ISSN 2319 – 7757 EISSN 2319 – 7765

Indian Journal of Engineering

Bluetooth Technology – crisis and Solutions: a perspective

Pinki Sethi[1], Annu[2], Himanshu Rehani[3]

1. Senior lecturer, CSE department, Dronacharya college of Engg, Gurgaon, India
2. 6th sem B.tech Student, CSE department, Dronacharya college of Engg, Gurgaon, India
3. 6th sem B.tech Student, CSE department, Dronacharya college of Engg, Gurgaon, India

Received 15 February; accepted 20 March; published online 01 April; printed 16 April 2013

ABSTRACT

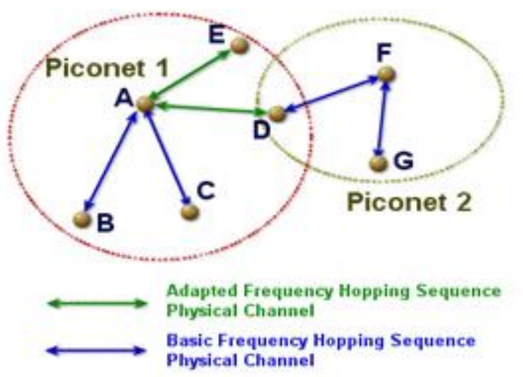
This research introduces a number of problems faced by the Bluetooth technology when attempting to use it for building adhoc networks. The research provides a brief overview of Bluetooth and describes some of the major issues that need to be addressed, if it is to be successful as a networking technology. Some important objectives that any solution must meet are also introduced and motivated. An initial exploration of some key issues such as topology formation and throughput maximization is also provided.

1. INTRODUCTION

1.1. Bluetooth

Bluetooth is a recently proposed standard for short range, low power wireless communication. Initially, it is being envisioned simply as a wire replacement technology. Its most commonly described application is that of a “cordless computer” consisting of several devices including a personal computer, possibly a laptop, keyboard, mouse, joystick, printer, scanner,etc., each equipped with a Bluetooth card (Johansson et al., 2001). There are no cable connections between these devices, and Bluetooth is to enable seamless communication between all them, essentially replacing what is today achieved through a combination of serial and parallel cables, and infrared links (Salonidis et al., 2001). However, Bluetooth has the potential for being much more than a wire replacement technology, and the Bluetooth standard was indeed drafted with such a more ambitious goal in mind. Bluetooth holds the promise of becoming the technology of choice for adhoc networks of the future. This is in part because its low power consumption and potential low cost makes it an attractive solution for the typical mobile devices used in adhoc networks.

This being said, there are many major technical hurdles to cross before this promise can be realized. This paper describes some of the key technical challenges that the Bluetooth technology faces and needs to overcome, if it is to fulfill its potential of becoming more than a wire replacement solution. Although the paper includes some initial research results in this area, it is primarily intended as an overview and possible road map of some of the major issues that must be tackled. This paper is organized as follows. We briefly describe the salient features of the Bluetooth technology in Bluetooth Operation. We describe key technical challenges that need to be addressed for its successful deployment in large scale adhoc networks in Challenges in Bluetooth Design. We discuss certain design objectives in Design Objectives. We describe our research approach in overcoming these challenges and provide some initial results in Distributed Topology Formation Algorithm.



An example of Bluetooth technology is illustrated

1.2. Bluetooth operation

In this section, we briefly describe the basic features of a Bluetooth network. Nodes are organized in small groups called piconets. Every piconet has a leading node called “master,” and other nodes in a piconet are referred to as “slaves.” A node may belong to multiple piconets, and we refer to such a node as a “bridge” (Bhagwat et al., 1999). A piconet can have at most 7 members. Refer to figure for a sample organization. Every communication in a piconet involves the master, so that slaves do not directly communicate with each other but instead rely on the master as a transit node (Salonidis et al., 2001). Inother words, Bluetooth provides a half-duplex communication channel. Communication between nodes in different piconets must involve the bridge nodes. A bridge node cannot be simultaneously active in multiple piconets. It is active in one piconet and “parked” in others (Hajek et al., 1988). Bluetooth allows different activity states for the nodes: active, idle, parked, sniffing. Data exchange takes place between two nodes only when both are active. Activity states of nodes change periodically. Bluetooth uses frequency hopping spread spectrum in the physical layer. Different piconets use different frequency hopping sequences (Das et al., 2001). The frequency hopping sequence of a piconet is derived from the

