Bluenet – a New Scatternet Formation Scheme

Zhifang Wang, Robert J. Thomas, Zygmunt Haas
ECE Cornell Univ, Ithaca, NY, 14853
{zw18,rjt1}@cornell.edu, haas@ece.cornell.edu

Abstract

Bluetooth is a new promising local area wireless technology designed to enable voice and data communication among various electronic devices. We believe that Bluetooth networks will provide reliable, flexible and cost-efficient telecommunication support for the post-deregulation electric power systems. Though not specified in version 1.0 of the Bluetooth specification, communication by way of multi-hop routing (so characteristic of ad hoc networks) within a scatternet will offer a new and exciting extension to this technology. And the topology of such an ad-hoc scatternet would have a significant effect on the overall performance of the network.

In this paper we present "Bluenet" as a novel and practical scheme for building an efficient scatternet and discuss the basic rules followed by the Bluenet scheme. Two methods are introduced to evaluate the performance of the resulting scatternets base on average shortest-path length and maximum traffic-flows respectively. Finally the effectiveness of the Bluenet scheme is demonstrated through simulations and comparison.

I. Introduction

Electric power systems have been dependent on communication links between computers and computers and between humans and computers for a long time. In the past, information generated by the data has been used to operate the system reliably and at a reasonable cost. In the new world of a restructured industry where economic information is used to set operating points and therefore determine system reliability, communication is even more important. The ability to economically create reliable, robust and secure networks will to a large measure, determine the efficiency and reliable nature of the resulting system. Networks for the electric power business must, accommodate many players associated with the production and delivery of electricity in a common service area, accommodate a distributed decision-making environment for power generation and consumption, enable coordination among decision makers to ensure effective system operation, enable operation in a way that attains maximum engineering efficiency and facilitate supplier and customer cooperation in an integrated supply-side and demand-side management scheme. To facilitate this paradigm of distributed decision-making we require, among other things, that the network support a timely collection of data, comprehensive communication links to all participants, sufficient computing power for real-time analysis and an effective real-time control capability.

In this paper we discuss a new algorithm for the formation of a so-called ad-hoc wireless network. We focus on wireless because of the costs associated with fiber networks in certain power system applications, because it is a promising new architecture, because it contains a set of protocols that are compatible with those of the Utility Communication Architecture (UCA) and because it promises to provide a secure and robust channel for data and information flow at low cost.

During recent years, there has been a substantial demand for a low cost, universal and wireless communication technology to enable various devices to communicate seamlessly without wires. This demand has led to a rapid development of the Bluetooth System. In July 1997, the Bluetooth Special Interest Group (SIG), which is now joined by nearly 1200 companies, published an open specification for Bluetooth wireless communication that is publicly available (http://www.bluetooth.com/) and royalty free. Due to the release of the specification and strong commercial support, Bluetooth radio links are expected to be available in a wide variety of products in the near future. And it is believed that Bluetooth will become the major technology for short-range wireless networks and wireless personal area network. Though not explicitly addressed in the specification, Bluetooth wireless communication provides an enabling technology for multi-hop ad hoc networks. It is a promising technology upon which to base the construction of inexpensive and large ad hoc networks, especially when the low cost of Bluetooth chips (about $5/chip) is taken into account. [1,2,3] This is especially true for power system substation applications where installing fiber is prohibitively expensive.

0-7695-1435-9/02 $17.00 (c) 2002 IEEE

A Bluetooth radio node operates in the unlicensed Industrial-Science-Medical (ISM) band at 2.45 GHz and adopts frequency-hop transceivers to combat interference and fading. In U.S., the ISM band is divided into 79 separate RF channels, each with a 1-MHz bandwidth. In most other countries, the ISM band is smaller and therefore contains only 23 channels. The nominal radio range of Bluetooth is 10 meters with a transmit power of 0 dBm and can be extended up to 100 meters with an amplified transmit power of 20 dBm. [1,3]

According to the standard, the basic structure for communication in a Bluetooth network is the *piconet*, a simple one-hop star-like network, which contains one *master* unit and up to 7 active *slave* units. All the active Bluetooth units inside a piconet share the same 1-Mbps Frequency Hopping Spread Spectrum (FHSS) channel in a time-division multiplexing scheme which is centrally controlled by the master unit. [1,2,3,6,7]

If multiple piconets cover the same area, some Bluetooth units may participate in more than one piconet (but can be a master in only one piconet) on a time sharing basis. Such a unit serves as a *bridge* between the overlapping piconets in proximity. A group of piconets in which connections exist between different piconets is referred to as a *scatternet*. A sample scatternet network is visualized in Figure 1.1.

Time multiplexing must be used for a bridge unit to switch between piconets. Because of the need to re-synchronize its radio from one piconet to another and to perform the necessary signaling, the bridging unit necessarily wastes some of its capacity while switching. This represents an important performance limitation in building scatternets.

