FIGURE 1: Bluetooth protocol stack model [4].

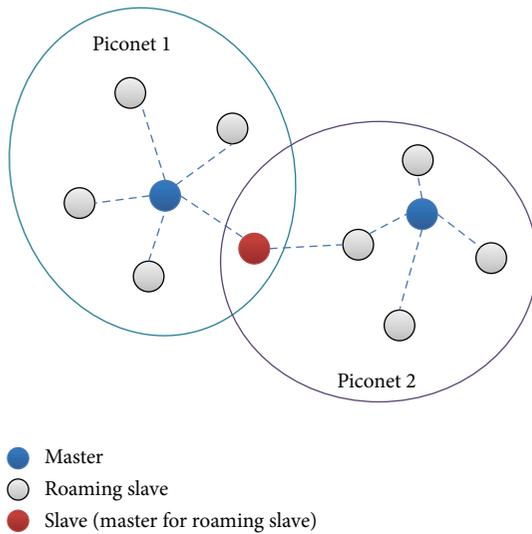


FIGURE 2: Data transfer from master to roaming slave.

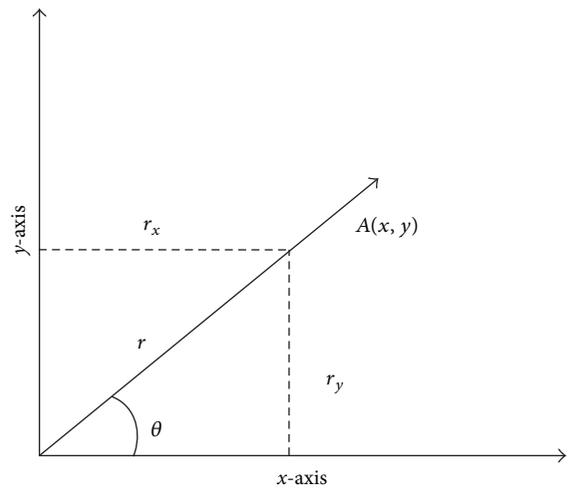


FIGURE 3: Graphical representation of object in x-y- coordinating system.

Equations (3) and (7) give us trajectory of the object in a piconet:

$$\begin{aligned} \therefore r &= \sqrt{(x^2 + y^2)}, \\ \omega &= \theta \dot{\theta} = \frac{x\dot{y} - y\dot{x}}{x^2 + y^2}. \end{aligned} \tag{9}$$

### 5. Connecting Bluetooth Devices

A connection between two devices occurs in the following way: if any of the devices has no information about another/remote device, the inquiry, listing, and page procedure have to be carried out.

**5.1. Inquiry Procedure.** The inquiry procedure enables a device to discover devices in range and determine the addresses and clocks for those devices. This procedure occurs in the following steps.

- (1) Inquiry procedure involves a source device sending out inquiry packets and then receiving the inquiry reply. When a source device wishes to discover new devices, it enters the inquiry state, where it broadcasts inquiry packets (ID packets), containing the IAC, to all devices in range. It will send these using the inquiry hopping sequence. The device in the inquiry state can also receive inquiry replies (FHS packets); however, it will not acknowledge these packets.

- (2) The destination device that receives the inquiry packets will have to be in the inquiry scan state to receive the inquiry packets.
- (3) The destination device will then enter the inquiry response state and send an inquiry reply (own address and clock) to source device.
- (4) After receiving inquiry reply, source device will create a record.
- (5) Master will show a list of devices to a user. Table 1 shows an example of this list.  
Suppose device 1 is a roaming 2 of slave 1 and device 2 is a roaming 1 of slave 2. Then master will focus on those devices which are needed for the connection, having minimum path from master to that device and all the devices.
- (6) Master will send a list of devices, as in Step (5), to all the devices which are in the list after or during page procedure.

5.2. *Listing Procedure.* The listing procedure is carried out in followings steps as

- (1) master will put its address and a few bits as a record in its database;
- (2) during the inquiry procedure, slave will enter the inquiry response state and send an inquiry reply (its own address) to its master. After receiving address of each slave, master will add a new record to its database;
- (3) now, slave 1 will create a record like in Step (1). During inquiry procedure of slave, roaming slave will enter the inquiry response state and send an inquiry reply (its own address) to slave. After receiving address of each roaming slave, slave will add a new record to its database as described in Step (2).
- (4) all slaves will send their own record to the master as shown in Figure 4;
- (5) master adds records to its own database.

5.3. *Paging Procedure.* With the paging procedure, actual connection can be established. The page procedure follows the inquiry procedure and listing procedure. Only the Bluetooth device address is required to set up a connection. Knowledge about the clock (clock estimate) will accelerate the set-up procedure. A unit that establishes a connection will carry out a page procedure and will automatically be a master of the connection. The procedure occurs as follows.

