Relay Reduction and Disjoint Routes Construction for Scatternet Over Bluetooth Radio Systems

Chih-Yung Chang, Ching-Feng Lin, Tzu-Ting Wu, Chung-Hsien Hsu
Department of Computer and Information Science, Aletheia University
32 Chen-Li St., Tamsui, Taipei, Taiwan
changcy@email.au.edu.tw

Abstract

Bluetooth[1] is a new technology for low cost, low power, and short-range wireless communication. Equipped with Bluetooth chip, mobile devices can mutually communicate with each other by constructing a piconet. A Bluetooth device that participates two or more piconets is called relay which provides device with multi-hop (or inter-piconet) communication services. In a Bluetooth scatternet, the number of relays and the degree of each relay are significant factors that determine the performance of entire network. One objective of this paper is to propose an effective protocol that can adjust the network architecture dynamically by relay reduction. A communication environment with characteristics of connected network, high bandwidth utilization and low maintenance cost thus can be constructed. In addition, we propose a routing protocol that is able to reduce the path length and create two disjoint routes for any pair of source and destination devices located in different piconets. Simulation results show that the proposed protocols are performance well in terms of routing length, bandwidth consumption, and transmission delay.

Keywords: Bluetooth, piconet, scatternet, relay, hopping sequence.

1. Introduction

The advances of computer technology and the population of wireless equipment have promoted the quality of our daily life. The trend of recent communication technology is how to make good use of wireless equipments for constructing an ubiquitous communication environment. Bluetooth is a low cost, low power, and short range communication technology that operates at 2.4GHz ISM bands. Two types of links, ACL and SCO links, are respectively used for transmitting data and voice over Bluetooth radio systems. To avoid the effect of co-channel interference, the radio frequency (RF) module hops over 79 channels in a speed of 1600 times per second. The short packet and the fast hopping increase the communication reliability of two Bluetooth devices. A piconet consists of at most eight Active Bluetooth devices which includes one master and multiple active slaves. The master in a piconet is responsible to manage at most seven active slaves and hundreds parked slaves[5][6].

A host can participate two or more piconets simultaneously and alternatively play role of slave in various piconets. The master in a piconet can also play the role of slave in another piconet. However, a host can not play the role of master in two or more piconets simultaneously. Mobile device that participates two or more piconets is defined as *relay*. A relay delivers messages among piconets so that the resources or services will not be restricted due to the maximum number of active members in a piconet. The packet transmission among piconets can be achieved by their common relays [3].

When a master is intended to invite a Bluetooth device as its slave, it must firstly switch to inquiry state. After the information of Bluetooth address and clock value of slaves is obtained, the master changes to paging state. Master in paging state will assign the 3-bit long *Active Member Address* (AM_Addr) to the dedicated slave. Packet sent by master contains the AM_Addr of specific slave that has the right to transmit packet to master in the next free time slot. In case that the AM_Add is exhausted, master can not further invite device as its active member. Thus, the AM_Addr is an important resource for master to organize its piconet. An active relay that alternatively participate *n* piconet will occupy *n* AM_Addr. In a scatternet, too many relays will consume the AM_Addr resource and increase the maintenance overhead. A relay reduction procedure is proposed in this paper for removing relays so that the remaining scatternet is connected and the number of occupied AM_Addr and maintenance overhead could be reduced.

The performance of a connected scatternet relies on the number of relays and the degree of each relay in this scatternet. Scatternet containing a large number of relays will profit from low probability of disconnection, short routing path and fast flooding. However, scatternet with a large number of relays suffers from the drawbacks of consumption of active member address, creating a large amount of packets in flooding and difficulty of synchronization among piconets. If the degree of a relay is large, the frequent role switching among piconets will also increase the probability of packet lost. Thus, given a connected scatternet, how to choose suitable relays and remove the others becomes a key factor that determines the performance of a scatternet. One objective of this paper is to propose a relay reduction protocol so that the unnecessary relays could be removed from scatternet.

In addition to the relay reduction problem, the route construction problem is another important issue for providing devices with communication services. Bluetooth devices that are intended to communicate with each other may belong to different piconets, even if they are within the communication range of ten meters. It is possible that two devices desiring to communicate with each other are unfortunately belonging to two disconnected scatterenets. In literature[8], a randomized algorithm is proposed to construct a connected scatternet which comprise all hosts within ten meter communication range. Given a connected scatternet, how to construct an efficient routing path is a basic and important issue for providing devices with communication service over scatternet. In this paper, a routing protocol is proposed to reduce the path length and create disjoint paths.

In this paper, we firstly examine the effects of the number of relays and the degree of each relay. Then we propose a *relay reduction protocol* that reconfigures the scatternet dynamically such that the performance of a connected scatternet will be improved. Another problem is investigated in this paper that the service provider and consumer may not belong to same piconet. How to construct short and disjoint routing paths becomes an important research issue.

In literature, a comprehensive study aims at developing on-demand routing protocol in Ad-Hoc networks. Flooding is a general technique applied for the search of destination host. Based on flooding scheme, a Routing Vector Method (RVM)[4] is proposed to construct a routing path in a given scatternet. Similar to those devices in Ad-Hoc network, Bluetooth devices can actively construct a dynamic network and communicate with other devices in a distributed and active manner. However, the restriction on number of active members existed in a piconet causes the route that is constructed by flooding scheme inefficient. Length of route created by flooding is long and could be reduced for improving the drawbacks of bandwidth consumption and transmission delay. In this paper, a path reduction protocol and a new routing protocol are respectively proposed for reducing the routing path and creating disjoint path. Simulation results show that the proposed protocols are performance well in terms of routing length and data traffic overhead.

The remaining parts of this paper are organized as follows. Sec. 2 illustrates the relation between relay and piconet by giving some examples. A relay reduction protocol that can adjust the configuration dynamically is proposed in Sec. 3, such that a communication environment suitable for QoS service can be established. Sec. 4 proposes a path reduction protocol and a RVM-based routing protocol for creating disjoint and short paths. Sec. 5 demonstrates the performance improvement of the proposed protocols via experimental results. Finally, the conclusion and future work are given in Sec. 6

2. Backgrounds and Basic Concepts

Bluetooth is a new technique for low power, low cost, and short-range communication. Each Bluetooth host in a piconet can play the role of master, slave, or relay. A master device is responsible to manage at most seven slaves in its a piconet. Bluetooth device is not allowed to play the role of master in two or more piconets simultaneously, while a host can play the role of slave in two or more piconets. Slave that participates multiple piconets is said to be a *relay*, which is able to transmit packet from one piconet to another.