Since two Bluetooth units cannot communicate directly before a master-slave relation is established between them (even if they are in the proximity of each other's radio range), communications within a large-scale Bluetooth network have to be based on a scatternet. To maintain a scatternet inevitably costs some of the network's resources. Therefore, an efficient scatternet requires tradeoffs between keeping a decent level of connectivity and reserving enough network resources for communications.

A method for forming an efficient scatternet under a practical networking scenario is still an open issue. There are very few papers that address this problem in the current literature. In [4], G. Miklos et al consider randomly generated topologies of Bluetooth nodes and investigate the possible correlation between scatternet formation rules and scatternet performance through a number of simulation studies. In [5], G. Zaruba et al introduce "*Bluetrees*" as a practical protocol for forming connected scatternets, which has two variations, namely, *Blueroot Grown Bluetree* and *Distributed Bluetree*. The former builds a scatternet starting from some specified node called *Blueroot*, while the latter speeds up the scatternet formation process by selecting more than one root for tree formation and then merging the trees generated by each root. One distinct feature of the Bluetree scheme is that all resulting scatternets assume a topology of spanning tree, where the parent node is master and the children nodes are slaves, as depicted in Figure 1.2. Though the scheme selects the smallest possible number of links to form a connected scatter and tries to spend the least of network resources on maintaining the scatternet, the resulting scatternet has inherent deficiency due to its hierarchical structure. First, it lacks reliability. If one parent node is lost, all the children and grandchildren nodes below it will be separated from the rest of the network and part of the tree or even the whole tree has to been rebuilt in order to retain the connectivity. In a mobile network, this may happen quite frequently, making the Bluetree very susceptible. Second, it lacks efficiency in routing because all the routing paths have to traverse the tree in upward and downward directions. This becomes even worse in a larger system. Third, the parent nodes in Bluetrees are very likely to become communication "bottlenecks" and make it difficult for the network to afford multiple communication pairs.[1]

In this paper we propose a new scatternet formation scheme – "*Bluenet*". Through a 3-phase algorithm, Bluenet constructs the scatternet in a distributed way; i.e., there is no need to designate any root node and it can be carried out at each node based only on the local knowledge of the node's neighbors. Unlike the hierarchical structure in Bluetrees, Bluenets is a much flatter structure. The resulting scatternets maintain a degree of connectivity while avoiding wasting network resources on too many redundant links. Though the Bluenet seems to spend a little more on maintaining scatternet links, the simulation shows that it can carry far more communication traffic than Bluetree.

The remainder of the paper is organized as follows: In Section II we present "Bluenet" as a novel and practical scatternet formation scheme and justify the rules used in the algorithm. Section III introduces the performance evaluation for the resulting scatternets based on average shortest-path length and maximum traffic-flows. In Section IV the effectiveness of the Bluenet scheme is demonstrated through a simulation comparison with Bluetree based on the suggested evaluation method in Section III. Finally the concluding remarks are given in Section V.

---

[1] For convenience, sometimes we refer to the resulting scatternets from Bluetree scheme also as "Bluetree". The same is true for Bluenet.

II. Bluenet -- a new Bluetooth Scatternet Formation Scheme

II.1. Essential Background Information

When considering the problem of scatternet formation, we have to keep in mind that two Bluetooth units cannot exchange information freely unless a master-slave relationship has been set up between them. In order to avoid the excessive delays of forming connections between potential communication pairs, a scatternet network must be formed in advance, if possible, with necessary routing information collected and stored.

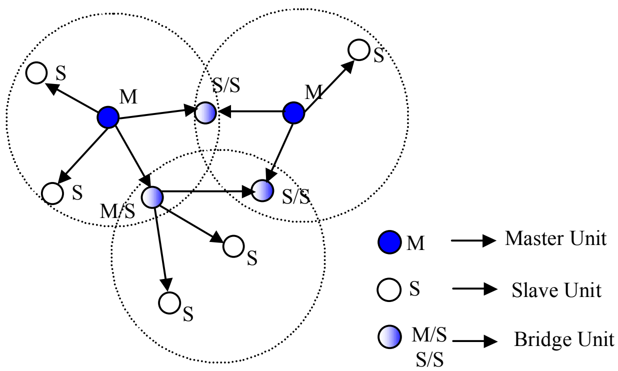


Fig. 1.1 Bluetooth Scatternets

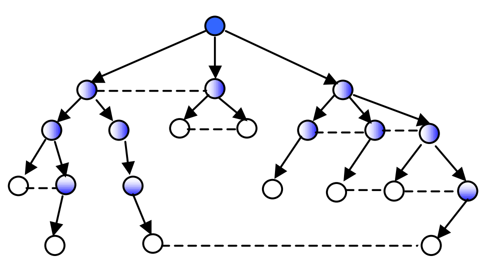


Fig. 1.2 A Rooted Bluetree

For a given set of Bluetooth units that are spread randomly in a specific geographical region, the *visibility graph network* is defined as the network consisting of all the units and all the potential links within the Bluetooth's radio range[2]. The *scatternet network* consists of all the master-slave links, which in most cases are only a small fraction of the set of all the potential links. Otherwise too many network resources will be consumed maintaining those links. In our study, we mainly consider the *intra piconet overhead* and the *bridge overhead*. Intra piconet overhead is associated with the master in a piconet that has to perform polling and coordinating among all its slaves. Bridge overhead is associated with the bridge node switching delay between different piconets. Generally, the intra piconet overhead is much smaller compared to the bridge overhead. Therefore some researchers neglect the former, taking into account only the latter.