- (1) A source device pages the destination device. Source device searches for another destination devices. The source device sends out a page packet (ID packet), using the page hopping sequence, to notify other devices that it wants to obtain destination device and/or their services.
- (2) The destination device receives the page. A destination device listens for page trains containing its

TABLE 1: User devices.

From User's list (selected these)	Background (in mobile)
Device 1	Device 1 addr + extra bits
Device 1	Device 2 addr + extra bits

own device access code (DAC). When a destination device wishes to receive page packets, it enters the page scan mode. The scanning will follow the page hopping sequence. If a destination device receives a page packet, it will enter in slave response state.

- (3) The destination device sends a reply to the source. Once a destination device has received its own DAC from the source (in the ID packet), it will enter this state. It will send a response message (its DAC again) to the source using the page response hopping sequence.
- (4) The source device sends FHS packet to the destination device. When the source device has received a reply to its original page message, it will enter this state. It will then send FHS packet to the destination device using the page hopping sequence.
- (5) The destination device sends its second reply to the source device. Once the destination device has received the FHS packet from the source, (Page Master Response State: Steps (3) and (4)), the destination device will send a reply to the source (an ID packet containing the destination DAC).
- (6) The destination device and source device then switch to the source channel parameters. When the source device has received the second reply (Page Slave Response State: Steps (3) and (5)), it knows that the destination device has received the FHS packet (Page Master Response State: Steps (3) and (4)). The source device is now the master of the destination device (the slave) and the destination device will switch to the source device's channel parameters. The destination device is now the slave of the source device (the master).

The connection state starts with a poll packet sent by the master to verify that slave has switched to the master's timing and channel frequency hopping. The slave can respond with any type of packet. The master will send the page packets to slave. Using extra bits, slave detects the roaming slave and sends the page packets which it has got from master. When roaming slave wants to send any information to master, it will send this to slave. Then slave will forward the package to master.

## 6. Data Transfer: Packets

Each packet consists of 3 entities, the access code (68/72 bits), the header (54 bits), and the payload (0–2745 bits) as shown in Figure 5.

- (1) Access code: access code is used for time synchronization, offset compensation, paging, and inquiry.

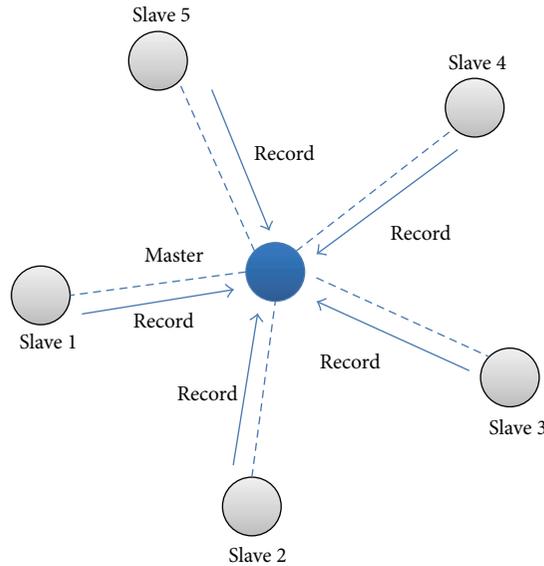


FIGURE 4: Slave sending its own record to master.



FIGURE 5: Packet Format.

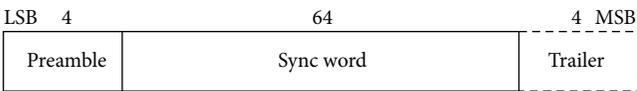


FIGURE 6: Access code format.

There are three different types of access code: Channel Access Code (CAC), Device Access Code (DAC), and Inquiry Access Code (IAC). The channel access code identifies a unique piconet, while the DAC is used for paging and its responses. IAC is used for inquiry purpose [6] as shown in Figure 6.

- (2) Header: the header contains information for packet acknowledgement, packet numbering for out-of-order packet reordering, flow control, slave address, and error check for header [4] as shown in Figure 7.
- (3) Payload: the packet payload can contain either voice field or data field or both. It has a data field; the payload will also contain a payload header [3].

### 7. Connecting Slave Which Goes out of the Piconet

To maintain security and connectivity in Bluetooth, another procedure will be added after paging procedure. Following paging procedure, connection will be created. Now the first package that will be sending before any data transfer will contain the list of devices going to get the same file.

Reconnection is required only if a roaming slave 1, as shown in Figure 8, went out of piconet, and if it acts as a roaming slave in some other piconet.