In a piconet, master assigns an Active Member Address (AM_Addr) to each active slave. By applying TDD frequency access technology, the time line is divided into odd slots and even slots. Under the control of master, the even time slot is reserved for master to transmit packet to slave, whereas the odd slots is reserved for slave that receives packet from master in even slot[9][10]. There are at most eight devices in a piconet. In case that the number of Bluetooth devices is more than eight, at lease two piconets are required to cover all devices.

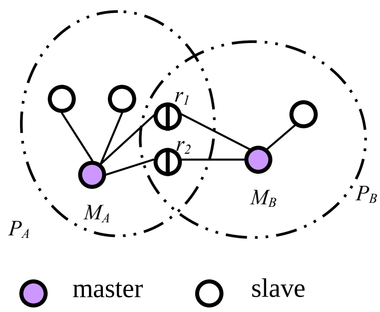


Figure 1. Scatternet structure before executing Relay Reduction Protocol.

A relay that simultaneously participates two or more piconets will be synchronized with the master in each piconet and alternatively switch between piconets. For example, relay r_1 simultaneously participates piconets P_A and P_B, as shown in Fig. 1. The packet transmission from piconet P_A to P_B is illustrated in follows. In the first time slot, r_1

synchronizes with piconet P_A and receives a data packet from master of P_A. In the second time slot, r_1 switches to and synchronizes with piconet P_B, and then forwards the received packet to master of P_B at the odd time slots.

Although relay is necessary for constructing a connected scatternet, however, too many relays will draw many disadvantages. One disadvantage is that the consumption of too many AM_Addr, causing master restricted to invite other device as its active member. The unnecessary relay will also increase the maintenance cost of entire scatternet, creating the difficulty of synchronization among those piconets the relay participates. This also causes a high probability of packet lost while relay inconsistently switches among piconets. Therefore, one focus of this paper is to investigate how to remove the unnecessary relays such that the entire scatternet is connected and the maintenance cost is low. An effective configuration of scatternet should have the following features:

1. Connected: any Bluetooth device is guaranteed to connect with any other Bluetooth device in a scatternet.
2. Small number of relay: given a connected scatternet, the advantages of small number of relay include not only low cost for maintaining relay, but also low AM_Addr consumption. This is because a relay will occupy an Active Member Address in each piconet. Thus, the relay reduction will lead to the result that other devices can easily obtain the resource of Active Member Address and successfully participate this piconet.

As illustrated in Fig. 1, there are four and three active slaves in piconets P_A and P_B, respectively. Two relays, r_1 and r_2, are existed between these two piconets. If the relay nodes $\{r_1, r_2\}$ are reduced to $\{r_1\}$ and the link connecting r_2 and M_A is broken for the consideration of balance, two piconets P_A and P_B will have equal number of active members and the released Active Member Address that is originally consumed by r_2 in P_A can be reused by some other device to participate piconet P_A.

Due to that there are at most seven active members in a piconet, two Bluetooth devices that intend to communicate to each other may belong to different piconets. A relay node is thus required to perform the packet forwarding operation for communication among piconets. In addition to the relay reduction problem, the route construction problem is another important issue for providing devices with inter-piconet communication services. In literature, a comprehensive study aims at developing on-demand routing protocol in Ad-Hoc networks. Flooding is a general technique applied for the search of destination host. Based on flooding scheme, a Routing Vector Method (RVM)[4] is proposed to construct a routing path in a given scatternet. Similar to those devices in Ad-Hoc network, Bluetooth devices can actively construct a dynamic network and communicate with other devices in a distributed and active manner. However, the restriction on number of active members existed in a piconet causes the route that is constructed by flooding scheme inefficient. Length of route created by flooding is long and could be reduced for improving the drawbacks of bandwidth consumption and transmission delay.

In this paper, a path reduction protocol and a new routing protocol are respectively proposed for reducing the routing path and creating disjoint path. Based on conventional RVM[4] routing protocol, this paper proposes a path reduction protocol and a routing protocol for reducing the path length and creating disjoint routes between any two devices. Data transmission delay and bandwidth consumption are therefore reduced. Two issues are addressed in this paper. First, a dynamic configuration approach is proposed to reduce the unnecessary relays and establish an efficient configuration for a given connected scatternet. Another issue is to investigate a routing protocol that creates disjoint routing paths with small number of hop count. Data transmission over scatternet can be more efficient.

3. Dynamic Relay Reduction Protocol for Scatternet

In this section, a distributed relay reduction protocol is proposed for removing the unnecessary relays from a given scatternet. Subsection 3.1 gives the basic definitions and basic concepts of relay reduction. The relay reduction protocol is then proposed in subsection 3.2.

3.1 Basic Definitions and Concepts

One objective of this paper is to investigate a relay reduction protocol that adjusts network configuration dynamically for a given connected scatternet. For the explanation and presentation of the remaining parts of this paper, we define symbols that will be used in this paper as follows.

Definition: **Piconet** (P_i)

A piconet consists of a master and at most seven slaves. The following symbol is used to denote a piconet:

$$P_i = \{(M_i, S_{i,j}) \mid 1 \le j \le 7\},$$

where M_i represents the master of Piconet P_i, and $S_{i,j}$ denotes one of the slaves dominated by M_i. For example, a scatternet containing three piconets P_1、P_2 and P_3 is presented in Fig. 2 where M_2 is a master of piconet P_2 and is responsible to manage slave $S_{2,1}$ and two relays.

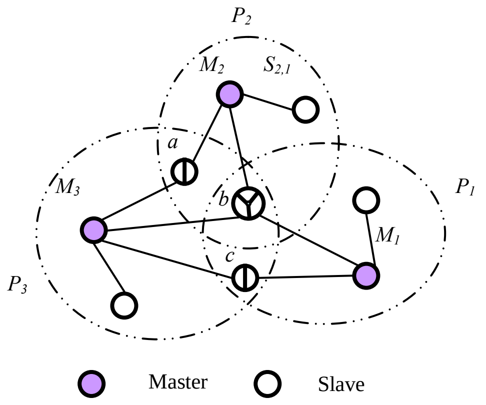


Figure 2. A simple scatternet environment, where $a=R^1_{p=\{2,3\}}$, $b=R^1_{p=\{1,2,3\}}$, $c=R^1_{p=\{1,3\}}$.

Definition: **Connected-Scatternet** (*CS*)

A scatternet consists of two or more piconets. A scatternet is said to beconnected if the following condition is satisfied:

$$CS=\left\{\cup P_i \middle| \forall (P_i \cap P_j) \neq \varnothing \right\}.$$

For the area covered by a scatternet, the above definition illustrates that if all piconets of a scatternet are connected by relays, the scatternet is called a connected-scatternet. For example, scatternet shown in Fig. 2 is a connected scatternet.

Definition: **Relay** (R_p^k)

In a connected-scatternet, sufficient number of relays is required for connecting all piconets. For the sake of explanation and presentation of the relations among relays and piconets, we define the following symbol:

Set of Relay R={ R_p^k|*p: a set of piconets,* $1 \leqq k \leqq$ *number of relay in set p*}.

