Performance Comparison of Overlaid Bluetooth Piconets (OBP) and Bluetooth Scatternet

Sewook Jung, Alexander Chang, and Mario Gerla
Department of Computer Science
University of California, Los Angeles
Los Angeles, USA
{sewookj, acmchang, gerla}@cs.ucla.edu

Abstract - In a large scale Bluetooth network, scatternet is 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. We propose Overlaid Bluetooth Piconets (OBP) to interconnect piconets and form a virtual scatternet. Every piconet continuously changes its stages and collects metadata of piconets in the communication range. If metadata shows existence of data to transfer, an inter-piconet connection is made and data is transferred. We compared throughput and efficiency of OBP with those of scatternet. Results show the feasibility of OBP usage instead of scatternet.

I. 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 $625\mu s$. 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. The latest Bluetooth Specification 2.0 introduces the Enhanced Data Rate (EDR) packets and they are 2-DH1, 2-DH3, 2-DH5, 3-DH1, 3-DH3, and 3-DH5. The 2-DH(1,3,5) and 3-DH(1,3,5) packets are similar to DH(1,3,5) but uses $\pi/4$-DQPSK and 8DPSK modulations, respectively [1]. Bluetooth packet information is described in Table I.

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.

We propose *Overlaid Bluetooth Piconets (OBP)* enabling network services for mobile users without Bluetooth Scatternet. Bluetooth nodes first form several piconets, and OBP forms a virtual scatternet later. By using OBP, each Bluetooth piconet can collect metadata from the piconets in the communication range. Metadata contains information on transmission nodes, file names, and synchronization times. If there is a real data to transfer between piconets, it will be transferred afterward.

TABLE I. BLUETOOTH ACL PACKETS

Type	Payload (bytes)	FEC	Symmetric Max Rate (kb/s)	Assymetric MaxRate (Kb/s) Forward	Assymetric MaxRate (Kb/s) Backward
DM1	0-17	2/3	108.8	108.8	108.8
DH1	0-27	No	172.8	172.8	172.8
DM3	0-121	2/3	258.1	387.2	54.4
DH3	0-183	No	390.4	585.6	86.4
DM5	0-224	2/3	286.7	477.8	36.3
DH5	0-339	No	433.9	723.2	57.6
2-DH1	0-54	No	345.6	345.6	345.6
2-DH3	0-367	No	782.9	1174.4	172.8
2-DH5	0-679	No	869.7	1448.5	115.2
3-DH1	0-83	No	531.2	531.2	531.2
3-DH3	0-552	No	1177.6	1766.4	235.6
3-DH5	0-1021	No	1306.9	2178.1	177.1

This paper has two main contributions. First, we describe the idea of Overlaid Bluetooth Piconets and how it can be applied in the Bluetooth devices already in use. Second, we describe the feasibility of OBP by simulation results that are compared to that of Bluetooth scatternet simulation.

II. 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. In [2], overlay architecture is used to support similar networking conditions. In [3], Bluetooth devices measure real-world mobility patterns.

Forming a scatternet requires special scatternet formation algorithms. Even if a scatternet is formed, user's mobility disconnects the initial scatternet and frequent reconnections are needed. So, scatternet connections are not always useful in high mobility situations.

Consider that we are using scatternet unsupported Bluetooth devices. When a piconet is formed, slave nodes cannot communicate with outside piconet nodes. Master nodes can do

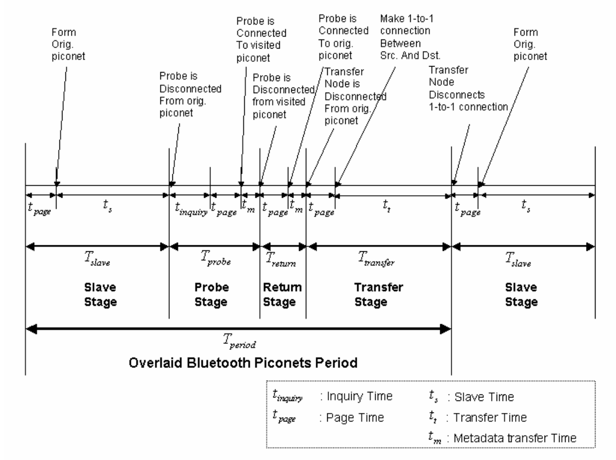

Figure 1. Overlaid Bluetooth Piconets (OBP) Period

inquiry and check free nodes (unconnected nodes) in the communication range. Slave nodes cannot do inquiry scan after their connections to a master. So, to do an inquiry scan or to be connected to another master, a slave node should disconnect from its master node and become a free node.