Starting from an original geographical distribution, all Bluetooth units are in the *standby* state; i.e., are not associated with any piconet. To get connected, they have to go through the *inquiry* and *page* states. In the *inquiry* state, a unit learns about the identity of other devices within its radio coverage area (its geographical neighbors). In the *page* state, a unit explicitly invites another device to join the piconet in which the inviting unit will become the master. The process of inquiry and paging may still be going on even after a piconet has been formed, so that bridge links can be set up and a scatternet built. For more detailed information on the Bluetooth operational states, readers are refer to the chapter 6 in [3] (page 86-87).

II.2. Bluenet Algorithm

The goal of our scatternet formation scheme is to form an efficient scatternet, which has reasonably good connectivity but preserves, as much as possible, network resources for communications. Ideally, the network resources should be spread as evenly as possible throughout the scatternet, so as to prevent bottlenecks from occurring in the later multiple-pair communications. In order to achieve this goal, a Bluenet scheme forms the scatternet based on the following rules:

Rule-1. Avoid forming further piconets inside a piconet;

Rule-2. For a bridge node, avoid setting up more than one connections to the same piconet;

Rule-3. Inside a piconet, the master tries to aquire some number of slaves; i.e. not too many and not too few while maintaining a connection only to active slave nodes if possible.

Since the communications inside a piconet is already well coordinated by its master, any further piconet inside the piconet only wastes the capacities of those nodes participating in it. Therefore rule-1 is applied to prevent such a condition. It can be done easily if the master just broadcasts the list of all its slaves in the piconet, so that each slave will be aware that some of its geographical "neighbors" may belong to the same piconet and will not either page them or answer their page.

Rule-2 is adopted in order to avoid excessive bridge overhead for the bridging units. But to implement rule-2,

2 The definition of visibility graph network is suggested by G. Miklos et al in [4].

we need to make a trivial modification to the paging process; i.e., when the bridge node tries to page some node belonging to a different piconet, that node should be able to provide the identity of its own piconet. Therefore the bridge node can determine whether to invite that node or not. If such a modification is not available, rule-2 can still be implemented by setting up a temporary master-slave link for the exchange of necessary identity information and then determining whether to keep the link or not. The latter alternative may cause additional time delays in the setting-up process.

The main consideration of rule-3 is that the size of piconets may affect the final structure of a scatternet. If the size is too small, there will be many piconets in the system, which means that many inter-piconet links have been set up to maintain the connectivity. This will result in a large amount of bridge overhead. Alternatively, if the piconet size is large, say 7 (the largest permitted), the master tends to become a bottleneck in the network because all the communication between its slave nodes (either intra or inter piconet communications) will have to go through the master. In addition, keeping inactive slave members in a piconet should be avoided because it wastes intra piconet overhead for the master.

Based on these rules, the Bluenet Algorithm is organized into three phases as following:

- *Phase1*: Initial piconets formed with some separate Bluetooth nodes left.

First all the nodes are void; i.e., no a master and no slave in any piconet. During the inquiry state, every Bluetooth node collects information about its neighbors within radio range; i.e. forms a local visibility graph. Then the Bluetooth nodes enter the page state randomly, trying to invite n ($\le N_{\max}$) of its neighbors to join its future piconet. Once a node becomes a slave of some piconet, it will stop paging or answering pages until instructed to begin again by the master. When phase-1 is finished, many separate piconets are formed throughout the system, with some Bluetooth nodes left unconnected to any piconet and all of whose "neighbors" are associated with a piconet. Some nodes become a master in a piconet and has at most $N_{\max}$ slaves. The master then obtains the necessary information about all the slaves in its piconet and broadcasts this information to its slaves. This prevents slave nodes from forming a piconet inside their own piconets later.

- *Phase2*: Separate Bluetooth nodes get connected to initial piconets.

During this phase, every separate Bluetooth node begins to page all of its neighbors (but selects at most $N_{\max}$ of them to become its slaves eventually) and tries to get connected to some initial piconets built in phase-1. If it gets connected to more than one piconet, it becomes a bridge node. After phase-2, all Bluetooth nodes get associated with at least one piconet.

- *Phase3*: Piconets get connected to form a scatternet.

At this point the master of each piconet instructs their slaves to set up outgoing links. When phase-3 is done, the whole Bluetooth system becomes interconnected with a very high probability if the original visibility graph network is a connected graph.