As described in Fig. 2, the symbol $R^1_{p=\{2,3\}}$ represents the first relay that connects piconets P_2 and P_3. Similarly, $R^1_{p=\{1,2,3\}}$ represents the first relay that connects piconets P_1 , P_2 and P_3.

In the following, we will introduce the characteristics of an ideal scatternet configuration and propose the Dynamic Scatternet Restructuring Protocol in follows.

Property 1： $P_i \cap P_j \neq \varnothing, \forall\ P_i, P_j \in Scatternet$

For any pair of two piconets containing one or more common relays, the application of Dynamic Relay Reduction Protocol will not cause disconnection, while the unnecessary relays can be removed effectively. Thus the consumption of Active Member Address can be reduced. The cost for maintaining relays and the probability of packet lost can also can be reduced.

Property 2：A configuration with the least number of relays in a scatternet can be achieved.

For a connected-scatternet, our goal is to guarantee that under the circumstance of a connected-scatternet, the number of relays can be reduced to minimum after performing the Dynamic Relay Reduction Protocol. To prevent a scatternet from disconnection and to obtain the minimal number of relays, the investigated protocol will first preserve the relay with larger degree. The proposed protocol removes relays with smaller degree and effectively explores resources of Active Member Address. For example, in Fig. 2, to achieve a configuration with minimal number of relays, the protocol will firstly preserve the relay with degree 3, such as $R^1_{p=\{1,2,3\}}$, instead of relays with degree 2, such as $R^1_{p=\{1,3\}}$ and $R^1_{p=\{2,3\}}$. Therefore, relays $R^1_{p=\{1,3\}}$ and $R^1_{p=\{2,3\}}$ will give up the relay role and serve only for M_1 and M_2, respectively. As regard to P_2 and P_3, $R^1_{p=\{1,3\}}$ and $R^1_{p=\{2,3\}}$ will no longer play the role of relay. One active member address is therefore released for use of other device. In the following, we will define and describe the table and symbols used in the Dynamic Relay Reduction protocol.

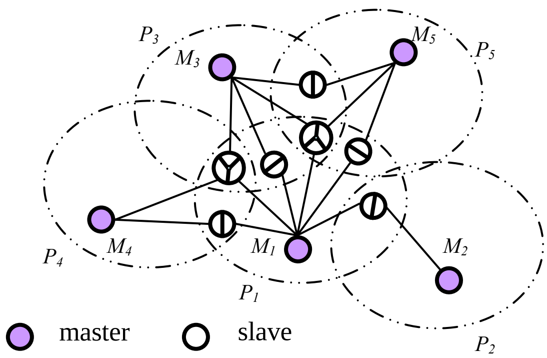


Figure 3. A scatternet before executing the Relay Reduction Protocol.

Definition: Connection Table (*CT*)

For the sake of relay reduction, a Connection Table (*CT*) is maintained in each relay. Devices in the first column of a connection table are relays that connect to those masters to which this relay connects. The first row of this table contains masters connected by relays listed in the first column. The values in the intersection of a row and a column is determined by

$$CT(R_p^k, M_j)=\begin{cases}1 & ,R_p^k \ connects\ M_j \\ null & ,R_p^k \ does\ not\ connect\ M_j\end{cases}$$

As shown in Fig. 3, the CT table maintained by relay $R^1_{p=\{1,3\}}$ is displayed in Table. 1. Because relay $R^1_{p=\{1,3\}}$ connects to M_1 and M_3, the first column of CT table contains relays connected by M_1 or M_3, i.e. $R^1_{p=\{1,2\}}$, $R^1_{p=\{1,5\}}$, $R^1_{p=\{1,3,5\}}$, $R^1_{p=\{1,3\}}$, $R^1_{p=\{1,3,4\}}$, $R^1_{p=\{1,4\}}$ and $R^1_{p=\{3,5\}}$. Because that these relays connect masters M_1, M_2, M_3, M_4 and M_5, the first row of CT records these masters. In Table 1, the values in the intersections of row $R^1_{p=\{1,3,5\}}$ and columns M_1, M_3 and M_5 have values 1. By inspecting Fig. 3, we can see that relay serves for masters M_1, M_3 and M_5. Consider another example in intersections of row $R^1_{p=\{1,3\}}$ and columns M_1 and M_3. The value 1 implies that relay $R^1_{p=\{1,3\}}$ serves for M_1 and M_3. By checking CT table, each relay can determine whether or not itself should remain the relay role such that the number of relays in entire scatternet is minimal and the scatternet is connected.

3.2 Dynamic Relay Reduction Protocol :

Initially, each relay transmits an NRC (Neighboring Relay Collection) message to those masters it connects, and requests each master to collect information about all relays belonging to its piconet. The CT (Connection Table) of a relay thus can be constructed by collecting the response information of those masters served by this relay. For example, take relay $R^1_{p=\{1,3\}}$ into consideration. The relay will send NRC message to M_1 and M_3, since relay $R^1_{p=\{1,3\}}$ connects to M_1 and M_3. On receiving NRC packet, masters M_1 and M_3 will forward the NRC to their relays. For instance, M_1 forwards the NRC message to relays $R^1_{p=\{1,2\}}$, $R^1_{p=\{1,5\}}$, $R^1_{p=\{1,5\}}$, $R^1_{p=\{1,3,5\}}$, $R^1_{p=\{1,3\}}$, $R^1_{p=\{1,3,4\}}$ and $R^1_{p=\{1,4\}}$, and each relay will send back to master M_1 the information of all masters it serves. Thus, on receiving the packet returned from M_1, relay $R^1_{p=\{1,3\}}$ uses the information to construct its CT as shown in Table 1. In Table 1, value 1 denotes the relay in the row connects to the master in the column, while null indicates the relay and the master are not connected. After CT is constructed, each relay can exam its CT and determine whether or not it should abandon the role.

Now, we use Table 1 to illustrate the relay reduction protocol. As shown in Table 1, $R^1_{p=\{1,3\}}$ connects to masters M_1 and M_3. The relay $R^1_{p=\{1,3\}}$ checks from its CT whether or not there exists any other relay connecting to M_1 and M_3 simultaneously. If there exists another relay connecting to M_1 and M_3, $R^1_{p=\{1,3\}}$ giving up its role will not cause P_1 and P_3 disconnection. Therefore, relay $R^1_{p=\{1,3\}}$ should determine whether or not it should abandon its role of relay and determine which piconet it should participate, according to the factor of slave-number balancing. All relays simultaneously execute the same relay reduction procedure will minimize the number of relays and guarantee that the restructured scatternet is connected.

Table 1: Contents of CT Table in Relay $R^1_{p=\{1,3\}}$.