Pinki Sethi et al.
Bluetooth Technology – crisis and Solutions: a perspective,
Indian journal of engineering, 2013, 3(6), 4-6,

node id and the clock information of the master. A node thus needs to know the identities and the clock information of the masters of all the piconets it participates in. It acquires this information from the master when it joins the piconet.

Synchronization information is also exchanged periodically. The bandwidth of the Bluetooth communication channel is currently 1 Mbps (Johanssan et al., 2001). Nodes in different piconets can transmit simultaneously even if they are within transmission range of each other. This is because they use different frequency hopping patterns. However, there can be only one communication at a time within in a piconet and this communication involves the master and one slave. The master decides the communication order (and duration) for the slaves (Das, et al. 2001). Besides the operation and constraints associated with the Bluetooth communication channel, another key aspect in the context of adhoc networking is the piconet formation process. A node discovers the nodes in its vicinity through the periodic use of an inquiry process that involves transmission using a well known frequency hopping sequence. The inquiry process has two main states: A transmits state and a scan state. In the transmit state, a node continuously transmits its identity as it hops on the different frequencies of the inquiry hopping pattern (Bhagwat et al., 1999). Transmission on each frequency is followed by a short listening period to determine if any node is responding to the inquiry. Nodes will be able to respond if at the time they have their receiver tuned to the frequency of the hopping sequence currently used by the transmitter (Hajek et al., 1988). However, because there is no coordination between nodes, there is no guarantee that two nodes engaged in the inquiry process will be able to hear each other. For one, they could both be in either transmit or scan mode. Furthermore, even when one is transmitting and the other listening, their lack of clock synchronization means that they may not be using the same frequency at the same time. Thus, in order to facilitate synchronization, the sender hops through the frequencies of the frequency hopping sequence faster than the receiver. Once the receiver has learned the identity of a new node as a result of the inquiry process, it transmits its own identity (Miller et al., 2000). Subsequently, if either one of the two nodes decides to involve the other in a piconet in which it is the master, it pages for the other node. The paging message is transmitted on a frequency hopping sequence intended for paging, and is derived from the address of the desired recipient (Salonidis et al., 2001). If the paged node is scanning the same frequency as that on which the paging node is transmitting, then the two synchronize and the recipient receives the information required to join the piconet. Once again the transmitter switches frequencies at a faster rate than the receiver to facilitate the synchronization. Once two nodes belong to the same piconet, their clocks are synchronized and they use the same frequency hopping sequence to exchange information (Bhagwat, et al. 1999).

2. CHALLENGES IN BLUETOOTH DESIGN

The Bluetooth specifications have left several design issues open to implementation, when it comes to its use as a networking technology. The objective is to allow designers' flexibility so as to cater to the individual network requirements. However for adapting the technology towards large scale deployment in adhoc networks it is imperative that there be a systematic procedure for attaining some of the most common design objectives. We first examine the open issues and then discuss why these need to be carefully "nailed down" in order to satisfy certain universal design objectives. There are multiple facets to the decision of how many piconets a node should join. On one hand, bridge nodes that belong to multiple piconets improve connectivity, which reduces the number of communication hops needed to transfer data between any two nodes and can, therefore, improve overall throughput (Miller et al., 2000). On the other hand, the larger the number of piconets a node joins, the larger the associated processing, storage, and most important, communication overhead. This is because a node needs to store certain information about each of the piconets it participates, and furthermore can only be active in one piconet at the time. Specifically, at any one time a node can be active in one piconet and must be parked in the other piconets to which it belongs.

