CBS: Constraint-based Approach for Scheduling in Bluetooth Networks

Gang Jiang, Jingli Zhou, Shengsheng Yu

College of Computer Science and Technology, Huazhong University of Sci.& Tech.

Wuhan, Hubei 430074, P.R.China

springfunny@hotmail.com

ABSTRACT

In Bluetooth networks, devices are organized into small piconets and large scatternets, and each node acts as the role of master, slave or gateway. Due to dynamic topology changes, different bandwidth available and unpredictable interference of media in Bluetooth networks, the congestion of data flow will inevitably emerges on the link, and the gateway has to switch between piconets on a time division basis, so its presence in the different piconet has to be controlled by scheduling mechanism such as inter- and intra-piconet scheduling. However, the time division in gateways will limit the network capacity and introduce bottleneck points in the network, and the switch between piconets will prevent the packet from transmitting smoothly and efficiently. Most of the published work on Bluetooth scheduling has focused on the polling scheme between master and slaves. In this paper, we put our approach on the inner constraints of Bluetooth networks and present a constraint-based scheduler (CBS), to adaptively cater to the changing role of each node throughout Bluetooth ad hoc networks, thereby it will save time and definitely enhance fairness and efficiency on packet scheduling in Bluetooth environment.

Keywords: Bluetooth, Ad hoc networks, Piconet, Scatternet, Constraint-based, Scheduling

1. INTRODUCTION

Recently, Bluetooth has emerged as a promising technique for short-ranged wireless telecommunication, with low-cost, low power, small-sized advantages to support short-range ad-hoc networks. It has originally been developed as a cable replacement between electronic device, and now gradually as a full solution for short range wireless Personal Area Networks(PAN).

Bluetooth supports ad hoc networking and provide both Point-to-Point and Point-to-Multipoint communications. In Bluetooth networks, devices are organized into small Piconets, which in turn inter-connected to form larger networks called Scatternet. The Scatternet is formed by multiple Piconets. In such ad hoc environment, no existing infrastructure is needed and users can select devices to communicate optionally and flexibly. When two Bluetooth communicate, one of them assumes the role of master and the other assumes slave, a single Piconet can accommodate up to seven active slaves simultaneously. In most cases, one kind of slave can participate in more than one Piconet, and will multiplexes between multiple Piconets. This node with multiple roles (PMP) is named gateway, or bridge, which acts as the role of bridge to adjacent piconets [1] (as shown in Fig.1). In most scenarios, different bandwidth available on each link and dynamic topology formation may cause congestion on the link, which is the bottleneck often emerges on the gateway [2].

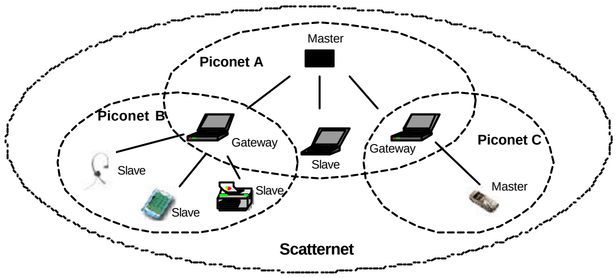


Fig. 1 Piconet and Scatternet in Bluetooth

The gateway has to switch between piconets on a time division basis. However, the time division in gateways will limit the network capacity and may introduce bottleneck points in the network, and the switch between piconets will prevent the packet from transmitting smoothly and efficiently, so the presence of gateway in different piconets has to be controlled by scheduling mechanism. In recent years, many scientists have developed a few scheduling algorithms for Bluetooth, such as Round Robin(RR), Exhaustive Round Robin(ERR), Priority Round Robin(PRR) and so on [3]-[4]. They emphasize the different policies for polling from master to the slave, and efficiently coordinate the units to guarantee fairness, and maximize the network capacity to optimize the channel utilization.