Each piconet continuously changes its stages. *Slave stage, Probe stage, Return stage,* and *Transfer stage* are used in this sequence, 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 are 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 to 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 to 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 an inquiry scan (acting as a probe node) and 50% chance of doing an inquiry (acting as a master node). Thus in this stage, probe nodes are created to be connected to other piconets (probed piconets). After the 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 the probed piconets and return to their original piconets. 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 conveys metadata received from the probed piconet and information about which nodes are used in the *Transfer stage* and when it is started.

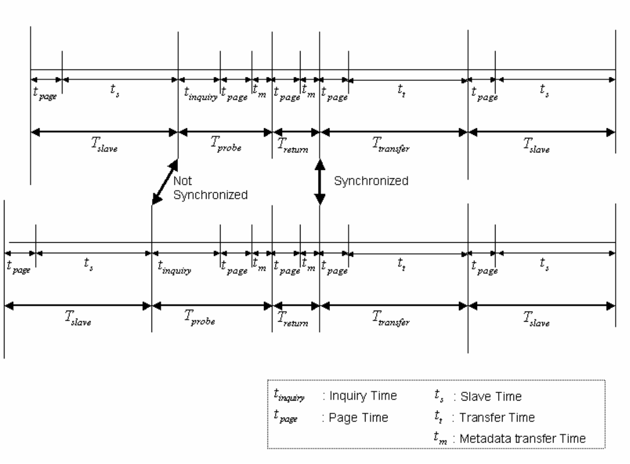

Figure 2. Synchronization between piconets

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*.

Two piconets may not be synchronized in the *Slave stage*. However, after a probe node is connected to the probed piconet, the probe node will receive exact synchronization point from the probed piconet. Two piconets can be synchronized after the *Transfer stage*. Figure 2 shows synchronization between piconets.

Each node in the piconet changes its role according to stages in OBP Period. Figure 3 shows each stage. In Figure 3 (a), 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. In Figure 3 (b), probe nodes (labeled D2 and S2) are disconnected from their original piconets and are connected to probed piconets. After these connections, the probe nodes and the nodes in the probed piconets exchange metadata. Synchronized transfer time will be assigned at this time. In Figure 3 (c), the probed nodes return to their original

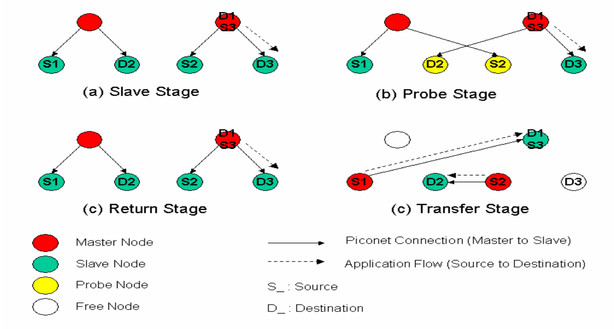

Figure 3. Overlaid Bluetooth Piconets Stages

piconets and convey the metadata to their piconet nodes. In Figure 3 (d), source and destination nodes are disconnected from their original piconets. Source nodes make connection to destination nodes and start inter-piconet transfers such as S1 → D1 and S2 → D2.

III. THROUGHPUT AND POWER ESTIMATION

Throughput and Power are estimated to make comparison between OBP and Scatternet.

