A Service Driven Routing Protocol For Bluetooth Scatternets

Abdolbast Greede and Donal O'Mahony
Network And Telecommunications Research Group (NTRG)
Computer Science Department
Trinity College Dublin, Dublin 2, Ireland
{Abdo.Greede,Donal.OMahony}@cs.tcd.ie

ABSTRACT

Bluetooth is a low-cost, low-power and unlicensed radio technology with the potential to become an important platform for ad hoc networking. It is based on a number of point-to-point links between a master node and several slaves that collectively form a piconet. Nodes may be members of more than one piconet simultaneously and by carefully scheduling their presence in each one, can route packets between piconets.

We propose a service-driven routing protocol that allows a mobile bluetooth node to efficiently discover what services are available in surrounding piconets and to form the minimum number of connections necessary to route packets to their destinations. We go on to briefly describe an implementation of this system in both real hardware and a simulated network environment

KEYWORDS

Bluetooth, Ad Hoc Networks, Scatternet Routing, Bluetooth Service Discovery

1. INTRODUCTION

Bluetooth is an emerging low-cost (<$10), low-power (1-100 mW), operating in the 2.4 GHz band [1,2]. It uses Frequency-Hopping Spread Spectrum (FHSS) technology. Bluetooth devices share 79 channels of 1 MHz bandwidth within the 2.4 GHz band. Two or more nodes sharing the same channel form a piconet, where one node acts as a master, controlling the communication in the piconet, and the others act as slaves.

Piconet formation takes place by establishing a connection between a master device and slave device through "inquiry" and "page" stages. The "inquiry" stage identifies the available devices, while the "page" process connects the master with the slaves discovered. Once a link is formed and a connection is made, a Service Discovery Protocol (SDP) is established, allowing Bluetooth devices to discover what other services Bluetooth devices can offer

The two major modes in Bluetooth are standby (listening on periodic time-slots for inquiries), or Active (part of a piconet). In addition three power-save modes have been defined: SNIFF, HOLD and PARK.
SNIFF mode: a slave reduces its duty cycle during periods of low activity. So the slave listens at every $_{TSNIFF}$ interval only, but is still considered an active member of the piconet.
HOLD mode: a transceiver unit neither transmits nor receives information with its current piconet. Prior to entering the HOLD mode, the master and slave agree on the time duration that the slave will remain in the HOLD mode.
While in **PARK mode**, a slave enters a low-power mode of very little activity. It no longer participates on the piconet channel, so it gives up its active member address AM_ADDR.

Since the radio-link is based on multiple channels, multiple piconets can co-exist in the same area without interfering with each other. A group of overlapping piconets is referred to as a scatternet.

The current Bluetooth specification has defined the scatternet but leaves open details on scatternet formation. It suggests putting links in power-save mode in order to "leave" a piconet and to be able to visit another piconet. If devices are connected to multiple piconets (scatternet), then since the hop-selection and timing for each piconet is independent, a device has to switch between the piconets to communicate with each piconet. Therefore negotiation of the communication schedule between the piconets would be required. Many approaches have been presented to solve the problem of scatternet formation.

In addition to scatternet formation there is another problem not addressed in the current specification, which is the routing protocol used in the scatternet. The routing problem deals with delivery of packets in such a scatternet to enable multihop ad-hoc networking. There has been little work done in this area of scatternet routing.

In this paper we study the problem of scatternet formation, Routing and service discovery in a Bluetooth network. Service Driven Routing Protocol (SDRP) is proposed. Simulation exercises are conducted to evaluate the performance of our approach.

2. RELATED WORK

The following subsections outlines some of related contributions, and work dealing with both scatternet formation and routing:

2.1. Scatternet Formation

Baatz et al. [3] present a scatternet formation method based On Sniff Mode. The objective of the sniff mode approach for scatternet formation is to make the communication schedule dynamically (i.e. not a priori) for each communication period. Since communication between two devices continues as long as data is exchanged in sniff mode, it has a dynamic nature that makes it suitable for scatternet scheduling. This approach is able to rapidly adapt to changing traffic conditions, as it requires no scatternet-wide coordination. The accounting scheme used provides link level fairness whilst it is able to reallocate unused bandwidth. We have adopted this approach in our solution as we think it has a great potential to solve the problem.