In this paper, we present a constraint-based approach rather than polling. We propose a constraint-based scheduler (CBS) to adaptively cater to the changing role of each node throughout Bluetooth ad hoc networks, thereby it will save time and definitely enhance fairness and efficiency on packet scheduling in Bluetooth environment.

This paper is organized as follows. We first give a brief introduction of some key concepts and related issues in Bluetooth networks, and discuss the QoS mechanism as well as the scheduling schemes in this environment. And then, we present a constraint-based model for packet scheduling in Bluetooth ad hoc networks. Based on this model, we propose our constraint-based scheduler and evaluate the performance. Finally we conclude the paper and present the future work.

2. OVERVIEW OF DATA FLOW AND QOS MECHANISM IN BLUETOOTH

In this section, we give a brief insight and overview of some issues in Bluetooth environment, including related concepts about data flow in Bluetooth, packet scheduling, and Quality of Service scheme to support the scheduling.

There are two link types to establish the connection between master and slave, SCO (Synchronous Connection Oriented) and Asynchronous Connectionless Link (ACL), these two kinds of link play a very important role in controlling data and enable QoS requirements in Bluetooth,

The SCO link is typically suitable for carrying real-time traffic such as video and audio, which allows the periodic transmission of SCO data. ACL link is suitable for supporting data applications. A polling scheme is used to control the transmissions between master and slave in ACL link. By way of polling, the master transmits packets in the even numbered slots and the slave responds in the odd numbered slots in downstream and upstream way separately [5], and a duplex link through Time Division Duplex(TDD) is established.

But sometimes the Piconets are not synchronized; the slot timing is not aligned because of the hop on a different frequency. When a switch to another Piconet happens, at least one TDD frame is wasted and different Piconets are not slot synchronized. To solve the problem, much attention is drawn on how to poll the slave efficiently to gain both fairness and utilization rate. For example, Manish Kalia proposes a SAR scheme to schedule the TDD efficiently, and N.Golmie presents a BIAS algorithm to realize this goal and extends his approach to QoS supports[6].

In this paper, we do not focus on the scheduling policy for polling slaves, we put our emphasis on the internal constraints from dynamic topology and unpredictable interference, since there is inherent relationship or constraint between piconets even if frequent switch happens and topology changes in this ad hoc environment, we are more interested in building the model to express the constraints, which is aimed at providing both flexibility and efficiency to meet QoS requirement.

First, we should issue the QoS requirements and enhanced mechanism in Bluetooth environment, then we can tailor our approach to a new kind of scheduler, which schedules the data flow in Bluetooth piconet/scatternet and enable efficient transmission of audio, video and data in mobile Bluetooth environment.

The factors to affect Quality of Service(QoS)

Due to frequent topology changes and unpredictable interference of media in Bluetooth networks, provisioning and guaranteeing QoS in such environment is a very challenging problem. For this, flow control, packet scheduling and other QoS based algorithms have been addressed in literature. Although the Bluetooth specification provides some Quality of Service supports, some deficiencies have been identified. The common factors which affect QoS are showed as follows:

1) Delay. Caused by L2CAP packets transmission, Re-transmissions in error conditions, SCO traffic(Because the timeslots reserved for SCO traffic can not be used for ACL traffic), as well as Packetisation(The delay arises when HCI Data Packet are reassembled at the Baseband receiver).
2) Bandwidth wasted by polling algorithm. This algorithm has the inherent problem that the master is not aware of the instantaneous traffic demand of the slave, so the master may poll a slave which has nothing to send.
3) Fixed symmetric bandwidth. The SCO(Synchronous Connection Oriented) link has deficiencies to support QoS because it can only provide a fixed symmetric bandwidth, but the stream voice and video require a variable bandwidth.

QoS requirements and enhanced mechanism

(1) Purpose: low loss, low latency, low jitter, quantitative QoS guarantee with aspect to an assured bandwidth.

(2) Requirements: simple and low cost, flexibility to support different QoS requirements on the same Bluetooth device, ability to roam among different wireless interfaces, capacity to interwork with other QoS mechanisms such as DiffServ, IntServ and so on.