If roaming slave 1 went out of the reach of its master (slave 1), then it will go for inquiry procedure. After inquiry procedure, it will get a list of devices with which it can create a connection. Now it will compare this list with the list it has gotten after this procedure and then find the common devices, that is, sensor in the list. If roaming slave 1 will get any device, then it will send a request to that device and go for paging procedure. Then, roaming slave 1 will update database and send a message to other devices to update their own records for the changes that have taken place in the scatternet. Otherwise it will show a warning message and disconnect itself from the scatternet. There might be multiple Bluetooth devices offering the same service, and most often the best one is the one from which we receive the strongest signal (because we will get a lower Bit Error Ratio and we will decrease the frequency of handovers). So, it will now find the device close to it. The flow chart of connection of slave is shown in Figure 9.

### 8. Handover Old Access Point to New Access Point

A Bluetooth device is connected with another Bluetooth device and suddenly moves out of range. After some time, the Bluetooth stack in the device declares the link to be dead and closes the LMP connection. At this point, the L2CAP used by the network traffic receives an event (LP\_DisconnectInd) and closes the connection down. The higher layer is kept in suspended mode until the roaming procedure is completed. This procedure is explained in [7].

At this point, the Bluetooth device performs an inquiry. If the inquiry does not find any access point, the node continues with inquiry, up to the point where it times out and closes

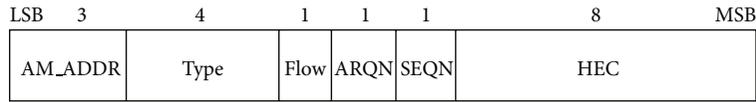


FIGURE 7: Header Format.

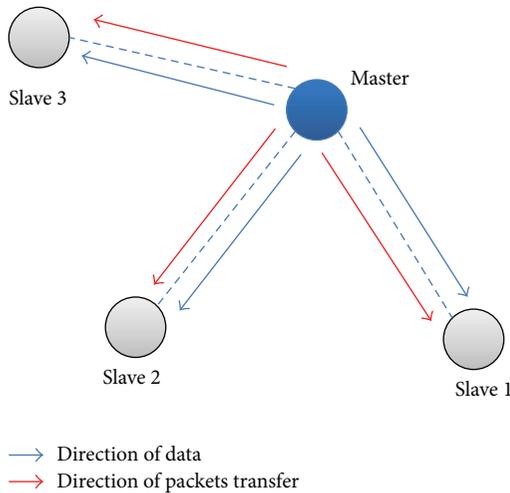


FIGURE 8: Data transfer in devices.

down the higher layer (either PPP or BNEP). If the Bluetooth device finds another Bluetooth devices, then if the “Service Name” of this Access Point is the same as the one of the previous access point, the device connects to it. If the “Service Name” is different and if the device has discovered other access points in the inquiry process, the device should try to connect to those access points. If the device cannot find an access point with the same “Service Name,” it can either try to connect to one of the other access points, continue doing inquiry, or return a failure message to the user.

While the node is performing its handover, the old access point will still continue to receive and buffer packets for the roaming node, but cannot deliver them. When the node is associated with the new access point, the new access point starts to receive packets for that node at this point, but does not grab earlier packets. In other words, all the packets sent from the infrastructure while the node was performing the handover are not received by the node. Those packets will need to be resent from the source, which will consume time. On the other hand, the old access point has stored most of those packets, and they will just wait in its buffer. When the old access point receives the deregistration message from the new access point, it can simply resend all the packets in its buffer associated with this node. The new access point will pick them up and deliver them to the node. The access point will collect the BR address from the slave and go for an inquiry procedure to find the old access point of this slave. Using extra bits, it will be easier to find the location of old access point.

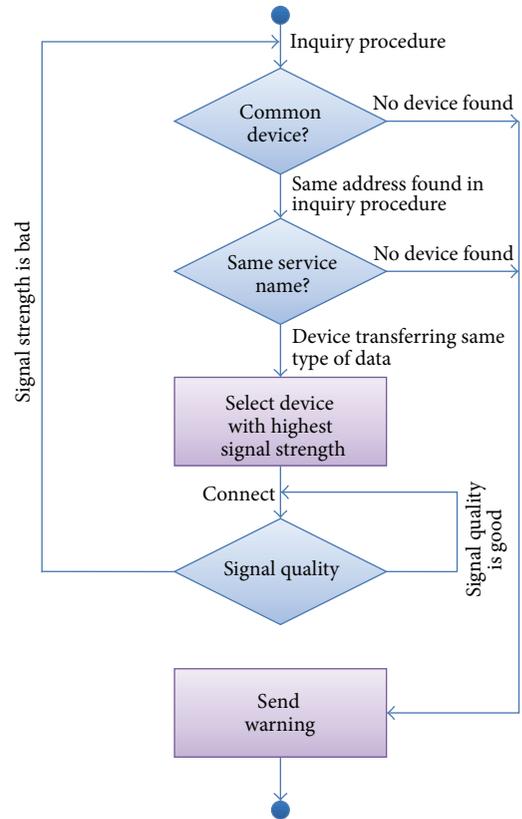


FIGURE 9: The flow chart of connection of slave.