Law et al [4] have studied scatternet formation in the situation where the devices are in-range of one another. Their approach involves reorganizing these devices into a new scatternet formation. Their main proposal is to eliminate as many of the scatternet structure as possible, reducing interference by merging some piconets. However the resulting network is no longer a scatternet, and it does not serve the main objective of a scatternet, which is to maximize the available bandwidth. At the same time interference is not a big issue with FH Technology, and so reducing it provides little extra benefit.

2.2 Scatternet Routing

Building a Scatternet offers Bluetooth the potential to become a multihop ad hoc network. Routing in multihop ad hoc networks has been the focus of much research for the last few years. Therefore one of the possible solutions is to apply the exiting ad hoc routing protocols to provide routing within a Bluetooth scatternet. However the Bluetooth environment presents new challenges.

For example, two Bluetooth nodes cannot hear each other unless they form a master-slave pair. Therefore traditional reactive routing protocols can't be applied. Also source-routing protocols need to store the full source-route within the Bluetooth packet. If the 48-bit Bluetooth address is used it could create a huge overhead. Bluetooth technology is expected to be used in a wide range of electronic devices with different service capabilities. In such network environments service discovery often represents the core objective of any routing. Traditional mobile ad hoc network (MANET) routing protocols like Cluster Based Routing Protocol [5], don't deal with this issue, so they may not be well suited for Bluetooth.

Some research has attempted to adopt ad-hoc routing methods to provide scatternet routing. Bhagwat et al [6] proposed a routing method, RVM (Routing Vector Method), for Bluetooth scatternet structure. It is based on the concept of route vector (source routing). In their approach requests are propagated through all piconets. This approach creates a lot of overhead traffic and requires a lot of piconet switching.

3. SERVICE DRIVEN ROUTING PROTOCOL (SDRP)

Bluetooth piconet formation requires two stages: "inquiry" and "page". The "inquiry" stage identifies the available devices in the vicinity. If any nearby Bluetooth device is listening for these inquiries it responds by sending its address, clock information, Class of Device (CoD), and other information in a Frequency Hopping Synchronize (FHS) packet. The information in the CoD field is subdivided into three separate parts as follows: the major service class, the major device class, and the minor device class. All the responding devices are potential slaves, and this CoD field can be used to filter this list. It provides an indication of the services provided by each device. After receiving all inquiry responses, the potential master usually has two lists, one containing the devices whose CoD service field match the requested service, and the other containing devices offering different services. The "page" process then connects the master with the slaves in the first list that have been chosen. These connected slaves are called the Active members.

Once a link is formed and connection is made, a Service Discovery Protocol (SDP) dialogue is established, allowing Bluetooth devices to fully discover what other Bluetooth devices can offer (i.e. services). SDP provides the final stage in filtering the list of slaves. Figure 1. shows the piconet formation stages.

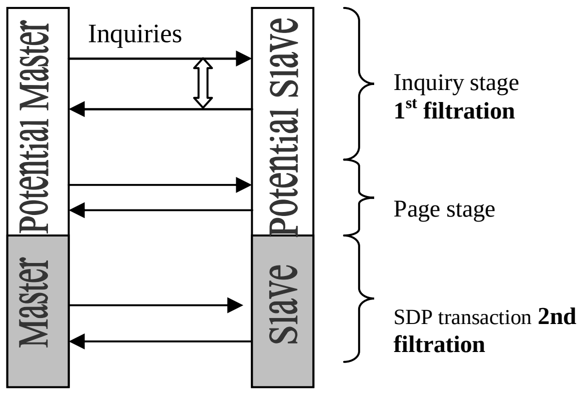

Figure 1: Piconet Formation

Once the piconet formation has been completed the master will have a list of active members. We call this the Active table (A-table). Usually the A-table contains the active address and services support by that device. While the master builds the A-table it may exclude some potential slaves during the filtration processes. Usually these slaves are maintained in another list, which we call the history table (H table)

When a device wants to discover a route to the service in the piconet (out of physical range), or within the scatternet network, multihop is required. We define the multihop level as the number of hops a node is willing to let its service scan propagate to. This should be a configurable option for the end user.

Because Bluetooth has the potential to enable a wide range of devices, we will expect to see a wide range of Bluetooth devices with different configurations. Some devices, such as access points, work only in slave mode by nature, whereas others are more suited to working in full slave/master mode. For some devices a multihop scan-level of more than 0 would be suitable, for example a PDA searching for a printing service may be willing to search a number of hops. However a keyboard may not be willing to hop through another device to its destination.