(3) QoS Framework in Bluetooth protocol

The following figure shows the Framework which provides Quality of Service in Bluetooth:

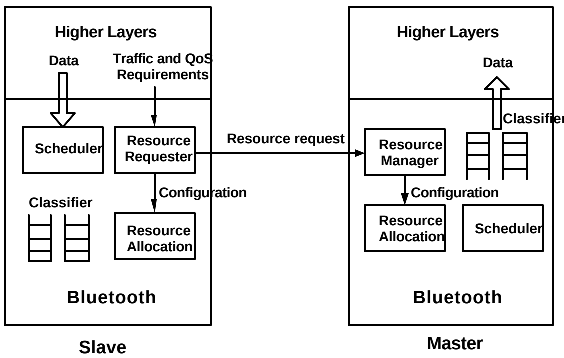


Fig.2 QoS Framework of Bluetooth

As we can see, different traffic flows can share the air-interface resources and the resources on the local/remote Bluetooth devices. The QoS flow originates at the slave. The Resource Requester and Resource Manager enable to reserve air-interface bandwidth for a QoS flow in the Piconet. The Resource reservation requires the signaling of Traffic and QoS requirements over the air-interface in case of a Slave-to-Master flow. The polling algorithm, Inter-Piconet scheduling algorithm, and Air-interface scheduling algorithm should be identified to determine the amount of resources assigned to a traffic flow. We can also see that the scheduler plays a very important role in supporting QoS requirements. In the following, we will analyze the constraints in piconet units and propose a model to schedule the video and audio streaming.

3. CONSTRAINT-BASED SCHEDULER (CBS)

From a Quality of Service point of view, it is preferred to introduce a new kind of scheduler to achieve both efficiency and flexibility. It is highly needed to minimize the effect of wireless hops in the end-to-end path, and settle the problem of packet transmission from a Master as well as Slave cannot span across voice slots [7]-[8].

As mentioned in last session, we will put our emphasis on the internal constraints and more interested in building the model to express the constraints within and between piconets.

The model of constraint-based scheduler

Considering the fact that in mobile ad hoc network, the only restriction for role switching is that one node, regardless of master or slave, can be active only in one piconet at one time, we may admit that every node has its constraint action according to the time division. The freedom degree, active or inactive status of each node, priority class and other factors can be added to a Constraint Table, which enables efficient scatternet operation by offering much flexibility for a node to adapt its activity. In the light of this idea, our scheduler (Fig.3) is concerned with the assignment of the roles of master, slave, and bridge dynamically so that the scheduler can deal with variable situation, and efficiently coordinate the presence of gateway in different piconets.

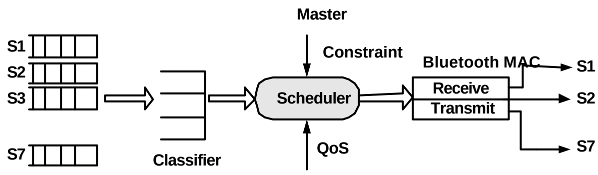


Fig. 3 Constraint-based scheduler in Bluetooth

Piconets can be connected into a scatternet by sharing slaves. The configuration of a scatternet has great effects on the performance of the network. For instance, when a scatternet contains more piconets, the rate of packet collisions increases. So it is highly needed to discuss the inner mechanism in piconet and scatternet, especially the relationship among distributed nodes and actions of each node.