A. Overlaid Bluetooth Piconet (OBP)

Slave stage, Probe stage, Return stage, and Transfer stage durations are denoted as (1)-(4) and OBP Period duration is the sum of all stages' durations and denoted as (5).

$$T_{slave} = t_{page} + t_s \tag{1}$$

$$T_{probe} = t_{inquiry} + t_{page} + t_m \tag{2}$$

$$T_{return} = t_{page} + t_m \tag{3}$$

$$T_{transfer} = t_{page} + t_t \tag{4}$$

$$T_{period} = T_{slave} + T_{probe} + T_{return} + T_{transfer} \tag{5}$$

t_{page} and $t_{inquiry}$ are page time and inquiry time, respectively. t_m is metadata transfer time in *Probe stage* and *Return stage*. t_s is slave time in *Slave stage* and used only for intra-piconet transfer. t_t is transfer time in *Transfer stage* and used for inter-piconet transfer. But, intra-piconet transfer is still possible during *Transfer stage* because not all the piconet links are disconnected all the time. If source and destination nodes are not used for inter-piconet transfer, they can be used for intra-piconet transfer.

Intra-piconet throughput in OBP is calculated as follows.

$$\theta^{sd}_{\text{int}ra} = C \cdot q_{sd} \cdot f_{sd} \cdot p_i \cdot \left(\frac{t_t}{T_{period}} + (1 - p_e) \cdot \frac{T_{transfer}}{T_{period}}\right) \tag{6}$$

Intra-piconet transfer is possible during t_t when source and destination are in the same piconet. It is also possible during $T_{transfer}$ when source and destination remain in the same original piconet because they are not used for inter-piconet transfer. C is the maximum capacity of a Bluetooth radio link, specified in Table I. f_{sd} is usage percentage of capacity calculated by 1 over the number of intra-piconet flows in one piconet. q_{sd} is the packet success rate (PSR) of the link (s, d) that can be obtained from the packet error rate (PER), as (7), while PER, denoted by r, can be calculated as a function of the bit error rate (BER), using the formulae (8) and (9), for DH and DM packet types, respectively [4].

$$q = 1 - r \tag{7}$$

$$r = 1 - (1 - b)^s \tag{8}$$

$$r = 1 - ((1 - b)^{15} + 15b(1 - b)^{14})^{s/15} \tag{9}$$

p_i is the probability of intra-piconet (internal) flow existence and p_e is the probability of inter-piconet (external) flow existence.

Assume that N is the set of nodes in the conference room and F is the set of all flows in all nodes. In that case, $|F|$ sources and $|F|$ destinations exist. So, the possibility of having a source or a destination at a certain node is $|F|/|N|$. And then, p_i and p_e are calculated as follows.

$$p_i = \frac{|F|}{|N|} \cdot \frac{n_{piconet} - 1}{|N| - 1} \tag{10}$$

$$p_e = \frac{|F|}{|N|} \cdot \frac{n_{probed\_piconet}}{|N| - 1} \cdot p_{probe} \tag{11}$$

$n_{piconet}$ and $n_{probed\_piconet}$ are the number of nodes in original piconet and in probed piconet, respectively. p_{probe} is probability that at least one piconet is probed. It depends on the communication range and nodes' moving range. If all nodes are in the communication range, all piconets are in the same range. So, at least one piconet detects probe node and connects to it. In this case, p_{probe} is 1. If all nodes are not in the communication range, p_{probe} is *communication area / moving area*. Near the boundary, communication area will be decreased because it is not a full circle. So, p_{probe} can be calculated as follows.

$p_{probe} = 1$ (all nodes are in the communication range) (12)

$p_{probe} \cong 1 - \left(1 - \frac{10^2 \pi}{Xr \cdot Yr}\right)^{(|N|/\overline{n_{piconet}}) - 1}$ (all nodes are not in the communication range) (13)

$|N|/\overline{n_{piconet}}$ is average number of piconets, and $\left(1 - \frac{10^2 \pi}{Xr \cdot Yr}\right)^{(|N|/\overline{n_{piconet}}) - 1}$ is the probability that all other piconets are not in the communication range of 10m in the moving area of Xr by Yr.

Inter-piconet throughput is calculated as follows.

$$\theta^{sd}_{\text{int}er} = C \cdot q_{sd} \cdot p_e \cdot \frac{t_t}{T_{period}} \tag{14}$$

Total throughput is the sum of intra-piconet transfer and inter-piconet transfer and it is calculated like the followings.

$$\theta = \sum_{(s,d)\in F} \theta^{sd}_{intra} + \theta^{sd}_{inter} \tag{15}$$

Power consumption for transfer in OBP is calculated as follows.

$$P = \sum_{(s,d)\in F} (P_t + P_r) \cdot h_{sd} \cdot f_{sd} + P_{con} \tag{16}$$

h_{sd} is the hop distance between source and destination. For the intra-piconet transfer, the hop distance is 1 (master and slave) or 2 (slave and slave), and for the inter-piconet transfer, it is 1. In [5], P_t and P_r are assumed as transmitting and receiving power consumption at the full capacity of a radio link. P_{con} is the power consumed for connection and disconnection in various stages.