III. Performance Evaluation of Scatternets

In this section we discuss two different but related metrics for evaluating the performance of a scatternet after it has been formed. One is called *Average Shortest Path* (ASP) and the other is called the *Maximum Traffic Flows* (MTF). The effectiveness of a specific scatternet formation scheme can be demonstrated by evaluating the appropriate metric for the resulting scatternets.

III.1. Average Shortest Path (ASP)

The ASP metric is defined as the average shortest path-length (hop counting) among all 2-node pairs in a Bluetooth network. Obviously ASP relates only to the network topology. ASP can be calculated by using any shortest path algorithm. In our study, we adopt the Dijkstra algorithm. It is apparent that for a given set of Bluetooth nodes, the minimal ASP, termed as ASP_0, is obtained from the visibility graph network, i.e., where all the potential links are utilized. Since usually the scatternet network contains only a small fraction of all potential links, it is almost certain that the ASP for the scatternet formed by any algorithm will be larger than ASP_0. We define the ASP ratio r to be

$$r = ASP_{sct} / ASP_0 , \qquad (1)$$

where ASP_{sct} is the actual ASP of the scatternet being evaluated

III.2. Maximum Traffic Flows (MTF)

It is also very important to learn about the information-carrying capacity of a communication network. That is why we choose MTF for evaluating the performance of a scatternet. But before giving the definition of MTF, we need to introduce *the multi-commodity flow* and Ford-Fulkerson Algorithm first.

Define a network as $W=(N,A,c)$, where N is the set of nodes, A is the set of all arcs (links), and $c: A \to \Re^+$ or $c: A \cup N \to \Re^+$ is the capacity function. Designating a particular node s as the source, and another particular node t as the sink, we wish to know the maximum amount of information that can be transmitted, per unit time, from s, through the network, to t, without violating the link and node capacity limits. This is called the *maximum flow* problem, which can be solved very conveniently and efficiently by the Ford-Fulkerson Algorithm.[10] [11]

Now, assume that there exist a set of sources S and a set of sinks T in the system, with the restriction that information transmitted from the source $S(i)$ must be directed to the sink $T(i)$, for $i=1,\cdots,length(S)$. The information flowing between the pair $S(i) \to T(i)$ is called the flow of a certain commodity then the network has many different commodities flowing simultaneously. The problem of maximizing the sum of the multiple-commodity flows is very difficult, see [11]. But some simple greedy algorithms can be used to approximate the optimal solution, which will be discussed later in this section with more details.

In a scatternet, communications may be going on simultaneously between multiple source-sink pairs. *Maximum Traffic Flow* (MTF) is a method of evaluating the information-carrying capability of the network by calculating all possible *maximum multi-commodity flows* in the system. For convenience, we refer to the maximum multi-commodity flow also as MTF in later discussion.

Suppose we have an n-node Bluetooth network and C_m denote the set of all possible m-commodities in the system, then the size of C_m is given by

$$|C_m| = \frac{n!}{(n-2m)!\,m!\,2^m}, \text{ for } m=1,2,\cdots,\lfloor n/2 \rfloor . \quad (2)$$

Table 3.1 lists some examples of $|C_m|$ for n = 40 and different m's.

Table 3.1 Examples of $|C_m|$ for n=40

m	$\lvert C_m \rvert$
2	2.7417e+005
4	8.0750e+009
6	5.8075e+013
8	1.2740e+017
10	9.0252e+019
12	1.9876e+022
14	1.1926e+024
16	1.4758e+025
18	2.0256e+025
20	3.1983e+023

Note that the size of all possible m-commodities can be very large even for a moderate-size network. Fortunately, through simulations we found that for the generated scatternets, the statistical characteristics (such as mean-value and deviation) of MTF's from random samples of m-commodities become quite stable when the sample size is large enough. For example, when n=40, and m=2~10, the sample size of 6,000 is large enough to obtain an accurate estimates of the statistics of MTF with an error < 1%.

Now we describe the greedy algorithm that is used to approximate the MFT for an m-commodity flow in an n-node network. Initially the network condition is $W=(N,A,c)$, with m commodities specified in $S \to T$.

a. Calculate all maximum flows for all the commodities in $S \to T$; by using Ford-Fulkerson Algorithm. Also keep a record of the consumption of link capacities and node capacities by the maximum flows.
b. Select the commodity pair with the largest maximum flow, then the remove the commodity from $S \to T$; and decrease the corresponding link capacities and node capacities from c.
c. Repeat step a & b until there is no commodity pair left in $S \to T$. The MTF will be the sum of the largest maximum flows obtained from each run of b.