Switching from one piconet to another involves a non-negligible processing overhead (Miller et al. 2000). In addition, while involved in communications in one piconet, a node is unavailable for communications in all the other piconet. This can also affect throughput, albeit this time negatively, as the participation of one node in multiple piconets proportionally reduces the capacity available for communications between any two of the piconets to which it belongs (Hajek et al. 1988). Note that the impact of this constraint also depends on whether the node is involved in piconets. Only as a slave, or whether it is the master of one of the piconets. In the latter case, any period during which the node is acting as a slave in some piconet, corresponds to a communication blackout for all the slaves of the piconet for which it serves as a master. Intuitively, this is an undesirable effect, even if its magnitude depends on the number of nodes involved in the affected piconets.

In this paper, we focus on an initial exploration of some of the above issues that are associated with the problem of "topology formation," when attempting to build an adhoc network based on the Bluetooth technology. These are, however, not the only issues that one would need to address in the context of a Bluetooth adhoc network, and there are many other interesting questions dealing with actual data transmission. For example, how does a master decide the order of data transmission among slaves? Also, as discussed earlier a node can be active in only one piconet at one time. How does a bridge node decide its order of participation in different piconets. The scheduling should be designed so that a master completes its communication with a bridge node while it is active in its piconet (Johanssan et al. 2001). This requires giving priority to bridge nodes as compared to ordinary slaves, and the priority of a bridge node should also depend on the number of piconets it participates in. These issues are closely related to administering different quality of service to different end nodes.

3. DESIGN OBJECTIVES

In this section, we describe some of our design objectives in deciding how to best form Bluetooth topologies, and subsequently discuss the challenges involved in satisfying these objectives while exploiting the flexibility offered by the Bluetooth specifications. Maintaining end to end connectivity whenever feasible, i.e., when there exists a selection of node states (slave, bridge, master) that forms a connected topology, is obviously a desirable feature. Let us examine the challenges involved in achieving this objective within the Bluetooth design constraints. Observe first that any Bluetooth topology must satisfy some basic properties. For one, the partitioning of nodes into masters and slaves implies that the graph associated with any Bluetooth topology is a bi-partite graph. This is because neither masters nor slaves can communicate directly, and therefore the set of nodes associated with masters only has edges to the set of nodes corresponding to slaves. Similarly, the constraint that a piconet cannot contain more than 7 slaves implies that all nodes associated with masters must have a degree less than or equal to 7. This also implies that if at any time the total number of masters is less than one eighth of the total number of nodes, then certain nodes will not belong to any piconet and thus the topology remains disconnected. These are constraints that any topology formation algorithm must take into account. In addition, it is not only the choice of role, i.e., master, slave, or bridge, that is important in determining connectivity, but the order in which nodes are assigned their role is also a key factor. In particular, because connectivity between piconets is ensured through bridge nodes and not all (slave) nodes are capable of playing such a role (the node must be able to "hear" the master of each piconet), connectivity between two piconets may be precluded if the corresponding node attempts to join one of the piconets after the piconet has become full, i.e.,

Pinki Sethi et al.
Bluetooth Technology – crisis and Solutions: a perspective,
Indian journal of engineering, 2013, 3(6), 4-6,

already has 7 slaves. This can possibly be fixed by having some slaves relinquish their membership in the piconet, but identifying when this is needed, e.g., connectivity might still exist between the piconets through a multi-hop path, and which node should leave the piconet, is a complex problem. Achieving connectivity is, therefore, a complex and possibly unachievable task, but it provides a benchmark against which heuristics can be evaluated. In above section, we briefly review how some very basic algorithms perform as we vary a number of system characteristics. Another factor affecting throughput is the number of piconets a node participates in, and as discussed earlier this number should be different for masters and slaves. There are many possible options to consider, but for the sake of simplicity we propose that a master participate in only one piconet, and that a slave participate in up to k piconets, where k is, therefore, the only remaining design parameter. Realistic values fork are probably 2 or 3: This introduces further constraints on the topology construction algorithm, but they are expected to ensure minimum throughput levels in the network.