B. *Bluetooth Scatternet*

In [5], throughput is calculated as follows.

$$\theta = \sum_{(s,d)\in F} C \cdot \min_{(i,j)\in(s,d)} (f^{sd}_{ij} q_{ij}) \tag{17}$$

$\min_{(i,j)\in(s,d)} (f^{sd}_{ij} q_{ij})$ denotes the smallest usable bandwidth portion on the links of a connection (s,d) (i.e the bottleneck), while q_{ij} is the packet success rate (PSR) of the link (i, j).

In [5], power consumption is calculated as follows.

$$P = \sum_{(s,d)\in F} (P_t + P_r) \cdot h_{sd} \cdot \min_{(i,j)\in(s,d)} (f^{sd}_{ij}) \tag{18}$$

h_{sd} is the hop distance between source and destination. Notice that the factor $\min_{(i,j)\in(s,d)} (f^{sd}_{ij})$ in (18) adapts the power consumption to the bandwidth of the bottleneck link along the path.

IV. SIMULATION

In this section, we present the simulation environment that we used for evaluating our approach.

TABLE II. SIMULATION PARAMETERS

Moving Area (Xr, Yr)	$15.1\times15.1\,m^2$, $21.28\times21.28\,m^2$
Number of Nodes (N)	50
Number of Flows (F)	100
Moving speed of nodes (S)	0.0, 0.3, 0.6, 0.9, 1.2 m/s
Packet type (P)	DH5, 2-DH5, 3-DH5
Slave time and Transfer time (t_s, t_t)	5, 7, 10 sec

A. *UCBT Simulator*

For evaluation purposes, we implemented OBP algorithms in the UCBT ns-2 based Bluetooth simulator [5], because it is the only publicly available open source Bluetooth simulator that supports mesh-shaped scatternets.

UCBT implements the majority of the protocols in the Bluetooth. The simulator has recently added support for mesh-shaped scatternets, but it assumes that all nodes are in the communication range. Therefore, we also added to UCBT a simple scatternet formation protocol (described in section C), besides our OBP algorithms.

B. *Mobility*

For simulating mobility, we use the revised random waypoint model and Nomadic community mobility model in [10]. We assume a piconet master is moving according to the random waypoint model and slaves are staying in the short range (< 3m) of their master. All piconet members are moving to randomly chosen direction and speed. Maximum speed is predefined to limit node's speed. Direction is changed periodically with an offset in the range [-10, 10] degrees with respect to the original direction. When a node reaches the boundary of the simulation area, it is mirrored back into the simulation area.

C. *Scatternet Formation*

We implemented a scatternet algorithm based on [6, 7, 8]. On the first phase, nodes execute inquiry or inquiry scan with a probability of 1/4 and 3/4, respectively. When an inquiry node discovers an inquiry scan node, it will page the inquiry scan node. This way, the inquiry node becomes a master of the other node in the newly formed piconet. After this first phase, piconets are formed. On the second phase, master nodes