We have classified the Bluetooth devices depending on their configuration:

Multihop Scan Node (MSN): The Multihop level is a configurable option that should be available in any Bluetooth devices. It represents the number of hops a node is willing to let its service scan propagate to. For MSN nodes this is more than 0.

Relay Service Node (RSN): This class of node represents devices within a scatternet that are willing to act as a relay between piconets. Not all nodes in a scatternet may be able to efficiently work as a relay. For example if an Internet access point were to have to relay packets between piconets it would create a delay that would affect its performance.

Generous Master Node (GMN): Some Bluetooth devices may have no power constraints, and if these devices are working in master mode they can offer another option for requested services not present in the active piconet, but which they are aware of through their H-Tables.

3.1 Table Distribution

The master is aware of its piconet members and it can operate as a knowledge server for its slaves. The master fulfills the same role as Directory Agent within SLP[10].For some slaves it is more efficient to have a copy of these member tables. Both the MSN and RSN nodes can act more efficiently if they contain the member tables. The advantages derived by the MSN nodes are fairly limited, simply reducing the need to contact the master node when requesting a service. However, by supplying the RSN nodes with the tables, the efficiency of inter-piconet communication can be significantly improved.

By distributing the tables to the RSN nodes these nodes are provided with full knowledge of the services provided by the piconet it is joined to. By definition RSN nodes are likely to be members of more than one piconet simultaneously. They will then contain member tables for each of the piconets to which they are connected. If a MSN node in one of those piconets makes an inquiry the RSN node can respond with information about services in one of the other piconets it is connected to, without having to actually switch piconets to ask the master to provide this information. Switching piconets would introduce a lot of time delay. By reducing this delay for inquiries the efficiency of inter-piconet communications will be significantly improved.

A multihop value of 1 enables communications within a single piconet (intra-piconet communication), while a multihop value greater than 2 allows communication with devices within the scatternet.

3.2 Requiring A Service Within Piconet

In any given piconet, MSN and RSN nodes contain knowledge about the services offered by other slaves. Consider a case where a slave, B, of the piconet hosts the service required by slave A. To discover and use this service a connection has to be established between Slave A and Slave B. If Slave A and Slave B are in physical proximity of each other (Figure 2.) then it is more efficient to form a new piconet, providing increased bandwidth. Slave A can request synchronization information (clock and Bluetooth address) held by the master to bypass the inquiry stage and so the new piconet can be formed more efficiently.

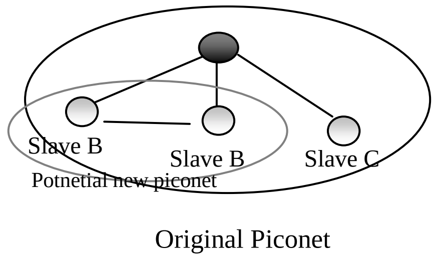

Figure 2: Potential new Piconet Case

However, if the slaves are not in the direct range of each other (Figure 3.) Intra-piconet communication can be used to enable such communication. Because the slave is aware of the location of the service, a fairly straightforward source-routing technique can be applied. Packets are relayed through the master using a forward flag (FF) with value 1. This implies that the packet should be forwarded to the corresponding slave in the same piconet (the forward flag reflects the multihop level).

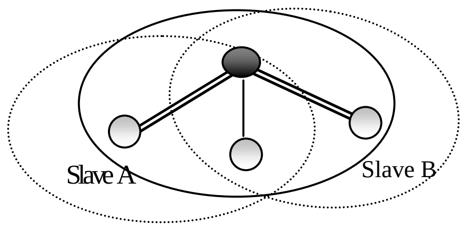

Figure 3: Intra Piconet Case

3.3 Requiring A Service In Scatternet

In another case the requested service may not be available in the piconet. However it may be available in a neighbouring piconet (within the scatternet), which is reachable through one or more relay (RSN) nodes.

In our approach we assume that scatternet structure is already in place, having been formed using the sniff-mode approach[3], Our approach is one that allows nodes in a scatternet to take advantage of the fact that one of the nodes in the same piconet is also a member of another piconet. This node could be used to build communications between piconets and increase the potential range of communications.

There are two possible scatternet structures: one where the relay node is a slave in each piconet, and the other where the relay node is a master in one piconet, and a slave in another. A study [7] has shown that the first structure is more efficient for multihop.