$R^1_{p=\{1,3\}}$	M_1	M_2	M_3	M_4	M_5
$R^1_{p=\{1,2\}}$	1	1			
$R^1_{p=\{1,5\}}$	1				1
$R^1_{p=\{1,3,5\}}$	1		1		1
$R^1_{p=\{1,3\}}$	1		1		
$R^1_{p=\{1,3,4\}}$	1		1	1	
$R^1_{p=\{1,4\}}$	1			1	
$R^1_{p=\{3,5\}}$			1		1

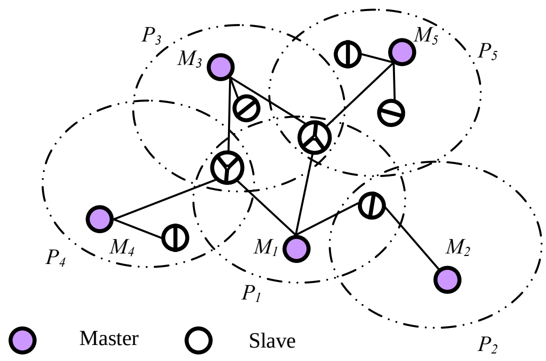


Figure 4. The resultant scatternet after executing the relay reduction operation.

Fig. 4 is the resultant scatternet of the scatternet in Fig. 3, after executing the Dynamic Scatternet Restructuring Protocol. In comparison with Fig. 3, the roles of relays $R^1_{p=\{1,5\}}$, $R^1_{p=\{1,3\}}$, $R^1_{p=\{1,4\}}$ and $R^1_{p=\{3,5\}}$ have been changed to slave. Thus, four Active Member Address are saved, while the remaining scatternet is connected. The reduction of the number of relay also reduces the probability of packet lost and increases the transmission efficiency.

The proposed Dynamic Relay Reduction Protocol is given below.

Step 1: Each relay in a Scatternet broadcasts NRC message to all its connected masters.

Step 2: Those masters M_i who receive NRC message will collect information of relays within its piconet P_i and information of masters that are connected by relays within this piconet $P_{i.}$

Step 3: After each relay received the information provided by those masters this relay connect to, each relay will create a Connection Table. The horizontal axis and vertical axis of CT table denote relays and masters, respectively, where

$$CT(R_p^k, M_j) = \begin{cases} 1 & , R_p^k \text{ connects } M_j \\ null & , R_p^k \text{ does not connect } M_j \end{cases}$$

Step 4: Let *rows* denote the member of rows in *CT*. Relay R_p^k executes the following procedures according to its CT table:

for i = 1 to *rows*
if （CT_i cover my row）
{
1. relay abandons its role of relay and plays the role of slave.
2. $\forall P_i \in P, Let\ P_{\min} = \min|P_i|, \forall P_i \in P$. Relay R_p^k plays role of slave in Piconet $P_{\min}$ and breaks all connection between R_p^k and all masters in piconets $P - P_{\min}$.
}

Applying the Dynamic Scatternet Restructuring Protocol can adjust a connected-scatternet to ideal scatternet dynamically. In the next section, we will describe how to reconfigure the adjusted scatternet for obtaining a new scatternet with characteristics of small number of hop count, disjoint path and QoS services.

4. Routing Protocol for Bluetooth Radio System

In Sec. 3, we have described how to reduce the number of relay dynamically. In this section, a routing protocol is proposed for providing features of small number of hop count, disjoint path and QoS services based on the reduced scatternet. We will first discuss previous works and related research about the Routing Protocol of Bluetooth Radio System. Then, the proposed routing protocol is described.

In Bluetooth Radio System, a scatternet consists of a lot of piconets. In a piconet, a master can serve at most seven active slaves. In a piconet, master and its slaves can communicate with each other in a manner of TDD (Time Division Duplex). However, if two hosts belonging to different piconets are intended to communicate with each other, a routing path should be constructed between different piconets through masters and relays. Due to that every piconet contains at most seven slaves, two hosts intended to communicate to each other may fall in different piconets. For example, in Fig. 5, even if the physical distance between hosts S_1 and S_2 is within ten meters, they may belong to different piconets due to the following cases occur.

Case 1. During execution of inquiry and inquiry scan operations, S_2 matches M_2 first, so that they are not in the same piconet[8].

Case 2. The piconet to which S_1 belongs collects already seven slaves, causing that S_2 cannot participate the same piconet.

Case 3. Even if the piconet containing S_1 has less than seven slaves, S_2 and M_1 may not match during the inquiry and paging operations. This causes S_1 and S_2 are in the different piconet.

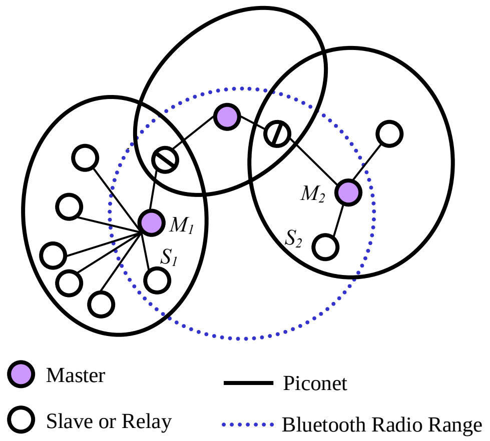
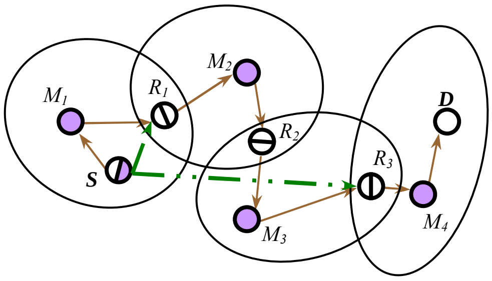


Figure 5. Multi-hops route.

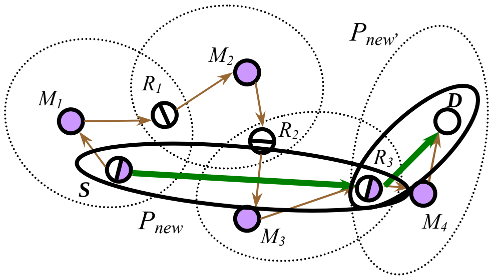
Let hosts S_1 and S_2 be the service provider and consumer, respectively. If they belong to different piconets, an efficient routing algorithm is needed to construct a routing path over this scatternet. In literature, Bhagwat proposed a RVM[4] routing protocol that adopts flooding technique to broadcast the route search packet. On receiving the route search packet, the destination host generates a reply packet for tracing back the route the search packet passing through. The first routing path is thus constructed. As required, a second routing path, containing the common relays of the first path as few as possible, can be constructed. However, of the scatternet comprises of many piconets, the routing path constructed by RVM technique may introduce large transmission latency owing to that the created route has a large number of hop count. Furthermore, a long routing path also consumes the bandwidth resource and generates heavy data traffic. Appling RVM routing protocol to the scatternet shown in Fig. 5, the constructed routing path between S_1 and S_2 will have hop count of six. That is, a packet sent from source host S_1 to destination host S_2, requires passing through six hops. In fact, the distance of S_1 and S_2 may less than one-hop distance (ten meter). In this section, a path reduction protocol is proposed for reducing the hop count of an existed route, while maintaining the original scatternet structure.

