Video Streaming over Overlaid Bluetooth Piconets (OBP)

Sewook Jung
Department of Computer Science
University of California, Los Angeles
3771B Boelter Hall
+1-310-206-8589
sewookj@cs.ucla.edu

Alexander Chang
Department of Computer Science
University of California, Los Angeles
3771B Boelter Hall
+1-310-206-8589
acmchang@cs.ucla.edu

Mario Gerla
Department of Computer Science
University of California, Los Angeles
3732F Boelter Hall
+1-310-825-4367
gerla@cs.ucla.edu

ABSTRACT

In a large scale Bluetooth network, scatternet has been regarded as the only interconnection method among piconets. But, most Bluetooth devices do not support scatternet connection. Moreover, in high mobility situations, scatternet is not useful because of frequent disconnections and reconnections. Overlaid Bluetooth Piconets (OBP) interconnects piconets and forms a virtual scatternet. This demo shows the possibility of using OBP instead of Scatternet for Video Streaming.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless Communication – *Wireless Personal Area Network, Bluetooth Piconet interconnection.*

General Terms

Algorithms, Measurement, Performance, and Experimentation.

Keywords

Bluetooth, Piconet, Scatternet, OBP, and Video Streaming

1. INTRODUCTION

Bluetooth is a short-range wireless network technology that supports ad-hoc network. Up to eight nodes are organized in a star-shaped cluster, called *piconet*. The cluster head is called *master* and the other nodes are called *slaves*. Two *slaves* cannot transfer packets directly. So, *Master* should intervene between two *slaves*, when *slave* transmits packets to the other *slave*. *Piconets* are interconnected through *bridge* nodes and interconnected *piconets* form a *scatternet*. *Bridges* are the nodes participating in more than one *piconet* with a time-sharing method.

Bluetooth data communication usually uses Asynchronous Connectionless Links (ACL) that has time slots of 625μs. Data packets may use 1,3, or 5 slots and they may be Forward Error Coded (FEC). FEC packets are DM1, DM3, and DM5 (with the digits indicating the number of slots used). The non-error coded ones are DH1, DH3, and DH5 [1].

Many Bluetooth chips are produced and already installed in many personal devices such as Laptop, PDA, and Cellular phone. But, scatternet connection is not supported in all Bluetooth chips. Lack of scatternet connection requires different interconnection techniques. *Overlaid Bluetooth Piconets (OBP)* enables network services for mobile users without Bluetooth Scatternet. Bluetooth nodes first form several piconets, and OBP forms a virtual scatternet later.

WiNTECH'06, September 29, 2006, Los Angeles, California, USA.
ACM 1-59593-538-X/06/0009.

2. Overlaid Bluetooth Piconets (OBP)

Overlaid Bluetooth Piconets (OBP) does not require scatternet connection. So, all Bluetooth devices can use OBP and form a virtual scatternet, even if they do not support scatternet. OBP can be used for the network that has challenging conditions, such as frequent disconnections, or long delays due to mobility of nodes [2].

Consider that we are using scatternet unsupported Bluetooth devices. When piconet is formed, slave nodes cannot communicate with the outside piconet nodes. Master nodes can do inquiry and check free nodes (unconnected nodes) in the communication range. Slave nodes cannot do inquiry scan after its connection to a master. So, to do inquiry-scan or to be connected to another master, a slave node should disconnect the connection and become a free node.

Each piconet continuously changes its stages. *Slave stage, Probe Stage, Return Stage* and *Transfer Stage* are used, and they form OBP Period as shown in Figure 1.

In *Slave stage*, every node keeps its original piconet connection and intra-piconet transfers are made. Some nodes may not have any piconet connection. These nodes remain as free nodes and

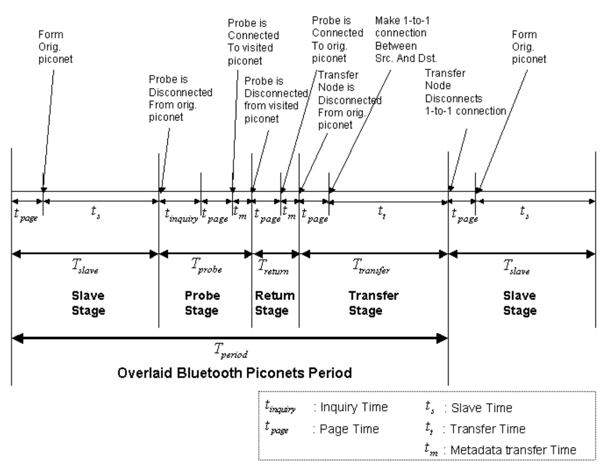

Figure 1. Overlaid Bluetooth Piconets (OBP) Period

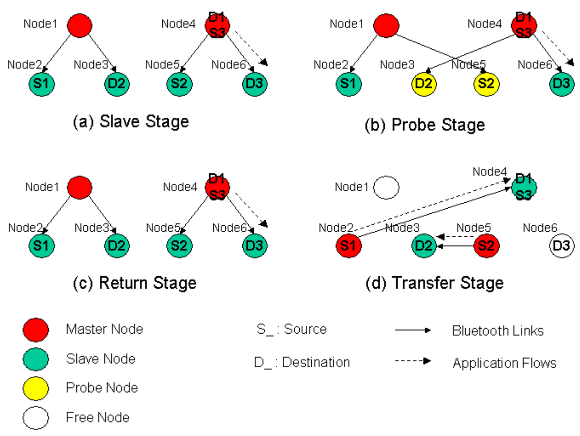

Figure 2. Overlaid Bluetooth Piconets Stages

denoted as *singleton nodes*.