4. DISTRIBUTED TOPOLOGY FORMATION ALGORITHM

This section is intended as a first exploration of a possible topology formation algorithm. Our goal is not to construct a sophisticated algorithm, but instead to evaluate how a simple and lightweight solution performs under different configurations. The proposed algorithm operates using only local information, and can adapt to changes rapidly. By evaluating its performance, we seek to gain a better understanding of when and why more sophisticated solutions may be needed. Our investigation is carried using a detailed simulation model of the Bluetooth communication channel (Miller et al., 2000). Special attention was given to accurately account for the operation of the inquiry process and how nodes alternate between transmit and scan state. Some randomization was introduced to determine when nodes switch from one state to the other. This randomization was selected so as to ensure that each spends on average the same amount of time in transmit and scan modes. A node can have one of the following states: (i) unassigned, (ii) master, (iii) slave, and (iv) bridge. We assume that every node as a unique ID. The topology formation algorithm operates as follows:

1) Initially all nodes have unassigned states.
2) When two nodes synchronize for the first time and both are unassigned, the one with the highest ID becomes master, and the other node becomes a slave in the piconet of this master.
3) When two nodes synchronize and one is unassigned while the other is a master, the unassigned node joins the piconet of the master if it has less than 7 slaves.
4) When two nodes synchronize and one is unassigned while the other is a slave, the unassigned node becomes the master of a new piconet, and the other node joins the piconet as a slave unless it is already a bridge node inb piconets.
5) If two nodes discover each other and neither is unassigned, then we consider the following cases separately. If both are masters, then neither changes state. If one is a master and the other is a slave in a different piconet, then the slave joins the other piconet and becomes a bridge between the two piconets, provided the slave does not belong to b piconets. Optionally, the master may refuse the new slave if it is already has a bridge to the slave's piconet.

This simple algorithm satisfies all the topology constraints mentioned in the previous section. However, it is not clear how effective it is at meeting the performance objectives outlined in above Section. In the rest of this section, we report some initial simulation based results on its performance when it comes to end-to-end connectivity (Hajek et al., 1988). Our performance metric is end to end connectivity or rather the number of (disconnected) components in the logical Bluetooth topology formed by the above algorithm. This number is obviously lower bounded by the number of components in the "physical topology" graph which contains an edge between a pair of nodes as long as they are within each other transmission range. Ideally, if the physical topology is connected, one would like the logical topology generated by the topology formation algorithm to also be connected. In general, an important goal of any topology formation algorithm is to generate a number of components that is as close as possible to that of the underlying physical topology. Thus, an important performance measure is the difference in the number of components of the logical topology and the physical topology. The smaller this difference, the better the algorithm, at least in terms of connectivity.

5. CONCLUSION

This research is intend as a brief introduction to the many challenges that the Bluetooth technology faces if it is to succeed as a technology for building adhoc networks. We have described many of the issues that need to be tackled and that have been left unspecified by the current standards. We identified a number of objectives that any solution should aim at meeting, and provided an initial investigation of some of these problems. This is obviously preliminary work, and we are actively investigating many of the problems outlined in this paper. We hope that the paper will also entice others in exploring what we feel is a promising and rich research area.

REFERENCES

1. Bhagwat P, Seigal R. A routing vector method (RVM) for routing in Bluetooth scatternets, 1999
2. Das A, Ghose A, Razdan A, Saran H, Shorey R. Enhancing performance of asynchronous data traffic over the Bluetooth wireless ad-hoc network, 2001
3. Hajek B, Sasaki G. Link scheduling in polynomial time, 1988
4. Johansson N, Korner U, Tassiulas L. A distributed scheduling algorithm for a Bluetooth scatternet, 2001
5. Miller, Bisdikian C. Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications, 2000
6. Salonidis T, Bhagwat P, Tassiulas L, Lamaire R. Distributed topology construction of Bluetooth personal area networks, 2001