### 9. Data Sharing in Slaves

Data sharing is one of the most important methods to increase the speed during data transfer and reduce the load of master. In this case, two or more slaves will share the data between them, so that source will not have to send the data individually to all the slaves. The flow chart of data sharing method in slaves is shown in Figure 10.

### 10. Conclusion

It is a widely held belief that a lack of the ability to create a “proper” Bluetooth pan-piconet network is one major drawback of the present incarnation of the Bluetooth specification. However, it must be noted that Bluetooth was originally designed as a low-cost/low-power system. The designing of a roaming protocol is to ensure the Bluetooth enabled roaming devices can freely move inside the network coverage without losing its connection or break of service in case of changing the base stations. The discovery process to establish connection in Bluetooth network is time-consuming process.

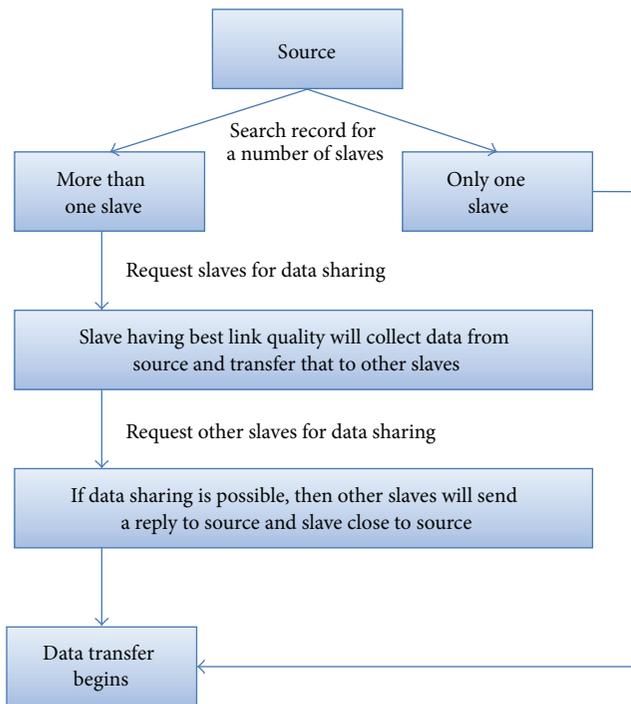


FIGURE 10: Flow chart of data sharing in slaves.

Using this roaming protocol, we also can reduce the time. Since Bluetooth is a short-range wireless communication protocol, we shall design protocol for a long-range wireless communication.

## Acknowledgment

This work was supported by the IT R&D Program of MKE/KEIT (10041854, Development of a smart home service platform with real-time danger prediction and prevention for safety residential environments).

## References

- [1] F. Mazzenga, D. Cassioli, A. Detti, I. Habib, P. Loreti, and F. Vatalaro, "Performance evaluation in bluetooth dense piconet areas," *IEEE Transactions on Wireless Communications*, vol. 3, no. 6, pp. 2362–2373, 2004.
- [2] H. Pals, Z. R. Dai, J. Grabowski, and H. Neukirchen, "UML-based modeling of roaming with bluetooth devices," in *Proceedings of the 1st Hangzhou-Lubeck Conference on Software Engineering (HL-SE' 03)*, pp. 1–7, 2003.
- [3] G. Mejia and L. J. Mahecha, "A multi-agent system approach for the supply chain of fresh products in Bogota, Columbia," *Joint Pricing and Inventory Decisions for Perishable Products*.
- [4] N. Minar and M. Tarique, "Bluetooth security threats and solutions: a survey," *International Journal of Distributed and Parallel Systems*, vol. 3, no. 1, pp. 127–148, 2012.
- [5] Y. T. Chung, *Bluetooth announcement system [Dissertation]*, Universiti Teknologi Malaysia.
- [6] Bluetooth SIG, "Specification of the Bluetooth System, Core, Part B," Version 1.1, 2001, <http://www.bluetooth.com>.

- [7] J. Tourrilhes, "BlueTooth roaming proposals," Working document submitted to the BlueTooth PAN working group, 2000, [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Papers/](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Papers/).



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