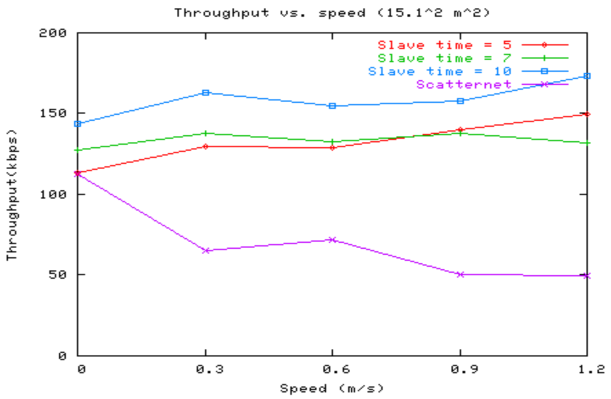

Figure 4. Throughput vs. Speed

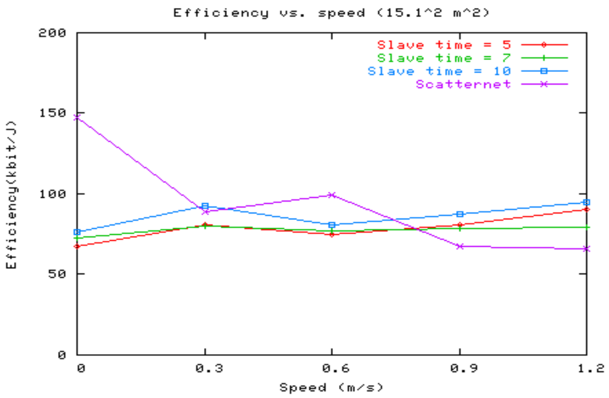

Figure 5. Efficiency vs. Speed

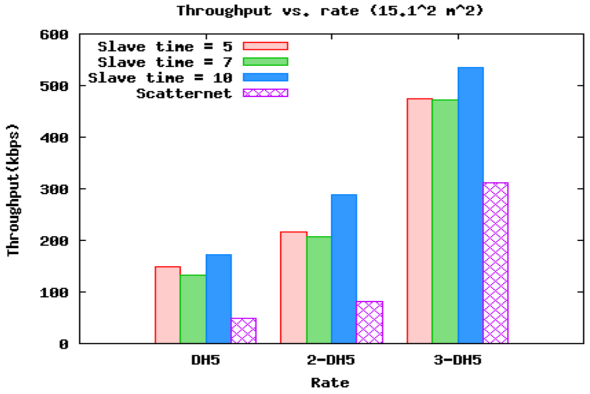

Figure 6. Throughput vs. Rate

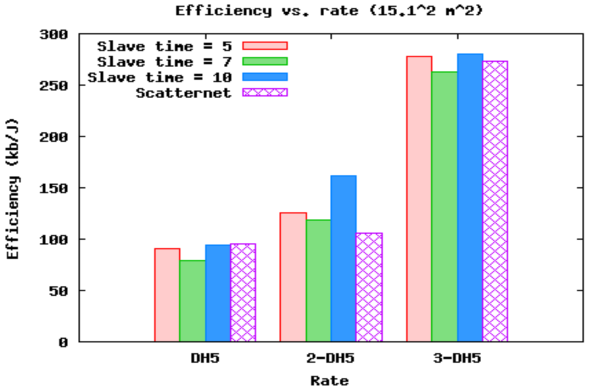

Figure 7. Efficiency vs. Rate

execute inquiry and slave nodes execute inquiry scan. When master detects nodes that have longer than MAX_HOP distance (we define it as 4), master connects them and a scatternet is formed.

D. Parameters

Parameters are described in Table II.

V. RESULTS

We evaluate throughput and efficiency (throughput / power consumption) versus speed, data rate, and time. We also check number of distinct probed piconets per second versus slave and transfer times. Transfer time is set same as slave time value and we will call it slave time.

A. Throughput vs. Speed

Figure 4 shows throughput vs. speed results. We use maximum moving speed varying from 0 to 1.2 m/s to evaluate the throughput versus speed. DH5 packets and 15.1×15.1 m^2 area are used for this test. As the speed increases the throughput of scatternet decreases because of frequent link disconnections. However, the throughput of OBP cases stays the same or increases as the speed increases. High mobility makes higher chance of meeting other piconets, which produces more inter-piconet transfers in OBP and thus increases throughput.

B. Efficiency vs. Speed

Figure 5 shows efficiency vs. speed results. With the same testing environment as in section A, the power consumption in OBP cases are higher than that of scatternet because of higher throughput, frequent connections and disconnections, and metadata transfers. Even though the power consumption is higher in OBP, the throughput is much higher than that of scatternet, which results in better efficiency in high mobility cases for OBP.

C. Throughput vs. Rate

Figure 6 shows throughput vs. rate results. For this test, DH5, 2-DH5, and 3-DH5 packets are used. The speed is set to 1.2 m/s speed and the area is set to 15.1×15.1 m^2 for this test. When higher capacity packets are used, throughput increases as we expected in all cases. All OBP cases' throughputs are better than those of scatternet because of multiple one-to-one transfers in OBP.