4.1 Path Reduction Procedure

As shown in Fig. 6(a), when source host *S* hops to establish a routing path to destination host *D*, it will broadcast a Route Search Packet based on flooding. As soon as the destination host receives the Route Search Packet, it will perform the following Path Reduction Procedures:



（a）Before path reduction procedure



（b）After path reduction procedure.

Figure 6. Path Reduction Procedure.

When a host *D* receives the Route Search Packet and performs route reply packet through reversal routing path, the Bluetooth Address and Clock values of *D* will also be appended in the Route Replay Packet to relay R_3 via master M_4. Meanwhile, *D* will first enter Hold mode in the piconet managed by master M_4, and then enter Page scan mode to construct a new piconet for achieving the objective of shorter routing path. When R_3 receives Route Replay Packet, it appends its Bluetooth Address and clock information to Route Replay Packet. R_3 then transmits Route Replay Packet to master M_3, enters Hold mode in its original piconet, and enters Page scan mode to construct a new piconet to shorten the routing path. Relays R_2 and R_1 will perform the similar procedures when they receive Route Replay packet. Therefore, *D*, R_3, R_2 and R_1 can transmit their Bluetooth Address and clock information to Source Host *S* in final.

When source host *S* receives Route Replay packet, it enters Hold mode and try to play the role of master by entering page mode and constructing new piconets with R_1, R_2, R_3 and *D*. In such a circumstance, source host *S* examines Bluetooth addresses of those hosts that have successfully constructed new piconets with host *S*. In this example, we assume that *S* is in Page mode and can construct connection with devices R_1 and R_3. (Assume that the connections between *S* and R_2 is not established due to frequency mismatching while the connection between *S* and *D* can not be constructed due to their distance is out of Bluetooth Radio Range). Form the received Route Reply Packet, *S* knows that R_3 is the farther relay. As a result, *S* constructs a new piconet P_{new} with R_3, as shown in Fig. 6(b).

After the connection of *S* and R_3 is built, those relays whose sequence order of Bluetooth address in Route Reply packet is less then R_3 will be informed to stop executing the page scan operation and quit the routing service. The route length will be reduced due to the abandonment of R_1 and R_2. The relay R_3 then performs the same procedures as source *S* does so that the subroute from R_3 to destination could be reduced. Consequently, relay R_3 enters page mode and try to construct a new piconet with host *D* that currently stays on page scan mode. After these procedures have been done, a new route $S \rightarrow R_3 \rightarrow D$ is built. The hop count of new route is two, which is shorter than the hop count seven of original route. In Path Reduction Procedure, only BT Address and CLK are required for constructing new link. The time-consuming operations such as inquiry and inquiry scan are omitted. The path reduction procedure will be completed in a very short time.

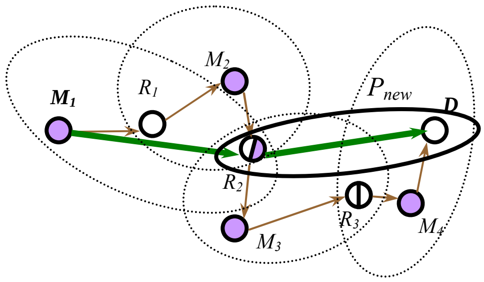


Figure 7. During the execution of path reduction procedure, the processing of source host when it plays the master role

In the method mentioned above, no problem arises even if the role of source host *S* is slave or relay. However, if the source host plays a master role, constructing a new piconet with other relay will violate the rule that two masters cannot coexist in a piconet [1]. To solve this problem, the path reduction protocol should be modified as illustrated in the following. Let M_1 enter page mode and tries to construct connection with R_1, R_2, R_3 and D that are in page scan mode. The host that is closest to destination is chosen and invited to participate the piconet controlled by M_1. Problem that the master M_1 already has seven slaves and has no space to invite another slave participating its piconet will not arise. This is because that master is the one who initiates the request for constructing a routing path. It must be able to control the number of active slaves to avoid the capacity problem. Assume the relay R_2 is invited into the piconet controlled by M_1 successfully and a new piconet P_{new} is constructed in executing the path reduction procedure. The established new route will be : $M_1 \rightarrow R_2 \rightarrow D$, as illustrated in Fig. 7.

In this section, a RVM-based routing protocol is proposed for the reduction of routing path so that the route length of source host S and destination host D is reduced and the bandwidth utilization is exploited efficiently. Protocol for creating disjoint paths will be proposed in the next section. Applying this protocol, the objective of transmission speedup and backup route creation can be achieved.

4.2 Creating the Disjoint Routes

To speed up the data transmission and create backup route, in this section, we propose a protocol for constructing disjoint routes. Keep going on Fig. 6(b), after the route reduction procedure is completed, if the source host raises request of creating multiple routes, we propose protocol to reduce the required time and amount of packets for constructing the multiple disjoint routes. Due to that the information of the 48-bits Bluetooth address and clock values of each relay and destination have been collected in source host, new piconets could be constructed to establish disjoint routes by using the unused relays（R_1、R_2）. The cost of large amount packets due to flooding therefore can be avoided.

Route$_1$：$S \rightarrow R_3 \rightarrow D$
Route$_2$：$S \rightarrow R_2 \rightarrow D$。

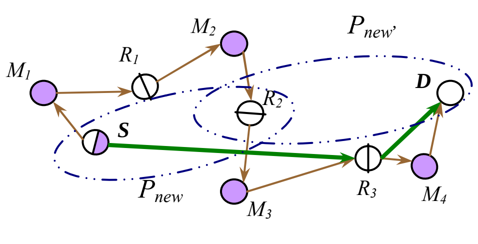


Figure 8. Creation of disjoint Routes.

When source host S is intended to construct the second disjoint path, R_1 and R_2 will stay on page scan mode. During the second route construction, R_3 will abandon page scan mode since it has already participated the first route. The destination host D should enter page scan mode to construct another route. The optimal condition we expect is to construct disjoint routes by similar way as mentioned before. However, there are detailed problems should be taken into consideration.

