

# A Modern Study of Bluetooth Wireless Technology

Mrs. Pratibha Singh, Mr. Dipesh Sharma, Mr. Sonu Agrawal

RIT, Raipur, Dept. of Computer science & Eng., RIT, Dept. of inf. Tech., SSCET, durg,  
Dept. of Computer sci. & Eng.

Raipur, (Chhattisgarh), India

er.pratibha@yahoo.in, download.dks@gmail.com, agrawalsonu@gmail.com

**ABSTRACT:** *A Bluetooth ad hoc network can be formed by interconnecting piconets into scatternets. The constraints and properties of Bluetooth scatternets present special challenges in forming an ad hoc network efficiently. This paper, the research contributions in this arena are brought together, to give an overview of the state-of-the-art.*

*Simply stated, Bluetooth is a wireless communication protocol. Since it's a communication protocol, you can use Bluetooth to communicate to other Bluetooth-enabled devices. In this sense, Bluetooth is like any other communication protocol that you use every day, such as HTTP, FTP, SMTP, or IMAP. Bluetooth has a client-server architecture; the one that initiates the connection is the client, and the one who receives the connection is the server. Bluetooth is a great protocol for wireless communication because it's capable of transmitting data at nearly 1MB/s, while consuming 1/100th of the power of Wi-Fi. We discuss criteria for different types of scatternets and establish general models of scatternet topologies. Then we review the state-of-the-art approaches with respect to Bluetooth scatternet formation and contrast them.*

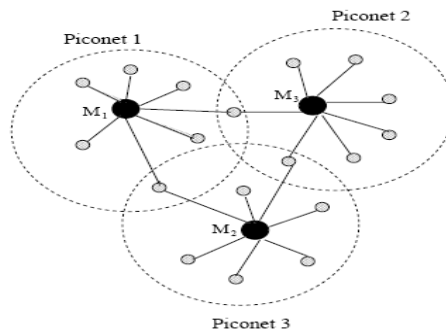
**KEYWORDS:** *Bluetooth, Scatternet formation, Piconet, Ad hoc network.*

## I. INTRODUCTION

Bluetooth is a networking technology aimed at low-powered, short range applications. It was initially developed by Ericsson, but is governed as an open specification by the Bluetooth Special Interest Group. 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. 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. 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 make it an attractive solution for the typical mobile devices used in adhoc networks. Bluetooth is a specification for Wireless Personal Area. It is a way to connect and exchange information and data between mobile phones, laptops, digital cameras and video games. The communication is wireless and has the range of up to 10 meters. Imagine the situation. You go to your office. You connect your notebook to the LAN port. You switch it on. It goes through the entire process of

booting up and then you transfer the data to your desktop computer this around process takes around 10-15 minutes, depending upon speed of your notebook. Bluetooth will also enables to transfer files, photos, and songs from the mobile to other device. The Bluetooth comes in with a wireless headsets and it comes in free with the mobile phone or computer, the wireless headset also useful for people who like to be on the go or while driving the car, as they are hands free. This paper includes some previous work done on bluetooth scatternet. This paper is organized as follows.

We briefly describe the salient features of the Bluetooth technology in Section II. We describe key technical challenges that need to be addressed for its successful deployment in large scale adhoc networks in Section III. We discuss certain design objectives in Section IV, and briefly review the existing research in Section V. In section VI ,We describe Survey on researches that had been done previously and we conclude the paper in section VII.



**Fig. 1.** An example Bluetooth topology is illustrated. The nodes are organized into 3 piconets. The masters of these piconets are M1; M2; M3 respectively. The remaininnodes are the slave nodes or bridge nodes. Slave nodes S1 and S2 can communicate via master M1; Nodes S1 and S3 can communicate via masterM1; bridge B and masterM2;

## II. BLUETOOTH OPERATION

Bluetooth basics is as follows:

- 1.Connection establishment
- 2.Concept of an ad-hoc piconet

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.” A piconet can have at most 7 members. Refer to figure 1 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. In other 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. 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.

Connecting two devices via Bluetooth requires phases:

1).inquiry: This process consists of a sender broadcasting inquiry packets, which do not contain the identity of the Inquiry sender or any other information.

\_ Inquiry Scan: In this state, receiver devices listen for inquiry packets, and upon detection of any such packet, the device broadcasts an inquiry response packet. This contains the identity of the device and its native clock.