At first glance, the interconnection topology in a single channel system is a complete "graph" data structure [9]. The network of N nodes is denoted by sets $N = \{n_i : i = 1,...N\}$. However, combined with the inner mechanism, we may find that the topology is not a complete graph, there are several constraints $C = \{c_i : i = 1,...M\}$ contained in Bluetooth(Fig.4), which will affect packet scheduling and enable QoS requirements. The constraints exist in the course of scatternet topology formation, packet routing and channel scheduling. The rules of constraints are listed as follows:

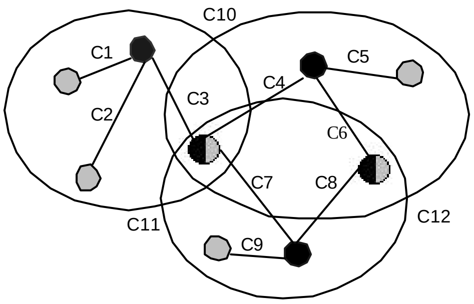


Fig. 4 Constraints in topology changes

End-to-end communication constraint: the communication only takes place between a master and a slave and never between two masters or slaves directly, but each node can transmit data to any other nodes in the network.

Channel Assignment constraint: since Bluetooth is a frequency hopping system which defines multiple channels for communication, the scheduler should aware which subgroup of nodes share a common channel and which nodes forward video/audio stream from one channel to another [10].

Polling constraint: a strict alternation of slots between the master and slaves, the master transmits packets in the even numbered slots and the slave responds in the odd numbered slots.

Dynamic master/slave function constrains: take place in topology changes. The dynamic unit should schedule their activities. An effective way is using Rendezvous Window Distribution[11].

Unit role constraint: in one piconet, only one master and several slaves (n<=7) involve in.

Gateway function constraint: the gateways should adjust and coordinate their presence intervals to each piconet they join; it should realize a tradeoff between bandwidth usage and delay while switching.

Gateway presence constraint: the gateway can be a master only in one piconet.

Master presence constraint: aware when it can poll the slave and take actions to deal the response.

Slave presence constraint: should answer the poll request from the master so it cannot participate in several piconets once an active link exists.

Parameters in constraint-based scheduler

To describe the constraints, we introduce some parameters (listed in Table 1) to express the scheduler:

TABLE I
PARAMETERS TO BE USED IN CONSTRAINT-BASED SCHEDULER

Parameters	Definition
Number of Piconets	The number of p iconet
Node ID	Endow master, slave, gateway different ID
Freedom degree	Flag to express constraint level of each node
Number of slaves	The number of slaves in a Piconet <7
Priority class	Master, slave, gateway have different priority class
Active Fla g	Active or inactive status of each node

Note: Degree of Freedom(DOF): the slave is controlled by master in one or more piconets, we may define the number of piconets in which a node belongs. If the master is only in one piconet, the degree is 1, while in N piconets, the degree is N. But we will define N<=2, for the purpose of relieving a node of being overloaded. It is a flag to describe the constraint level of nodes. If N=2, it should be the gateway, which acts in the list of {M, S}, so its presence tak es the role of M/S or S/S.

Theoretical description of the scheduler

Our approach takes into account collecting information (status) about all nodes participating in the process at a single point before and after actual connection happens. Let us assume initial state of network is S_i before the topology changes, it moves to another status S_j, the corresponding relationship is showed as: $S_j = F(S_i)$, this function reflects that the status change is formulated by some inherent factors, which can be driven by different constraints. The different constraints can be expressed by the algebraic equation $F = \{ f_j(x_1,....x_n) \mid j = 1,...m\}$ while the status variables are $S = \{s_i \mid i = 1,...n\}$, the coupling properties of the variables are determined by the constraint propagation, the coupling properties can be shown as a matrix $T_{m\times n}$ with the rule of

$$T_{i.j} = \begin{cases} 1 & if\ s_j\ isshownin\ f_i \\ 0 & if\ s_j\ isnotin\ f_i \end{cases}$$

For this, each Master has a Constraint Table shown in Table 1, which states the changing status and control the data flow within Piconet. Using the table and coupling properties, as well as adaptive constraint propagation, the constraint-based scheduler is able to adaptively cater to the changing role of each node, whenever it occurs in the course of scatternet topology formation, packet routing or channel scheduling. So it may save time and definitely enhance fairness and efficiency.