In this scenario the requested service may not be available in the piconet, but may be available in a neighbouring piconet. In the following example (Figure 4) a laptop (a MSN node) in one piconet wishes to contact a printer service. There is none available in its own range, and none even in the same piconet. However one of the nodes in that piconet is willing to act as a relay to another piconet, which does contain devices willing to provide the service.

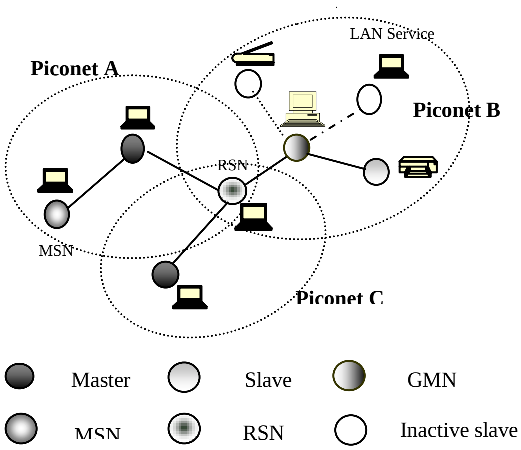

Figure 4: Scatternet Example

As specified in the table-distribution section, the RSN and MSN nodes will contain copies of the A table for their piconets. The laptop (a MSN node) is aware of the services in its own piconet and knows that the printing service it requires is not available. However it is also aware that there is an RSN node in the piconet, and so can extend the inquiry by sending the request to this node.

A Service Specific Inquiry is sent to the RSN node. By consulting its tables it can see that the printing service is available in piconet B, and so replies to the inquiring node with the required forward flag to reach the service, information concerning the sniff intervals used for switching piconets (as this will effect the delay time), and a unique ID that should be supplied with future data packets.

The RSN node keeps a table mapping this ID to the corresponding partial route. This cuts down the overhead in the packet headers by not requiring a full source-route for each packet.

In another example the RSN node may not have contained an entry for the requested service, and so could have forwarded the inquiry to other RSN nodes in its other piconets, if the forward flag is sufficient.

/* when an RSN node receives service enquire message */
if RSN is aware of service
    Send back response to MSN node
else
    if forward flag is sufficient for forwarding
        forward the inquire to the next RSN
    else
        respond saying no node is reachable

The use of a Generous Master Node (GMN) can improve performance. This uses the H-Table to effectively "cache" information about other services in the piconet, which the master is not currently connected to. In the previous example assume the master in piconet B is a GMN, and so will also distribute its H table to the RSN node. This table will contain records for the LAN Service and scanner, even though they aren't connected to the piconet. So if the original inquiry was for LAN service, the RSN node can take advantage of the H-Table

3.4. Table-Maintenance and Selection of the Route

The response to the inquiring node includes the Sniff interval (affecting the time delay for switching between piconets) and FF value (the number of hops the service is away) required to reach the service. This represents the crucial information required to make the appropriate path selection. The path selection could be:

- **Manual Selection:** Presents the list of responses to the user and leaves the selection process up to him/her.
- **Automatic Selection:** Use some pre-defined algorithm to select the most suitable service – as regards minimizing the time delay required to reach the service.

The master will naturally keep his table up-to-date through periodically polling his slaves. Therefore changes in its table should cause a corresponding update in the tables that have been distributed to the slaves. This causes no additional traffic as periodic polling was already required to maintain slave synchronization to the channel.

4. IMPLEMENTATION

In this section, we examine the performance of our approach for scatternet routing. Only intra-piconet communication has been implemented in real hardware due to limitations in the available hardware at the time of development.

Instead simulation exercises have been conducted for the service-driven routing protocol, using the Network and Telecommunication Research Group (NTRG) labs and facilities at Trinity College. We have conducted the simulation within the ad hoc wireless stack developed by the NTRG [8]. The NTRG stack was chosen because of its simplicity and extensibility. It has been extended to include link and SDP layers, providing inquiry, paging, and service-discovery functionality. The stack also supports the JEmu [9] radio simulator, greatly simplifying the creation and running of test scenarios with several devices for debugging and evaluation. The JEmu simulator was modified to emulate a simple Bluetooth radio layer.

We implemented an approximation of the bluetooth media access control layer which allowed for piconet formation and inter-piconet packet relay. Our implementation of SDRP ran over this.

Figure 5 shows a screenshot from our simulator. For simplicity within the simulation Bluetooth addresses were represented by 7-bit addresses, instead of 48-bit Bluetooth addresses. This screenshot shows a very simple configuration of 5 nodes, representing two piconets in a scatternet structure.