2) Page: When paging, a sender device tries to form a connection with a device whose identity and clock are known. Page packets are sent, which contain the sender's device address and clock, for synchronization.

\_ Page Scan: In this state a receiver device listens for page packets. Receipt is acknowledged and synchronization between the devices is established

### *1. WHY IT IS CALLED BLUETOOTH?*

The heart of the Bluetooth brand identity is the name, which refers to the Danish king Harald "Bluetooth" Blaatand who unified Denmark and Norway. In the beginning of the Bluetooth wireless technology era, Bluetooth was aimed at unifying the telecom and computing industries. Bluetooth can be used to wirelessly synchronize and transfer data among devices. Bluetooth can be thought of as a cable replacement technology. Typical uses include automatically synchronizing contact and calendar information among desktop, notebook and palmtop computers without connecting cables. Bluetooth can also be used to access a network or the Internet with a notebook computer by connecting wirelessly to a cellular phone.

## **III. 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. A predominant open issue is how to decide which nodes become masters, slave and bridges. In Bluetooth, nodes are assumed physically equivalent with respect to their Bluetooth capabilities, so that the master and slave states are purely logical. This is a useful feature in the context of adhoc networks where nodes will likely be reasonably homogeneous, but it also introduces several problems. This is because the decision for a node to become slave or master affects the connectivity that will be available to other nodes. In addition, a node needs to decide the number of piconets it should join, and when multiple choices are possible, which subset of piconets to choose. This latter issue arises because a node may have several masters within its communication range. Note that the master of one piconet can participate as a slave in another one. 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. 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. 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. 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 piconet. As a matter of fact, the number of slaves that a piconet should have is itself an open issue. The Bluetooth specification imposes an upper bound on this number (7), but performance considerations should also be taken into account. For one, as discussed above, the number of piconets in which a master participates should be different from that of a slave. In general, even in the absence of any other constraints, e.g., assuming all nodes are capable of communicating with all other nodes, the best (throughput wise) configuration in terms of masters, slaves, and bridges is unclear. Having as few masters as possible can increase the number of nodes that are reachable either directly or in a small number of hops. However, it also means that more nodes are sharing the communication channel associated with each master. Similarly, the number of bridge nodes that should exist between different piconets is also unclear. Many bridges can facilitate load distribution and improve connectivity, but this comes at the cost of increasing the complexity of synchronizing communication Schedules an added overhead when switching from piconet to piconet (recall that a node can be active in only one piconet at the time).

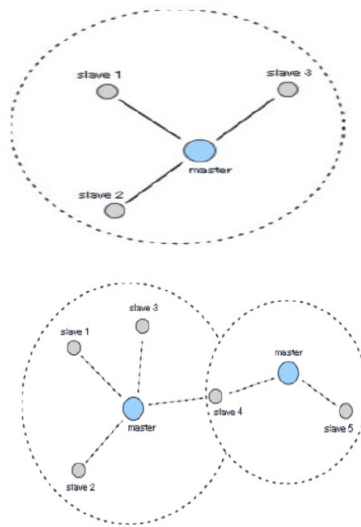


fig 2: piconet and scatternet

For Bluetooth to succeed as a technology on which adhoc networks can be built, it is not only essential to find light-weight solutions to the above problems, but those solutions must be fully distributed. In other words, they should not assume the existence of a central entity with access to the entire system/network state, and nodes decisions should only be based on information about their own state and that of their “neighbors.” However, the definition of what a node’s neighborhood consists of is itself not clear. Does it consist only of nodes belonging to the same piconet(s), or does it also include other nodes within communication reach? More generally, a neighborhood could be defined as all nodes that are  $k$  or less “hops” away (hop count corresponds to the number of masters/piconets that need to be traversed). Clearly there is a trade-off between the accuracy (or optimality) of the decisions that can be made under different scenarios. In

general, the more information is available, the better the decisions. However, this comes at the cost of a higher latency, a higher processing cost, and a higher control overhead. It is, therefore, important to identify a design point that is both implementable and capable of providing a reasonably efficient operational solution. One of our goals is to start exploring the space of potential solutions to identify the range of available options.

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?. 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. 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.