The module of our scheduler consists of several components, namely, the pseudo code of the procedures is:

Step 1: Record_constraint(),
Step 2: Route_packet(),
Step 3: Channel_scheduling(),

During the course of tree topology formation, Record_constraint() realizes the creation for constraint table. When a node is to join the network, it sends out frequent constraint announcements, and the root of tree can gather the information from all the child nodes and will record the possible constraints in this process. Next, we will show an adaptive propagation scheme to add a new constraint to the list.

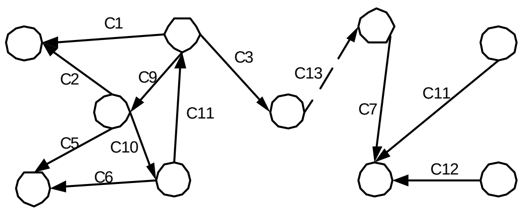


Fig.5 Add a new constraint C13

The goal of our scheme is to adjust the method for constraint propagation/update according to the dynamic topology, so as to create adaptive constraint tables to gain flexibility and efficiency.

If the device node e matches the condition in Constraint table c: if degree = 0 and the matching constraint between c and e can be propagated, we can add c in the matching list of e, also let us assume v as the node list, we may create a new relationship line around $v_s(e)$ corresponding to constraint table c. In addition, let us traverse the $v_s(e)$ related to c to find the reasonable way to propagate between v_s and v_e. (v_e is the node which has no constraint), and push it to the table stack, the head node is the stack element at bottom, and the tail node is the element at top. So a new constraint is added in the constraint table. When a constraint is to be used, the top element c_i is pulled immediately, and matched with $v_s(e)$ to create a new relationship.

An example is shown in Fig.5, let c_1 ~ c_{12} to be the original constraints, the new constraint c_{13} is allowed to participate in when the status changes.

Route_packet() provides a way to guarantee that any packets can be transmitted to the specified destination. According to the constraint table and the address space allocated to each child, the packet can be sorted in a specified route.

Channel_scheduling(). There are two phases in this procedure, one is determining the role of each node according to the dynamic constraints table, the other is estimating the temporary channel to optimize allocation on each time slot and avoid the interference in transmission. Since the constraint table has been formed, this operation will simplify the channel scheduling process. The Availability of channel can be denoted as the system throughput as follows.

Each transmission is modeled as a flow f_i, $1<i<N$, which is characterized by e_i (maximum acceptable packet loss rate), d_i (each packet should be scheduled within the time period), and v_i (the time interval between two successive packets), let M_i to be the total number of packets that flow f_i is supposed to transmit and M_i^a to be the number of packets that actually get transmitted. Then the throughput for flow f_i, say t_i is defined to be the ratio of M_i^a to M_i, the overall throughput is $\frac{\sum_{i=1}^{N} M_i^a}{\sum_{i=1}^{N} M_i}$, we define the loss rate for flow f_i as $\boldsymbol{e}_i = 1 - e_i - t_i$, our aim is to minimize $\boldsymbol{e}_i$ and maximize $\sum_{i=1}^{N} M_i^a$, it is indicated that our constraint-based scheduler improves the flow throughput according to the experimental results.

As a whole, Bluetooth is a Master driven standard; all packets are exchanged between a master and its slave within piconets, the constraint-based scheduler can make use of the measurement result to improve packet transmission in a receiving channel in a dynamic environment.

4. SIMULATION RESULTS

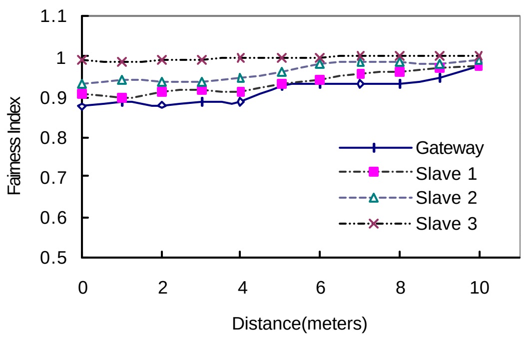
In this section, we present simulation results to evaluate the performance of the constraint-based scheduler (CBS), based on Discrete Events Simulations [12]. We have modeled a set of objects to describe the masters, slaves, and gateways, and take 2 piconets and consider 3 slaves and 1 gateway as our samples. We assume every node participating in a high bandwidth of 640Kbps intra -piconet connections.