IV. Simulation and Comparison

In what follows, we examine the suggested scatternet formation scheme by numerical simulations in Matlab. First, n Bluetooth nodes are randomly located within a rectangular area A of size ($X \times Y$ $meter^2$). The nodes are uniformly placed in A and remain stationery during the simulation. (In this paper we confine our study to stationery Bluetooth networks and will deal with mobility in the future research.) Then the visibility graph network V can be determined according to the Bluetooth radio coverage R. Throughout our simulation, a radio range of 10 meters is assumed with 0 dBm transmitter power. Next, scatternets are built by using the "Bluenet" or "Bluetree"[3] scheme. Finally, the performance of resulting scatternets is evaluated and compared based on the metrics introduced in Section III. Simulations are carried out with a typical set of parameters defined in Table 4.1. The parameter $N_{\max}$ is chosen to be 5 because the simulations show that such a

[3] For the Bluetree scheme, we only choose one of its variations, the "Blueroot Grown Bluetree", to form scatternets, in which the Blueroot is selected randomly from all Bluetooth nodes. Because the only difference between the two variations of Bluetree scheme is their building time and the resulting scatternets assume almost the same properties.

setting will result in best scatternet performance on the average, which is coincident with the analysis in [5].

Figure 4.1 shows a typical visibility graph network for a set of 40 Bluetooth nodes, in which the dotted lines represent all the potential radio links. Different scatternets can be formed according to either a Bluenet or Bluetree scheme. As examples, we plot one resulting scatternet from each scheme in Figure 4.2 and Figure 4.3 respectively. In these figures, the solid circles represent master nodes while the empty circles represent slave nodes. All the scatternet links can carry two-way communications. The arrows on them are only to depict the mater-slave relation, i.e., pointing from a master to it slaves. Comparing Figure 4.2 and Figure 4.3, we can find that the Bluetree scheme always establishes a spanning-tree scatternet, which contains only (n-1) links. While the Bluenet scheme tends to build a mesh-like scatternet, which utilizes a little more links and assumes more balanced network structure.

Table 4.1 Parameters of Bluetooth network

	Value	Description
n	40	Total number of Bluetooth nodes
$X \times Y$	20 × 20	Area size ($meter^2$)
R	10	Radio range ($meter$)
$N_{\max}$	5	Max number of slaves in a piconet

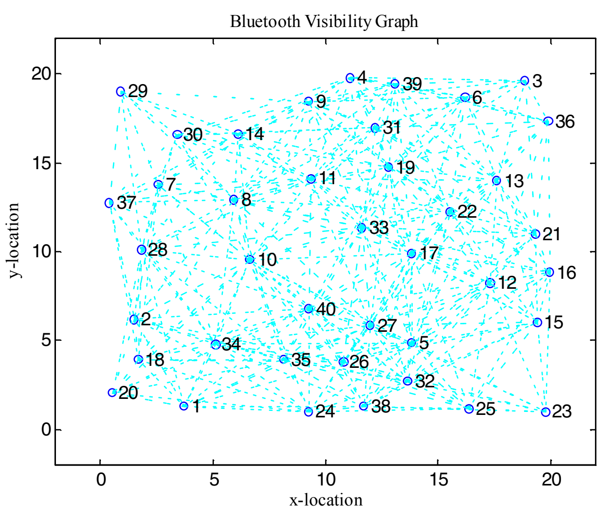


Fig 4.1. The Visibility Graph Network

IV.1. On Average Shortest Path

Given the original distribution as in Figure 4.1, 200 scatternet samples are generated according to the Bluenet scheme and the Bluetree scheme respectively, and the corresponding ASP ratios are then calculated. We exhibit the results in Figure 4.4. The ASP ratios of the Bluenets are symbolized by "•" and those of the Bluetrees by "×". From the figure we can see that most of the former fall below the latter, with only very few exceptions. The mean ASP ratio of all the Bluenets is about 2.08 and that of the Bluetrees is about 2.31. The 0.23 decrease in the mean ASP ratio demonstrates that the Bluenets are more efficient with respect to shortest-routing hops.