In *Probe stage*, one slave is disconnected from each piconet and performs inquiry scan and we denote this slave as *probe node*. Master nodes perform inquiry and find out which probe nodes are available in the communication range. If a master node finds a probe node, master connects it. Several probe nodes may be detected at the same time. In this case, master node should decide which one to choose among them. At the first *Probe stage*, master node randomly chooses one probe node and connects it. At the later *Probe stages*, master chooses a probe node that is not connected before. If all probe nodes are connected before, master chooses the probe node that is connected earlier than other nodes. Master node keeps probe node connection log (bd-address and connection timestamp). *Singleton nodes* have 50% chance of doing inquiry scan and 50% chance of doing inquiry. Thus in this stage, one probe node in each piconet is disconnected and connected to another piconet (probed piconet). After connection, a probe node transfers metadata to nodes in the probed piconet and finds out whether there is useful data or not. If there is data to transmit, probe node and probed piconet nodes synchronize transfer start time and decide which node will send and receive.

In *Return Stage*, probe nodes are disconnected from probed piconets and return to the original piconet. Inquiry is not included in this stage because master node already knows that probe node is in the communication range. After connection to the original piconet, probe node gives metadata received from the probed piconet, and gives information of which nodes are used in the transfer time and when it is started.

In *Transfer Stage*, inter-piconet transfer related nodes are disconnected from the original piconets. If a master is related to this transfer, it will disconnect all of its slaves. After disconnection, source nodes connect destination nodes and transfer data. Inquiry is also not needed for this because source nodes already know that destination nodes are in the communication range.

After *Transfer Stage*, source and destination nodes return to their original piconets and OBP enters *Slave Stage*.

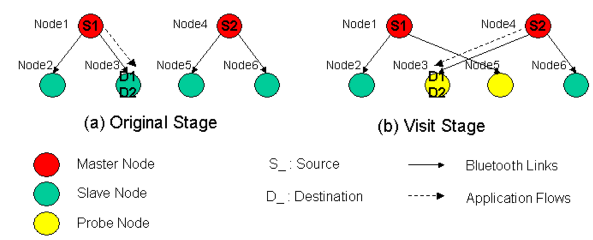

Figure 3. Video Streaming over OBP

Each node in the piconet changes its role according to stages in OBP Period. Figure 2 shows each stage. At Slave Stage, only intra-piconet transfer is possible. So, only the flow from S3 to D3 can be transferred. The flow will remain until Transfer Stage is started. After connection to a probed piconet, probe node and probed piconet nodes exchange metadata. Synchronized transfer time will be assigned at this time. At Return Stage, probe nodes return to their original piconets and give metadata (received from probed piconet) to their piconet nodes. At Transfer Stage, source and destination nodes are disconnected from original piconets. Source nodes make connection to destination nodes and start inter-piconet transfers such as S1 $\rightarrow$ D1 and S2 $\rightarrow$ D2.

3. Video Streaming over Overlaid Bluetooth Piconet (OBP)

Multiple One-to-one connections show higher capacity than that of scatternet [3]. If proper buffering is provided, video streaming over OBP shows better performance than scatternet. Buffering enables continuous streaming when connections are disconnected. We use Mplayer [4] for video streaming on Linux environment. It supports multiple video formats and buffering.

OBP is programmed as application and nodes are changing their state continuously. We assume that destination nodes already know source nodes and source nodes also know destination nodes. Therefore, we eliminate *Probe Stage* and *Return Stage*. Stages are changed as *Original Stage* and *Visit Stage* in Figure 3.

Probe nodes (Node3 and Node5) are connected to Node1 half time and Node4 half time. During *Original Stage*, Node3 receives data from Node1. Remaining data after showing video stream are saved on the buffer and use them during *Visit Stage*. Remaining data from Node4 are used same way as that from Node1.

4. REFERENCES

[1] Bluetooth SIG,. *Bluetooth Specification v2.0*, 2004

[2] Sewook Jung, Alexander Chang, and Mario Gerla, “Performance comparison of Overlaid Bluetooth Piconets (OBP) and scatternet”, *IEEE Wireless Communications and Networking Conference (WCNC),* 2006

[3] Sewook Jung, Alexander Chang, and Mario Gerla, “Comparison of Bluetooth interconnection methods using BlueProbe”, *The Second International Workshop on Wireless Network Measurement (WiNMee),* 2006

[4] MPlayer movie player, http://www.mplayerhq.hu