Fairness guarantee

We may define the performance evaluation criteria [13] for the scheduler, which is the fairness index for each device node P_i. We assume N_r as the number of packet received, N_d as the number of packet dropped, P_i as the expected number of packet received, which means $N_e = N_r + N_d$, the fairness index of each node is:

$$P_i = N_r / N_e \quad \text{or} \quad P_i = (N_e - N_d)/ N_e .$$

From Fig.6, we may find out CBS improves the fairness of each node in networks, as for Exhaustive Round Robin, the average fairness index for slave1, slave2, slave3, gateway are 0.92, 0.94, 0.97, 0.90 respectively, while for CBS are all close to 1.0, therefore CBS gains satisfactory results for fairness guarantee.



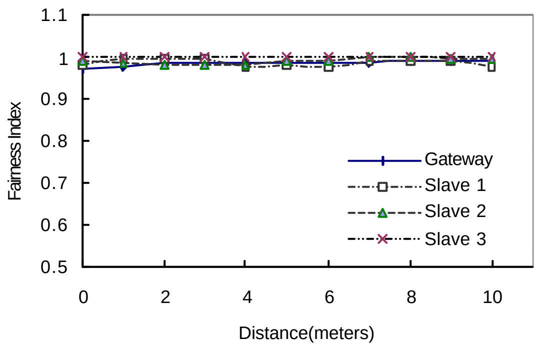


Fig 6 (a)/(b) Performance of Fairness a) ERR b) CBS

Throughput

In Fig.7, we see the performance improvement of CBS as compared to other scheduler such as Round Robin and Exhaustive Round Robin[14]. It shows that the proposed CBS obviously increases the network throughput.

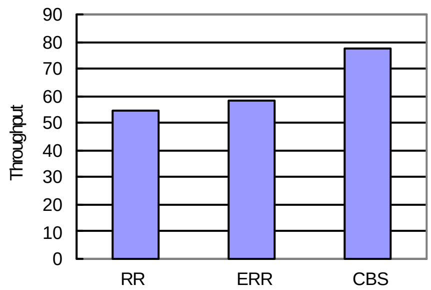


Fig. 7 Throughput of different scheduler (in % of slots)

Packet Error rate

We may increase the number of piconets and nodes to test the packet error rate. Since in Bluetooth, voice has a higher priority than data, we test the packet error rate of the voice traffic by way of CBS switch. In Fig. 8, it is showed that when the switch is on, the packet loss is less than it turns off. Especially the trend for CBS switch-on curve is smoother than switch-off, which indicates that it takes less affected by dynamic topology changes. In this case, if CBS is turned off, when the number of nodes increases to 6, there is a steep change(23%, from 0.74 to 0.91) of error rate due to the changing topology, however, CBS can effectively improve it.

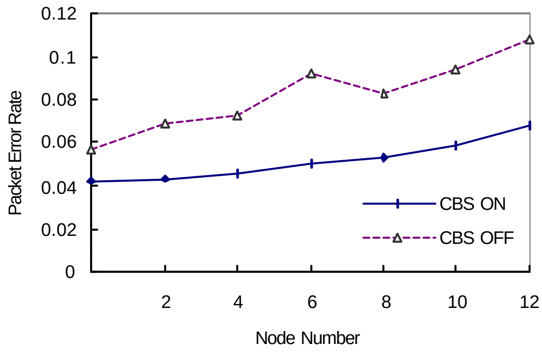


Fig 8 Packet Error Rate of the Video Traffic

5. CONCLUSION

In this paper, we put forward a constraint-based scheduling (CBS) method to adaptively cater to the dynamic topology thereby it enhances flexibility and efficiency in Bluetooth networks. Establishing the constraint-based scheduler is a promising approach because it takes into consideration of the inner constraints existing within and between piconets when Bluetooth network is formed, and it provides an adaptive scheme to avoid the negative effects such as mobile interference in Ad hoc networks. As a consequence, it will help avoid wasting slots on failed master/gateway attempts to communication, and introduce a way to maintain fairness and efficiency to support QoS requirements in Bluetooth networks.