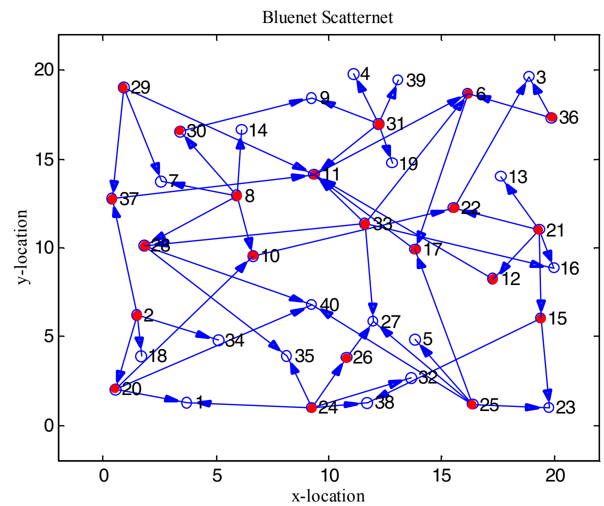


Fig 4.2. A Sample Bluenet

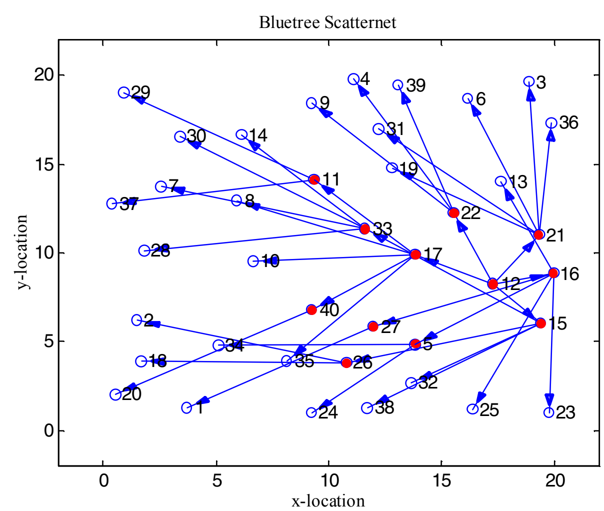


Fig 4.3. A Sample Bluetree

IV.2. On Maximum Traffic Flows

The procedure for MTF evaluation is illustrated in Figure 4.5. Starting from the same geographical distribution of a set of n (=40) Bluetooth nodes, k (=6) different scatternets are established according to the Bluenet and the Bluetree scheme respectively. Then each of the resulting scatternets is submitted to MTF Evaluation Package, which computes the MTF's for 6,000 randomly generated multi-commodities and calculates the statistics of all these traffic flows, i.e., the mean value and deviation.

In order to solve the maximum multi-commodity flow problem, we need to know the capacity function of a scatternet. Since the real capacity-limitations in a scatternet are its node capacities, we simply set all the link capacities equal to $C_0 = 1000Kbps$. The intra piconet overhead is assumed to be $\Delta B_1 = 10Kbps$ for each slave that a master node holds; the bridge overhead be $\Delta B_2 = 100Kbps$ for each additional piconet that a bridge node joins. Then the capacity of the i-th Bluetooth node can be written as:

$$C^i = C_0 - n_s^i \cdot \Delta B_1 - I_{bridge}^i \cdot (n_p - 1) \cdot \Delta B_2, \ i = 1,2,\cdots,n \ .(4)$$

where n_s^i is the number of its slaves if node-i is a master, otherwise n_s^i is equal to 0; $I_{bridge}^i = 1$ if node-i is a bridge unit and 0 otherwise; and n_p (>1) is the total number of piconets that a bridge node connects.

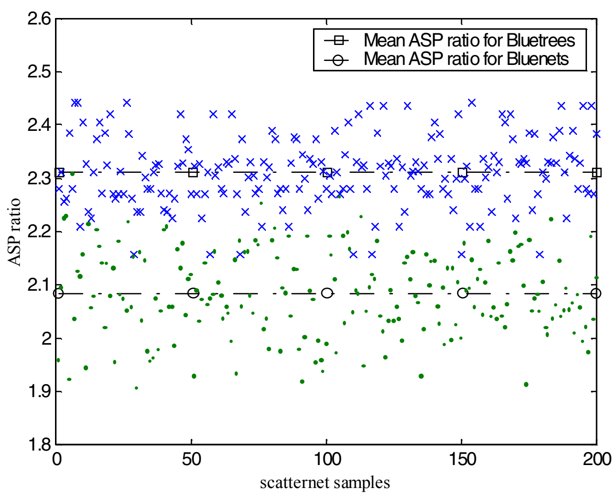


Fig 4.4. ASP Ratios for Bluenets and Bluetrees

In Figure 4.6 we present the evaluation results for one statistical Bluetooth test network[4]. Graph (a) is a plot of statistical mean of the MTF's and graph (b) is a plot of statistical deviation of the MTF's. In both graphs, the 6 dotted lines with "…" represent the data from 6 Bluenets and those with "–." represent the data from 6 Bluetrees, respectively. The solid lines represent the average performance of all six Bluenets or Bluetrees.