There are two main purposes for constructing a second route. First, the second route is created for the backup purpose. As soon as some link of the first route is disconnected, the second route can in stead transmit data from source to destination. Another purpose for creating the second route is to speedup the throughput from source to destination. To achieve this goal, the precise difference of hop count of the created disjoint paths should be controlled so that packets transmitted by these two routes can be received in destination continuously without any collision. In the two disjoint routes, hop count is the major factor that determines whether or not packet can be received in destination continuously. To avoid the collision of packets transmission occurred in destination D, the hop count difference of two disjoint routes should be maintained at $2k$ hops, where k is an integer.

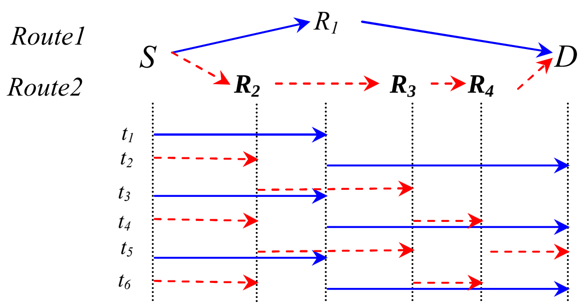


Figure 9. Data transmission of two disjoint routes.

For example, as shown in Fig. 9, the hop counts of two disjoint routes are *Route1*=2 ($S \rightarrow R_1 \rightarrow D$) and *Route2*=4 ($S \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow D$). Their hop count difference satisfies the constraint $2k$, where $k = 1$. Packets therefore could be transmitted in parallel by these two routes. In time t_1, source host S transmits packet 1 on link $S \rightarrow R_1$ of *Route1*. In time t_2, souce host S transmits packet 2 on link $S \rightarrow R_2$ of *Route2*) and R_1 transmits packet 1 to destination by using link $R_1 \rightarrow D$, simultaneously. In time t_3, packet 3 is transmitted on link $S \rightarrow R_1$ and packet 2 is transmitted on link $R_2 \rightarrow R_3$ of *Route 2*. In time t_4, the destination will receive packet 3 from link $R_1 \rightarrow D$ of *Route1* while packet 2 is delivered from link $R_3 \rightarrow R_4$ of *Route2*. Thus, we guarantee that after 4 time steps, the max hop count of routes, packets can be delivered to destination from two routes in different time steps continuously. The source data thus can be tramsmitted on these two routes in parallel.

In addition to the collision avoidance problem, the following problem should also be investigated.

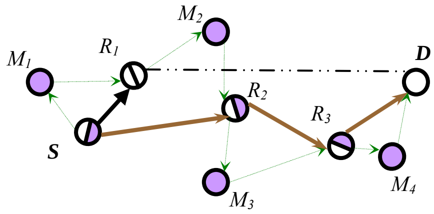


Figure 10. Construction of the second route.

As displayed by Fig. 10, the first constructed Route$_1$ is $S \rightarrow R_2 \rightarrow R_3 \rightarrow D$. Let R_1 be the current relay that executes the second-route construction procedure and there is no relay between R_1 and D that could be used for constructing the second disjoint route. R_1 can not construct a new piconet with destination due to that their distance is out of communication range. To construct the second route, R_1 should inform M_2 to ask one of its slave, whose traffic load is relative low, for entering page scan mode. The selected slave thus could play the role of bridge of R_1 and destination host. As shown in Fig. 11, M selects slave S_{21} to enter page scan mode and transfers the Bluetooth Address and Clock information of S_{21} to R_1 such that R_1 and S_{21} could construct a new piconet. Therefore, we have the following two disjoint routes:

Route$_1$ $S \rightarrow R_2 \rightarrow R_3 \rightarrow D$

Route$_2$ $S \rightarrow R_1 \rightarrow S_{12} \rightarrow D$

With such a mechanism, the probability of successfully constructing the second disjoint path will be increased and the difference of hop counts will be compensated to be *nk* exactly.

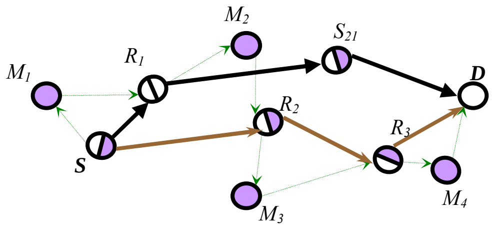


Figure 11. Construction of two disjoint routes.

4.3. *The Routing Protocol*

In previous section, detail of constructing disjoint routes are proposed. Examples are given to illustrate the concepts, constraints, and advantages of the proposed protocol. This section summaries the disjoint routing protocols formally. Given a pair of source and destination devices, the proposed protocol will construct two disjoint routes that are able to transmit data from source to destination in parallel. Length of the constructed routes is short relative to those routes constructed by flooding scheme. Objectives of high throughput, low bandwidth consumption, and low transmission delay are thus achieved. The construction of disjoint routing protocol consists three phases. Phase I tackles with the construction of the first route and prepares information for route reduction while Phase II performs the route reduction so that the constructed first route could consume fewer bandwidth and provide low transmission delay. Phase III aims at the construction of the disjoint route for either speeding up the data transmission or reserving as a backup route. The protocol is illustrated in follows.

Phase I: Construction of The First Route

Step 1: When the source host S hops to create a communication path to a destination device D, it floods the route search packet which contains the information of Source_Host_ID, Destination_Host_ID.

Step 2:On receiving the route search packet, destination host D creates and transmits a route reply packet to source host S in reverse order of the path that the route search packet flooded from S to D. Here, we assume that the path from source S to D is S $[\rightarrow M_0] \rightarrow R_0 \rightarrow M_1 \ldots \rightarrow R_{n-1}$ $[\rightarrow M_n] \rightarrow D$. where M_0 denotes that the second host on the route is possible a master or relay.

Step 3: Once relay R_i receives the reply packet, it appends its Bluetooth Address and Clock to packet and then transfers the packet to its master M_i. Hereafter, relay R_i changes its state from active mode to hold mode in original piconet so that it can enter page scan mode, waiting for the source host's paging.

As soon as source host S receives the route reply packet, it starts the following Phase II operation.

Phase II：Path Reduction Procedure

Let U denote the set of devices on the reduced route. Set U is initially a empty set. Let $\acute{R} = \{R_0, R_1 \ldots, R_{n-1}\}$ denote the set of relay nodes of a route created by flooding.

Step 1: If source node plays the role of Master, as soon as it receives the reply packet, it will enter a page mode and try to invite the destination device or relays in set $\acute{R}$ as a member of the current piconet. If source node does not play the role of Master, this indicates that the source host S plays the role of either slave or relay. In this case, the source host changes to Hold mode in the original piconet, enters page mode for creating new piconet with destination or relays in set $\acute{R}$.

Step 3: Source host takes Δt for connecting with the destination or the relay closest to destination on the route (relay R_i with max value of i).

Step 4: If the source host S successfully connects to the destination host D, a reduced new path $S \rightarrow D$ with one-hop is created. Set U is updated by $U=\{S, D\}$.