## **2. ARCHITECTURE AND ITS TECHNICAL WORKING:**

A simplified view of the bluetooth protocol stack is presented in figure 1.it shows the layers that correspond to the hardware and software components of bluetooth solution. On a PC or PDA the interface between the two is a physical PC bus such as a USB, compact-flash, or PC card bus. The hardware portion of the stack consists of the radio, base band controller, and link manager protocol (LMP). The LMP is used to set up and control the link and implement the bluetooth link-level security. The upper layers of the stack consist of logical link control and adaptation protocol (L2CAP), client protocols, and application profiles. The L2CAP segments and reassembles data into packets for transmission. It also interfaces with client protocols such as the bluetooth service discovery protocols (SDP), which enables applications to discover which services are available on bluetooth device, and RFCOMM, which enables a bluetooth device to emulate a serial port. Finally, application “profiles” define how particular user scenarios are accomplished. Although shown as an upper application layer in the simplified diagram, a profile can be viewed as a vertical slice through the protocol stack. A profile specifies mandatory option and parameters for each protocol. This approach decreases the risk or interoperability problems between different bluetooth devices.

## **IV. DESIGN OBJECTIVES**

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. We are primarily concerned with three major objectives:

1. Connectivity,
2. Distributed operation and low overhead,
3. Throughput maximization.

Next, we briefly expand on those three objectives, and what it takes to achieve them.

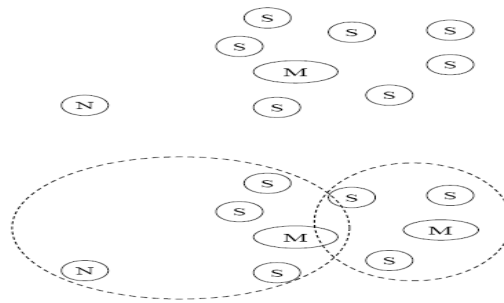
Maintaining end to end connectivity whenever feasible, i.e., when there exists a selection of node states (slave, bridge, master) that forms a connected topology, is obviously a desirable feature. Let us examine the challenges involved in achieving this objective within the Bluetooth design constraints. 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 both 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. 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., 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.

Our second design objective, namely a distributed operation and low overhead, is a must for any practical solution. As pointed out earlier, node state changes should be triggered in response to changes in the physical topology. Figure 2 gives an example of how the roles of existing nodes need to be changed to accommodate the arrival of a new node and maintain connectivity. In many instances, detecting and adjusting to topological changes is likely to require a certain amount of communications between nodes. One approach to minimizing overhead is to seek algorithms that rely only on local information, and hence have minimal communication overhead. However, it is unlikely that such simplistic algorithms will be able to efficiently accommodate all possible scenarios. As a result, they will need to incorporate additional design objectives to compensate for their limited decision horizon. For example, a simple strategy would be to seek topologies that have significant redundancy, e.g., connectivity between piconets is achieved through multiple bridges or by having nodes serving as bridges between multiple piconets. Similarly, trying to keep piconet sizes small can improve the odds of success of local strategies.

Our third design objective of maximizing throughput, while obviously desirable, unfortunately adds complexity of its own to an already complex problem. For example, the size of piconets, which plays a role in both determining connectivity and the overhead of any algorithm responsible for maintaining connectivity, also affects the throughput of the network. Consider a piconet with  $k$  slaves, and where every slave generates a traffic of intensity  $r$  per unit time.

such a configuration, the master needs to support a load of  $2kr$  per unit time assuming it itself does not generate any traffic (the load on the master increases if the master generates traffic). If the master has a bandwidth of  $B$ , then we must have  $2Kr \leq B$  and thus the nodal throughput  $r$  that the piconet supports is inversely proportional to the number of members in the piconet. This would call for keeping  $K$  small, and hence building a topology with many small piconets. On the other hand, a large number of small piconets will lead to long end to end routes, and this in turn may overload the transit piconets and, therefore, also limit the feasible nodal throughput. In general, the selection of the “right” size for piconets depends on how traffic is distributed between nodes and where nodes are located. For example, it is obvious that if nodes  $A$  and  $B$  are within communication range of each other and need to exchange a significant amount of traffic, then they should be assigned to the same piconet. However, other simple configurations do not necessarily yield similarly simple answers. For example, assuming a set of  $N$ ,  $N$  nodes all capable of communicating with each other and a uniform traffic. Pattern, the best topology for such a configuration is not obvious.