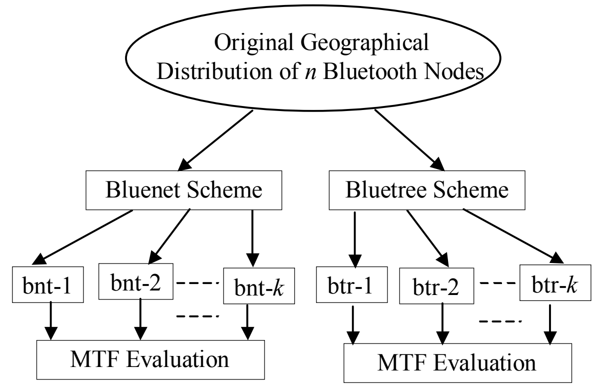


Fig 4.5. MTF Evaluation Procedure

Figure 4.6 shows that the 6 different scatternets generated from the "Bluenet" scheme have very consistent MTF performance as are those from the "Bluetree" scheme. But the Bluenets can accomodate larger traffic flow than the Bluetrees. On average the magnitude of difference is about 200 Kbps in single-commodity flows and about 400 Kbps in multi-commodity flows. On the other hand, from Figure 4.6(b) we find that except for the case of a single-commodity flow, the deviation of MTFs of the Bluenets and the Bluetrees are very close. (The smaller deviation of single-commodity MTFs of the Bluetrees may be caused by its tree-like topology. Because all single-commodity traffics go through very similar tree-paths in the network.) In order to show that the comparison in Figure 4.6 is not just a special case, we also present the MTF evaluation results from two other Bluetooth test networks (with different geographical distributions) in Figure 4.7 and Figure 4.8. Note that they are quite similar to each other. In our studies, we also repeat the simulation of multiple Bluetooth test networks and get comparison results similar to those shown in Figure 4.6~8. Therefore it should be safe to conclude that the Bluetnets are more efficient than the Bluetrees in terms of their information-carrying capability.

V. Conclusion and Further Discussion

In this paper we present a new scatternet formation sheme – "Bluenet". By following three basic rules, the Bluenet sheme manages to reduce as much of the bridge overhead as possible while maintaining a reasonable degree of connectivity in the resulting scatternet. As to the performance evaluation of the scatternet, we choose the Average Shortest Path to reflect its routing efficiency and

[4] A "Bluetooth test network" means a set of 40 Bluetooth nodes with a random geographical distribution in A, just like the one shown in Figure 4.1.

the Maximum Traffic Flows to reflect its information-carrying capability.

The efficiency of the scheme is demonstrated by comparing it with a previously proposed scatternet formation scheme – "Bluetree". The simulations (in Matlab) are performed on typical Bluetooth test neworks consisting of 40 Bluetooth nodes which are randomly placed in a $20\times20\,meter^2$ area. Through the simlulations, we show that the bluenets provide better routing efficiency by utilizing more links than the bluetrees. And more importantly, the bluenets are able to carry more communication traffic than the bluetrees due to its balanced network topology.

During our study, we discovered that the evluation of MTF imposes up a very heavy computation burden. For example, one run of MTF evaluation of a 40-node Bluetooth scatternet takes about 20 cpu hours using a Pentium III 550 Mhz/250 MB system. In our futher research, the technique of important sampling will be considered to improve the computation efficiency.

It would be useful if the efficiency of our scatternet formation scheme could be proved by comparing its closeness to an ideal upper bound of MTF performance given the same orignal Bluetooth network. Unfortunately to find such an ideal upper bound is is still an open issue and very difficult to solve. This will also become a future subject in our on-going research.

References

[1] Bluetooth Special Interest Group, "Specification of the Bluetooth System", version 1.0B, http://www.bluetooth.com/.

[2] J. Haarsten, "Bluetooth – the universal radio interface for ad-hoc wireless connectivity", *Ericsson Review*, No.3 1998.

[3] B. A. Miller, and C. Bisdikian, *Bluetooth Revealed*, Prentice Hall PTR, Upper Saddle River, NJ 07458, 2001.

[4] G. Miklos, A. R. Racz, Z. Turanyi, A. Valko, and P. Johansson, "Performance Aspects of Bluetooth Scatternet Formation", *MobiHoc 2000*, pp147-148, Boston, Aug 2000.

[5] G. V. Zaruba, S. Basagni, and I. Chlamtac, "Bluetrees – Scatternet Foramtion to Enable Bluetooth-Based Ad Hoc networks", *ICC 2001*, vol.1, pp 273-277, June 2001.

[6] S. Mateo, J. Y. Wilson, and J. A. Kronz, "Inside Bluetooth: Part I", *Dr. Dobb's Journal*, vol. 25, issue.3, pp62-70.

[7] S. Mateo, J. Y. Wilson, and J. A. Kronz, "Inside Bluetooth: Part II", *Dr. Dobb's Journal*, vol. 25, issue.4, pp58-64.

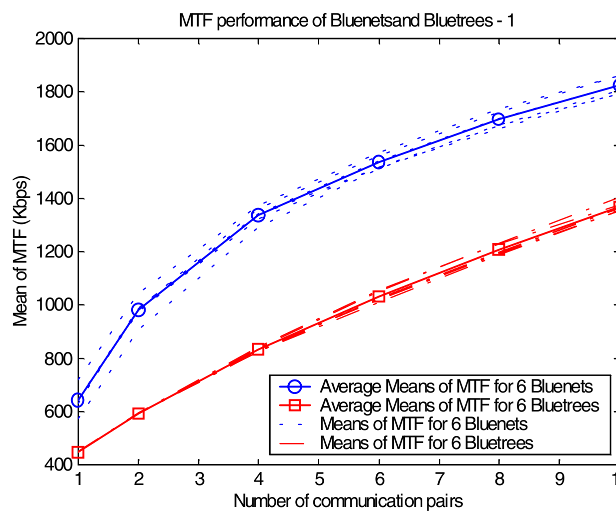
[8] B. Raman, P. Bhagwat, "Arguments for Cross-Layer Optimizations in Bluetooth Scatternets", *Proc. 2001 Symposium on Applications and the Internet*, pp176-84.