D. Efficiency vs. Rate

Figure 7 shows efficiency vs. rate results. With the same testing environment as in section C, the efficiencies of OBP and scatternet do not vary a lot for a particular rate. As the rate increases, the efficiencies increase as well in the same pattern of throughput in section C because the power consumptions do not vary very much among different rates.

E. Probe rate vs. Speed

Figures 8 and 9 show the number of distinct probed piconets per second with varying speeds in the areas of

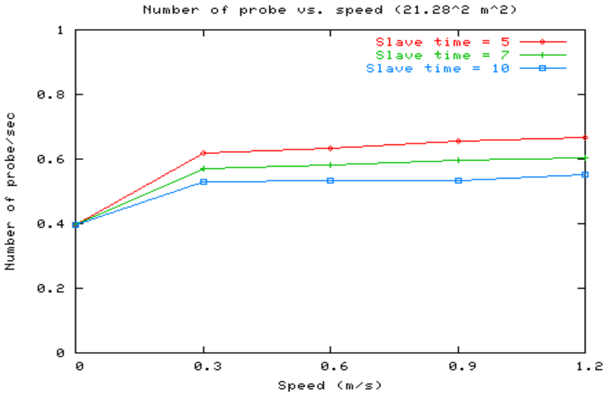

Figure 8. Probe Rate versus Slave time (21.28×21.28 m^2)

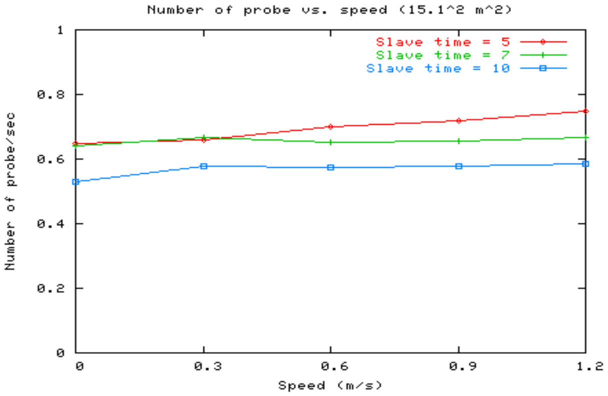

Figure 9. Probe Rate versus Slave time (15.1×15.1 m^2)

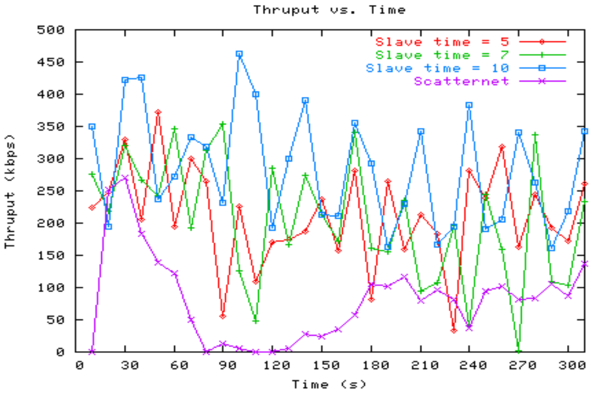

Figure 10. Throughput versus Time

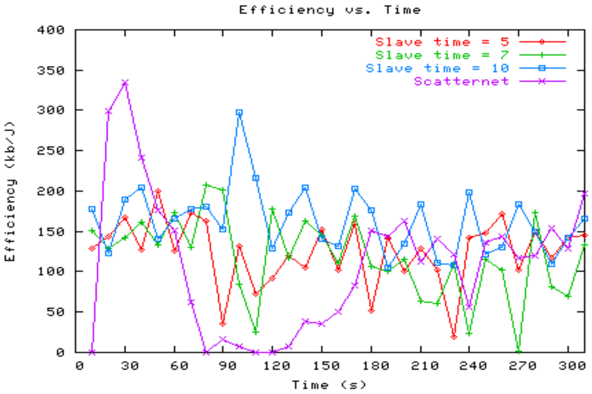

Figure 11. Efficiency versus Time

$21.28 \times 21.28\, m^2$ and $15.1 \times 15.1\, m^2$ respectively. When speed increases, the percentage of probed piconets increases in both areas. And this increase reflects the increase in throughput shown in section A. Also, in the larger area, the percentage increase between the speeds of 0 and 0.3 m/s is significant compared to other speed differences as expected since nodes start moving increases the chance of meeting other piconets. This is not the case for the smaller range as more piconets are already in the communication range even if speed is 0 m/s. Among different slave times, shorter one has higher probe rate than longer one as we expected because longer slave time increases total OBP period and thus decreases the number of probe.