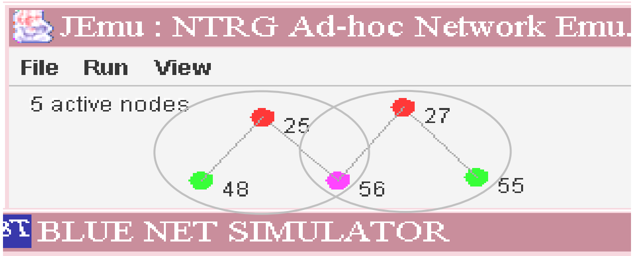

Figure 5: Simulator Screenshot.

In Figure 5, each of the circles (48,55) represents the slaves, while the circles (25,27) represent the masters, and the circle (56) represents the RSN node. Each of the devices includes the stack described above. Node 48 requests a service that isn't listed within its tables.

By modifying its multihop value, Node 48 can limit the search for this service to those nodes within its piconet (25, 56) or extend the search to piconets that are direct neighbours. Using a multihop value of 2 will locate the service if it exists in nodes 27 or 55. Note that in cases where the service is not available, it will not be necessary to cause node 56 to switch between piconets or for any inter-piconet packets to be sent. This cuts down substantially on unnecessary and time-consuming network traffic and makes ad-hoc networking with Bluetooth a viable proposition.

Because of the unique way SDRP works (merging the service discovery and routing), significant improvements have been achieved over related protocols. By cutting down the number of hops needed for device and service discovery, unnecessary traffic is eliminated, interference is reduced and there is less power consumption. By cutting down the number of switches needed, less delay time is also achieved.

5.CONCLUSIONS AND FUTURE WORK

The current Bluetooth technology has limitations, in relation to routing and service discovery in nodes that are physically beyond each other's range, even if those nodes are reachable through relay (bridge) nodes.

We have proposed Service Driven Routing Protocol, a routing algorithm that provides service discovery for multihop Bluetooth networks with less delay time (minimize piconet switching and propagate scan hop requirements) and less traffic overhead (partial source route). The protocol fulfills the special requirements of Bluetooth networks and offers an efficient routing approach for the Bluetooth networking environment.
Future work could include Route Knowledge Inheritance where the relay nodes will cache the information received from responses. Using this optimisation propagation throughout the network is not required to satisfy future requests for the same service. This is particularly important in a Bluetooth environment where switching between piconets can prove time-consuming.

Optimisations may help reduce the delay when relaying packets. One such improvement would assign a higher priority to packets that are being forwarded.

6. REFERENCES

[1] Bluetooth Specification Volume 1 & 2. Bluetooth SIG. Version 1.1, Feb. 2001, http://www.Bluetooth.com.

[2] Brent A. Miller Chatschik Bisdikian, Bluetooth Revealed, Prentice Hall, NJ, September 2000.

[3] Simon Baatz et al, "Adaptive Scatternet Support for Bluetooth using Sniff Mode", Proceedings of the 26th Annual Conference on Local Computer Networks, LCN 2001, Tampa, Florida, November 2001.

[4] Ching Law et al, "Performance of a new Bluetooth scatternet formation protocol", Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing 2001, Long Beach, California, USA, October 2001.

[5] M. Jiang, J. Li, and Y. Tay. "Cluster Based Routing Protocol (CBRP) Functional Specification (draft-ietf-manet-cbrp-spec-00.txt)". MANET Internet Drafts, August 1998.

[6] Bhagwat and Segall, "A Routing Vector Method (RVM) for Routing in Bluetooth Scatternets, in Mobile Multimedia Communications", 1999. (MoMuC '99). 1999 IEEE International Workshop on,1999, Page(s): 375 –379.

[7] Kalia, S. Garg, R. Shorey, "Scatternet Structure and Inter-Piconet Communication in the Bluetooth system", IEEE National Conference on Communications 2000, New Delhi.

[8] O'Mahony, D. & Doyle, L., An Adaptable Node Architecture for Future Wireless Networks, in Mobile Computing: Implementing Pervasive Information and Communication Technologies, Kluwer Publishing, August 2001

[9] Juan Flynn et. al. "A real time emulation system for ad-hoc networks" In Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference 2002 (CNDS '02), San Antonio,Texas, January 2002.

[10] Erik Guttman, "Service location protocol: Automatic discovery of IP network services," IEEE Internet computing, July 1999, Page(s) 71-80.