Step 5: If host S failed to connect to destination D, it will connect to the relay R_{max}, where R_{max} is the relay with maximal index to which host S is capable connected in set $\acute{R}$. Source host S updates sets U and $\acute{R}$ by:

$$U= \{S, R_{max} \},\ \acute{R}= \acute{R}\text{-}\{R_{max}\}$$

Step 6: If the destination D and all relays in set $\acute{R}$ are far from source S (their distance is larger than 10 meters), steps 4 and 5 can not work. Master M_0 will ask its idle slaves to enter the page mode, trying to connect to destination D or relays in set $\acute{R}$. In case that one of the idle slaves, say S_{0j}, connects to destination D, slave S_{0j} will transmit its Bluetooth address and clock to source S so that host S can construct a new piconet with slave S_{0j}. Source host S then updates set U by $U=\{S, S_{0j}, D\}$ and a reduced path $S \rightarrow S_{0j} \rightarrow D$ is thus constructed successfully. In case that all idle slaves can not connect to destination D, they will try to connect to any possible relay R_{max} in set $\acute{R}$. If the connection is successful, slave S_{0j} will transmit its Bluetooth address and clock to source S so that host S can construct a new piconet with slave S_{0j}. Source host S then updates set U by $U=\{S, S_{0j}, R_{max}\}$ and a reduced subpath $S \rightarrow S_{0j} \rightarrow R_{max}$ is thus constructed. Acting as a source host S, the relay R_{max} then performs again the path reduction procedure so that it can construct a reduced path to destination D. The reduced path is thus constructed.

Step 7: In case that step 6 fails, source host S can not perform the path reduction procedure. The source host S asks R_0 to play its role and to perform the path reduction procedure so that the subpath length from R_0 to destination D could be reduced. In this case, source host S removes from set $\acute{R}$ and update set U by setting $U=\{S, M_0, R_0\}$.

Phase III：Construction of the Disjoint Route

Let $\acute{R} = \acute{R}\text{-}U$. Due to that set U collects all devices on the reduced path, set $\acute{R}$ collects all the devices that appear on the route created by flooding and are not utilized on the reduced path. In Phase III, we utilize the devices in set for constructing a disjoint route. Let $U'== \{ \phi \}$. Set U' is used for collecting all the devices on the disjoint route.

Step 1: Adopt sets $\acute{R}$ set and U' for executing the Phase II procedure so that the disjoint route could be constructed and all devices on the disjoint route are collected in set U'.

Step 2: To ensure that packets transmitted by the first and second routes arrive destination at the different time slot, the difference of hop count of these two route should be $2k$, where k is an integer. That is, U and U' should satisfy the following constraint:

$$|U| - |U'| = 2k$$

In case that the above constraint is not satisfied, source host S applies operations described in step 6 of Phase II to insert additional device S_{0j} on the second route.

In this section, a routing protocol is proposed. The proposed protocol constructs disjoint routes for enhancing the throughput of traffic or preventing from route break. Compared to RVM, length of the two disjoint routes created by the proposed protocol has been largely reduced. In the next section, performance of the proposed protocol is examined.

5. Performance Study

This section proposes the performance investigation of the proposed protocol, in term of the number of piconet, path length, and the number of control packets. The simulation environment is described as follows. The space size is either 20*20, 40*40, or 80*80, while the radio transmission range of a Bluetooth device is set at a constant 10 units. The number of devices varies, from 100 to 140, and their locations are randomly determined.

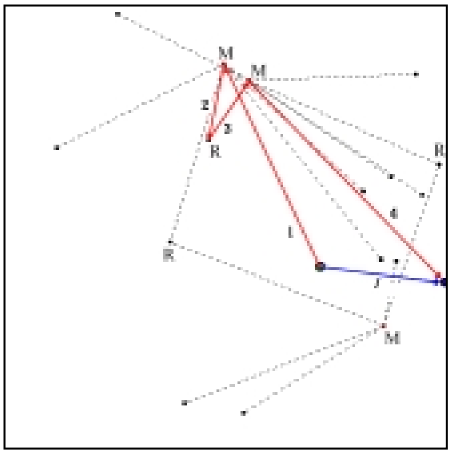


Figure 12. A snapshot of execution.

Figure 12 is a snapshot of the execution result. To observe and compare the behavior of RVM and the proposed protocol, we control the number of hosts to be small on scatternet. As shown in Figure 12, nodes marked by ‘M’ and ‘R’ plays the roles of ‘Master’ and ‘Relay’, respectively. Three piconets construct the scatternet. A dash line connecting two nodes denotes the link between a pair of slave and master. The source and destination hosts, marked by thick and big nodes, are selected from the different piconets. The bold lines denote the routing path where the marked value denotes the link order on the path. As shown in Figure 12, 4-hop route is created by RVM whereas 1-hop route is created by the proposed protocol. The proposed protocol reduces relays on the route found by flooding so that routing length could be minimized.

Initially, 100 hosts randomly construct a connected scatternet in a given space size. The simulator then randomly selects source and destination hosts from the existing hosts. A search packet is created by source host and is flooded for constructing a route, noted by RVM in Figures.

Hereafter, the route reduction procedure is applied to create the reduced route, denoted by ‘proposed protocol’ in Figures. Figures 13, 14, and 15 compare the number of hop of routes created by RVM and the proposed protocol in 20*20, 40*40, and 80*80 respectively. In general, applying the proposed protocol creates a shorter path than RVM. This is because that applying the route reduction removes a number of relays from the route created by RVM. In case that the space size is fixed at 20*20, the maximal distance between any two devices is $20\sqrt{2}$ units. Applying the proposed protocol reduces a number of relays on the route created by RVM so that the distance between two neighboring relays on the reduced path is closed to 10 units. Thus, route created by the proposed protocol has 3 hops at most, regardless of the number of piconets. As shown in Figure 13, route created by the proposed protocol has a constant number of hops, in average. However, the number of hops of route created by RVM scheme increases with the number of piconet since the RVM utilizes the flooding scheme to construct the route. This phenomenon also could be found in Figures 14 and 15 where the space size is fixed at 40*40 and 80*80, respectively.

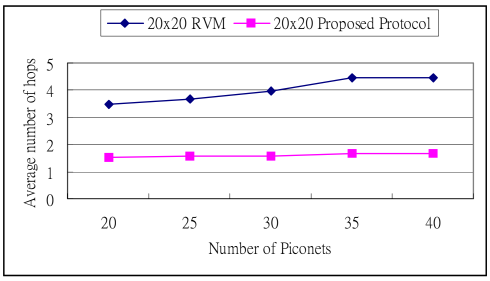


Figure 13. Comparison of the path length under 20*20 space size.