F. *Throughput vs. Time*

Figure 10 shows every 10 seconds' average throughput. We use 1.2 m/s speed, 2-DH5 packets, and $15.1 \times 15.1\, m^2$ range for this test.

In OBP, throughput varies a lot during the test time, because inter-piconet transfers (which is the main part of the throughput) are only possible during Transfer Stage. During this stage, the throughput is high and in other stages the throughput is low, and this is reflected in the oscillation of the throughputs in the figure. Shorter slave time one has shorter Transfer Stage and thus has shorter oscillation period where as the longer one has longer period. In the scatternet, node's movement disconnects some links and therefore decreases throughput at certain times and reconnection regains the throughput.

G. *Efficiency vs. Time*

Figure 11 shows every 10 seconds' average efficiency. Same parameters in section F are used. In *Probe stage* and *Return stage*, power for inter-piconet transfers disappears, instead, power for connections & disconnections is consumed. So, power consumption does not decrease as throughput decreases. In scatternet, power consumption is almost constant throughout the tests. Thus, the efficiency follows the throughput pattern in section F.

VI. CONCLUSION

In this paper, we presented a different approach to interconnect Bluetooth piconets in mobile environments. Overlaid Bluetooth Piconets (OBP) shows resiliency to mobility compared to traditional scatternet and produces significantly higher throughput. Scatternet requires scatternet formation and reformation as nodes are moving. OBP creates a virtual scatternet that does not require persistent connections and routing protocol. Thus, it is very well suited for mobile environments. The efficiency of OBP is very comparable to that of scatternet while keeping higher throughput.

Also with higher mobility, in OBP, the chance of meeting other piconets increases and thus various application flows can be supported which increases the throughput.

OBP is applicable to all currently available Bluetooth devices that may not support scatternet. And thus OBP is more practical for networking Bluetooth devices than scatternet.

In the future, we will add store-and-forward method to increase transfer opportunity, and metadata flooding to keep the virtual scatternet up-to-date. Moreover, we will implement OBP on the real testbed to show its applicability and practicality.

REFERENCES

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

[2] Kevin Falls, "A delay-tolerant network architecture for challenged internets". , Proceedings of ACM SIGCOMM, 2003.

[3] Stefano Basagni and Chiara Petrioli, "A scatternet Formation protocol for ad hoc networks of Bluetooth devices", IEEE Vehicular Technology Conference (VTC), 2002

[4] Ling-Jyh Chen, Rohit Kapoor, Medy Young Sanadidi, and Mario Gerla, "Enhancing Bluetooth TCP throughput via link layer packet adaptation", IEEE International Conference on Communications (ICC), 2004.

[5] Casaba Kiss Kallo, Sewook Jung, Ling-Jyh Chen, Mauro Brunato, and Mario Gerla, "Throughput, energy and path length tradeoffs in Bluetooth scatternets", IEEE International Conference on Communications (ICC), 2005.

[6] Pan Hui, Augustin Chaintreau, James Scott, Richard Gass, Jon Crowcroft, and Christophe Diot, "Pocket switched networks and human mobility in conference environment", SIGCOMM Workshops, 2005

[7] Ching Law, Amar K. Mehta, and Kai-Yeng Siu, "A new Bluetooth scatternet formation protocol", Mobile Networks and Applications, 2003

[8] Sewook Jung, Csaba Kiss Kallo, Mario Gerla, and Mauro Brunato , "Decentralized optimization of dynamic Bluetooth scatternets", Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: networking and services (MobiQuitous), 2005

[9] UCBT simulator. https://www.ececs.uc.edu/cdmc/ucbt/

[10] Tracy Camp, Jeff Boleng, and Vanessa Davies, "A survey of mobility models for ad hoc network research", IEEE Mobicom, 2002