**Fig. 3.** The effect of arrival of a new node N is illustrated. The nodes labeled S are slaves in a piconet with master labeled M: The new node N is within the transmission range of M only. The piconet has the maximum possible number of members and thus M can not accept node N as its slave. Two different piconets with masters labeled M need to be formed now.

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 for  $K$  are probably 22 or 33: This introduces further constraints on the topology construction algorithm, but they are expected to ensure minimum throughput levels in the network.

## V. RELATED RESEARCH

In this section, we mention very briefly a number of previous works that have also been motivated by the need to extend the standard specifications, if the Bluetooth technology is to be used in building adhoc networks.

Salonidis *et al.* presents a distributed topology construction scheme in Bluetooth networks [6]. The basic assumption behind the scheme is that all nodes are within transmission range of each other. The nodes conduct a leader election algorithm.

The winner knows the identity of all nodes and uses this information to design the desired topology. Thus the algorithm is not scalable if the number of nodes is large. This paper also shows that the average delay involved in synchronizing two nodes (the time spent in the inquiry and the page sequences before the nodes are able to exchange the clock information) is infinite if the nodes have a deterministic sequence of switching between inquiring and inquired (or paging and paged) modes. Bhagwat *et al.* presents a source routing mechanism for Bluetooth networks [1]. Das *et al.* [2] and Johanson *et al.* [4] present distributed scheduling policies for Bluetooth networks.

## VI. CASE STUDY

This section surveys the current state-of-art for bluetooth scatternet formation platforms.

In [1] A routing protocol which utilizes the characteristics of Bluetooth technology is proposed for Bluetooth-based mobile ad hoc networks. The routing tables are maintained in the master devices and the routing zone radius for each table is adjusted dynamically by using evolving fuzzy neural networks. Observing there exists some useless routing packets which are helpless to build the routing path and increase the network loads in the existing ad hoc routing protocols, they selectively use multiple unicasts or one broadcast when the destination device is out of the routing zone radius coverage of the routing table. The simulation results show that the dynamic adjustments of the routing table size in each master device results in much less reply time of

routing request, fewer request packets and useless packets compared with two representative protocols, Zone Routing Protocol (ZRP) and Dynamic Source Routing (DSR).

In [2] work targets small mobile computers with a Bluetooth wireless link. Embedded in cheap robots with data rich sensors, our target does not have enough processing power to do the required analysis on sensor data. We propose the use of parallel processing. In this paper they outline DynaMP, a dynamic message passing architecture. Using an ad-hoc network with on-demand routing based on AODV. DynaMP has a resource discovery mechanism; it distributes code and data using Java class loading, with a caching mechanism to reduce network traffic. They assume the network is unreliable and provide a retry mechanism in distributing the problem.

In [3] this paper studies the optimization of scatternets through the reduction of communication path lengths. After demonstrating analytically that there is a strong relationship between the communication path length on one hand and throughput and power consumption on the other hand, we propose a novel heuristic algorithm suite capable of dynamically adapting the network topology to the existing traffic connections between the scatternet nodes. The periodic adaptation of the scatternet topology to the traffic connections enables the routing algorithms to identify shorter paths between. They evaluate their approach through communicating network nodes, thus allowing for more efficient communications simulations, in the presence of dynamic traffic flows and mobility.

In [4] The vision of ad-hoc networking with Bluetooth includes the concept of devices participating in multiple "piconets" and thereby forming a "scatternet". However, the details of scatternet support for Bluetooth are not specified yet. This paper presents a scheme for Bluetooth scatternet operation that adapts to varying traffic patterns. Basing on sniff mode, it does not require substantial modification of the current Bluetooth specification and may thus be incorporated into currently available Bluetooth products. They present simulation results that confirm the applicability of our approach to realistic scenarios.

In [5] With the growth in the number of devices with an integrated Bluetooth module, the range of applications based on the Bluetooth technology becomes larger, going beyond peer-to-peer use-cases. This paper considers a hybrid network, consisting both of infrastructure and ad hoc parts, referred to as scatternet with infrastructure support. The introduction of the scatternet structure allows to extend the coverage and to enable access of a larger number of users. The formation algorithms are discussed and the importance of synchronization of the formation process for creation of a height- and width-balanced tree topology is illustrated. Simulation results presenting the impact of the link establishment policies on the resulting topology are given.