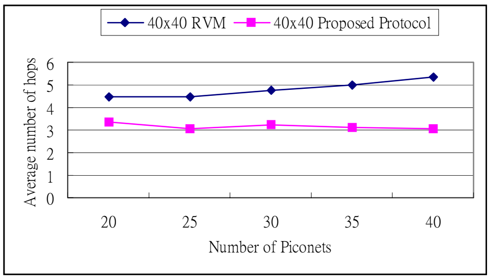


Figure 14. Comparison of the path length under 40*40 space size.

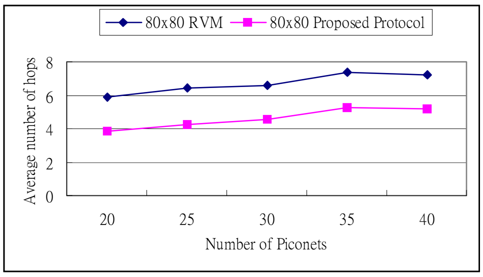


Figure 15. Comparison of the path length under 80*80 space size.

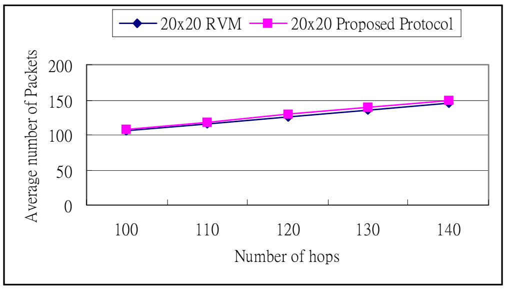


Figure 16. Comparison of the control packet traffic under 20*20 space size.

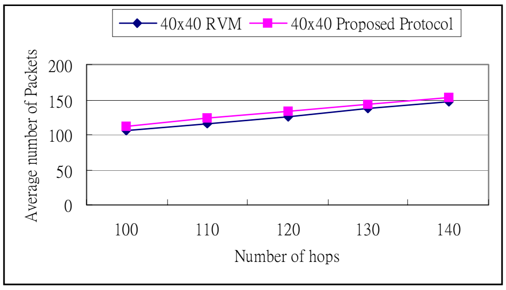


Figure 17. Comparison of the control packet traffic under 40*40 space size.

Figures 16, 17, and 18 compare RVM and the proposed protocol, in terms of control packet traffic. Due to that the proposed protocol performs extra operations for reducing the path length, the control traffic overhead created by the proposed protocol is larger than RVM. However, route created by the proposed protocol is expected to have a smaller amount of data packet traffic, due to that there are fewer relays participating the route. Figure 19 compares the data traffic over routes created by applying RVM and the proposed protocol. The data traffic overhead of route created by the proposed protocol is normalized to 1, provided that the space size is 20*20. Compared to RVM, routes constructed by the proposed protocol create a smaller amount of data traffic.

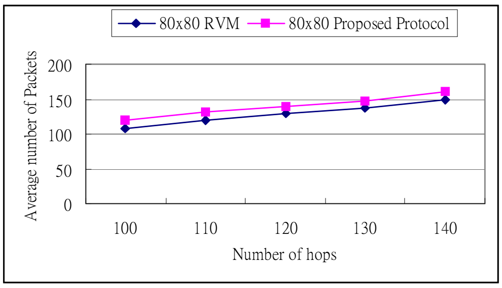


Figure 18. Comparison of the control packet traffic under 80*80 space size.

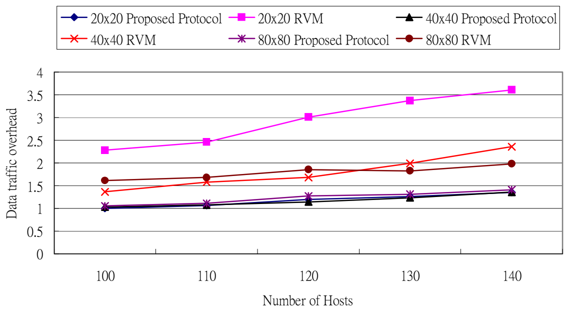


Figure 19. Comparison of the data packet traffic.

6. Conclusions

Bluetooth technology provides a low power, low cost and interference-resistance wireless communication. Constructing an efficient route is the essential requirement for providing inter-piconet communication service. This paper proposes a routing protocol for establishing an efficient route over Bluetooth radio system. To alleviate the flooding overhead and utilize the use of Active Member Address, a distributed protocol is proposed for reducing the unnecessary relays in scatternet. In addition, a routing protocol is proposed for reducing the length of routing path and creating disjoint routes as far as possible. Performance study demonstrates that the proposed protocols are performance well in terms of routing length and data traffic overhead.

Acknowledgements

The authors would like to thank the Ministry of Education, ROC, for partially supporting this research under contract no. 甲 -90-H-FA07-1-4 (Learning Techology) .

References

[1] The Bluetooth Specification, http://www.bluetooth.org 1.0b & 1.1

[2] D. K. Anvekar and M. Kapoor, "Frequency Lookahead And Link State History Based Interference Avoidance In Wireless Pico-cellular Networks," in the *IEEE* International Conference on Personal Wireless Communications (*ICPWC) 2000*, Hyderabad, India, December 2000.

[3] S. Baatz , M. Frank , C. Kuhl , P. Martini , Christoph Scholz, "Adaptive Scatternet Support for Bluetooth using Sniff Mode," *26th Annual Conference on Local Computer Networks*.

[4] P. Bhagwat, A. Segall, "A Routing Vector Method (RVM) for Routing in Bluetooth Scatternets," *The Sixth IEEE International Workshop on Mobile Multimedia Communications (MOMUC'99)*, pp375-379, *Nov 1999*.

[5] P. Bhagwat, "Bluetooth: Technology for Short-Range Wireless Apps," *IEEE Internet Computing 5*(3), (2001).

[6] K. Biesecker, "The Promise of Broadband Wireless," *IT Professional,* Vol. 2, No. 6, December 2000.

[7] S. Galli, K. D. Wong, B. J. Koshy, M. Barton, "Bluetooth Technology: Link Performance and Networking Issues," *European Wireless Conference*, Dresden, Germany, Sep. 12-14, 2000.

[8] T. Lindholm, "Setting up a Bluetooth Packet Transport Link," *http://citeseer.nj.nec.com/392522.html,* Tancred Lindholm Department of Computer Science Helsinki University.

[9] Z. Pei, L. Weidong, W. Jing, W. Youzhen, "Bluetooth-The Fastest Developing Wireless Technology," *WCC - ICCT 2000*. pp1657 -1664, Volume: 2, 2000.

[10] L. Ramachandran, M. Kapoor, A. Sarkar and A. Aggarwal, "Clustering Algorithms for Wireless Ad hoc Networks," *Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications* pp 54-63, August 11, 2000, Boston, MA USA.