[9] S. Zurbes, W. Stahl, K. Matheus, and J. Haartsen, "Radio Network Performance of Bluetooth", *ICC 2000*, pp 1563-1567.

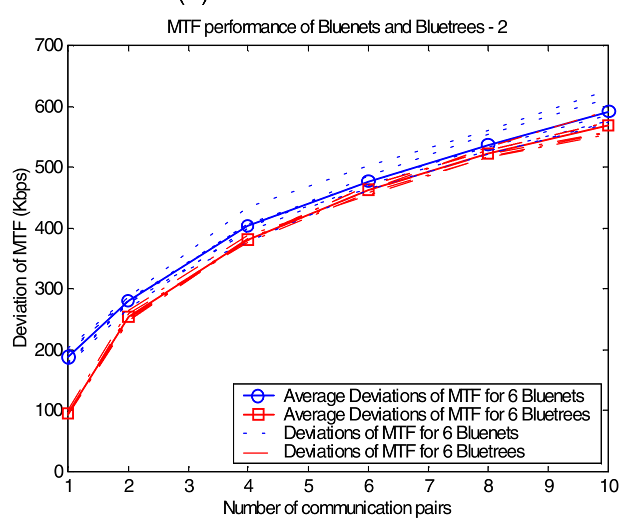
[10] P. Bhagwat and A. Segall, "A Routing Vector Method (RVM) for Routing in Bluetooth Scatternets", *MoMuC 1999*, pp375-379.

[11] R. H. Jones and N. C. Steele, *Mathematics in Communication Theory*, Ellis Horwood Limited, Market Cross Houre, Cooper Street, Chichester, West Sussex, PO 19 1EB, England, 1989.

[12] T. C. Hu, *Integer Programming and Network Flows*, Addison-Wesley Publishing Company, 1969.

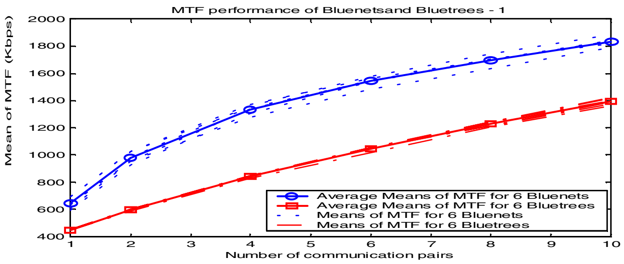


(a) The Means of MTF

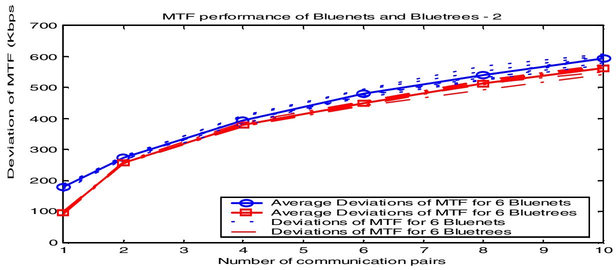


(b) The Deviations of MTF

Fig 4.6 MTF Evaluation Results for Bluetooth Network I

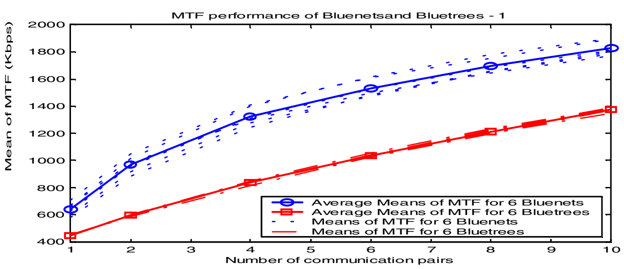


(a) The Means of MTF

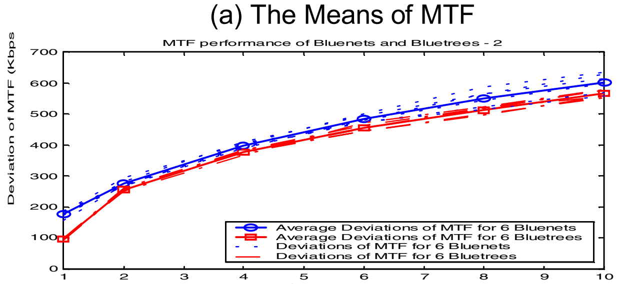


(b) The Deviations of MTF

Fig 4.7. MTF Evaluation Results for Bluetooth Network II



(a) The Means of MTF



(b) The Deviations of MTF

Fig 4.8. MTF Evaluation Results for Bluetooth Network III