In [6] this paper addresses the problem of scatternet formation for single-hop Bluetooth based personal area and ad hoc networks, with minimal communication overhead. In a single-hop ad hoc network, all wireless devices are in the radio vicinity of each other, recent scatternet formation schemes by Li, Stojmenovic and Wang are position based and were applied for multi-hop networks. These schemes are localized and can Construct degree limited and connected piconets, without parking any node. They also limit to 7 the number of slave roles in one piconet. The creation and maintenance require small overhead in addition to maintaining location information for one-hop neighbors. In this article they apply this method to single-hop networks; by showing that position Information is then not needed. Each node can simply select a virtual position, and communicate it to all neighbors in the neighbor discovery phase. Nodes then act according to the scheme by Li, Stojmenovic and Wang using such virtual positions instead of real ones. In addition, in this paper they use Delaunay triangulation instead of partial Delaunay



triangulation proposed in , since each node has all the information needed. Likewise, they can also apply Minimum Spanning Tree (MST) as the planar topology in our new schemes. Finally, they design experiments to study both the properties of formatted scatternets (such as number and the performances of different localized routing methods on them. The experiments confirm good functionality of created Bluetooth networks in addition to their fast creation and straightforward maintenance.

## VII. CONCLUSION

This paper was intended 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 and also gives the small description of related work that had been done in this area.. 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.

## ACKNOWLEDGMENTS

This work was partially supported by Mr. Dipesh Sharma who is Reader of Raipur institute of technology. My special thanks to the Mr. Dipesh sir who have contributed towards development of this work.

## REFERENCES

- [1] A Bluetooth Routing Protocol Using Evolving Fuzzy Neural Networks□ presented by Chenn-Jung Huang, Wei-Kuang Lai, Sheng-Yu Hsiao and Hao-Yu Liu.
- [2]Parallel Computation in Mobile Systems Using Bluetooth Scatternets and Java presented by Rob Shepherd, John Story and Dr Saad Mansoor.
- [3] Hop Count Based Optimization of Bluetooth Scatternets presented by Csaba Kiss Kall\_, Carla-Fabiana Chiasserini, Sewook Jung.
- [4]Bluetooth Scatternets: An Enhanced Adaptive Scheduling Scheme presented by Simon Baatz, Matthias Frank, Carmen K`uhl, Peter Martini, Christoph Scholz
- [5]Bluetooth Scatternet with Infrastructure Support:Formation Algorithms given by Tatiana K. Madsen, Fjølur Gudmundsson, Stefan Sverrisson, Hans P. Schwefel and Ramjee Prasad
- [6]Bluetooth Scatternet Formation for Single-hop Ad Hoc Networks Based on Virtual Positions Given by Yu Wang\* Ivan Stojmenovic† Xiang-Yang Li\*
- [7] B. Miller and C. Bisdikian. *Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications*. Prentice-Hall, 2000.
- [8] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. Lemaire. Distributed topology construction of Bluetooth personal area networks. In *Proceedings of INFOCOM'2001*, 2001.
- [9] T. Salonidis, P. Bhagwat, L. Tassiulas, Proximity awareness and fast connection establishment in bluetooth, in: First Annual Workshop on Mobile and Ad Hoc Networking and Computing, MobiHOC 2000, 2000, pp. 141–142.
- [10] A. Aggarwal, M. Kapoor, L. Ramachandran, A. Sarkar, Clustering algorithms for wireless ad hoc networks, in: Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, Boston, MA, USA, 2000, pp. 54–63.
- [11] M. Kalia, D. Bansal, R. Shorey, Data scheduling ans sar for bluetooth mac, in: IEEE 51st Vehicular Technology Conference Proceedings, VTC 2000-Spring, Tokyo, vol. 2, 2000, pp. 716–720.
- [12] A. Capone, M. Gerla, R. Kapoor, Efficient polling schemes for bluetooth picocells, in: IEEE ICC 01, vol. 7, Helsinki, Finland, 2001, pp. 1990–1994.
- [12] N. Golmie, N. Chevrollier, I. ElBakkouri, Interference aware bluetooth packet scheduling, in: Proceedings of IEEE GLOBECOM, 2001, pp. 2857–2863.