Our simulation results indicate that CBS can significantly lower the probability of packet loss and increase the throughput; since it has the advantages to make use of the constraint-based results to improve packet transmission in high-level interference environment.

Our future work is focused on extending CBS on statistical base. We should find more reasonable mathematical methodology to evaluate the constraints, optimize the result from several results proposed.

REFERENCES

[1] T.Salonidis, P.Bhagwat, L.Tassiulas, R.La Maire "Distributed topology construction of Bluetooth personal area networks." Infocom 2001

[2] Bluetooth Special Interest Group. http://www.bluetooth.com

[3] N.Johansson, U.Korner and P.Johansson, "Performance Evaluation of Scheduling algorithms for Bluetooth", In Broadband Communications: Convergence of Network Technologies, Edited by Danny H.K.Tsang and Paul J.Kuhn, Kluwer Academic Publishers, pp 139-150, 2000

[4] N.Johansson, L.Tassiulas. A Distributed Scheduling Algorithm for a Bluetooth Scatternet, The Seventeenth International Teletraffic Comgress, ITC' 17, Sept.24-28, 2001, Salvador da Bahia, Brazil.

[5] G.Zussman and A.Segall, "Capacity Assignment in Bluetooth Scatternets – Analysis and Algorithms", Proc. IFIPTC6 Networking 2002, LNCS 2345, Springer-Verlag, May 2002.

[6] Manish Kalia, M.; Bansal, D.; Shorey, R. "MAC scheduling and SAR policies for Bluetooth: a master driven TDD pico-cellular wireless systems." 1999 IEEE International Workshop on Mobile Multimedia Communications(MoMuC' 99)

[7] Z. Jiang and L. Kleinrock, "A Packet Selection Algorithm for Adaptive Transmission of Smoothed Video over a Wireless Channel," Journal of Parallel and Distributed Computing, vol. 60, pp. 494 – 509, 2000.

[8] K.Lee and M.El Zarki, "Scheduling Real-Time Traffic in IP-Based Cellular Networks", Proc. PIMRC' 2000, pp.1202-1206, sept.2000.

[9] Basagni, I. Chlamtac, "Bluetrees – Scatternet Formation and Routing in Bluetooth-based Ad Hoc Networks", proceedings of IEEE INFOCOM 2001, Anchorage, Alaska, USA, April 22-26, 2001

[10] Das, A.Ghose, A.Razdan, H.Saran, R.Shorey, "Enhancing Performance of Asynchronous Data Traffic over the Bluetooth Wireless Ad-Hoc Network", proceedings of IEEE INFOCOM 2001, Anchorage, Alaska, USA, April 22-26, 2001

[11] N.Johansson, E.Alriksson, U.Jonsson "JUMP Mode- A Dynamic Window-based Scheduling Framework for Bluetooth Scatternets", ACM MobiHoc 2001

[12] N.Golmie, R.E. Van Dyck, and A. Soltanian, "Interference of Bluetooth and IEEE 802.11: Simulation Modeling and Performance Evaluation" in Proceedings of the Fourth ACM International Workshop on Modeling, Analysis, and Simulation of Wireless and Mobile Systems, MSWIM' 01, Rome, Italy, July 2001

[13] S.Lu, T.Nandagopal and V.Bharghavan, "A Wireless Fair Service Algorithm for Packet Cellular Networks", ACM MOBICOM' 98, Dallas, Texas, 1998.

[14] Stefan Zurbes, "Considerations on link and system throughput of Bluetooth networks," in Proceedings of the 11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2000, vol.2, pp.1315-1319