Performance Evaluation of a New Scatternet Formation Protocol for Multi-hop Bluetooth Networks

Stefano Basagni
Dept. of Elec. and Computer Eng.
Northeastern University
basagni@ece.neu.edu

Raffaele Bruno
Institute for Informatic and Telematic
C.N.R. Pisa, Italy
bruno@guest.cnuce.cnr.it

Chiara Petrioli
Dipartimento di Informatica
Università di Roma "La Sapienza"
petrioli@dsi.uniroma1.it

Abstract

In this paper we evaluate the performance of a new protocol for scatternet formation in Bluetooth multi-hop networks. The protocol implements the three fundamental steps to scatternet formation, namely, device discovery, piconet formation and piconet interconnection. A Bluetooth extension to ns2, the VINT project simulator, has been developed to address all the details pertinent to scatternet formation. The performance evaluation allows us to assess the impact of the device discovery duration over the scatternet formation process, to evaluate the time for the protocol to generate a connected scatternet, and to investigate metrics that pertain to the "quality" of the scatternet, such as the average number of piconets, the average number of slaves per piconets, the average number of roles assumed by each node and the average length of the routes between any two Bluetooth devices.

Keywords

Bluetooth, scatternet formation, performance evaluation.

INTRODUCTION

Among the new technologies for wireless communication in the unlicensed ISM band (2.4GHz), the Bluetooth (BT) technology is expected to be one of the fastest growing. Bluetooth is a low cost, low power, short-range radio technology originally introduced as a cable-replacement technology for connecting small devices (headsets and cell phones, portable computers and printers, etc.).

The BT technology, as described in the Specifications of the Bluetooth System Version 1.1, includes several features that makes it one of the most promising technology for enabling multi-hop wireless networks, or, simply, ad hoc networks. The way the BT technology is used to form ad hoc topologies is via forming a scatternet. If two BT devices are into each others communication range (i.e., they are neighbors), in order to set up a communication link, one of them assumes the role of master of the communication and the other becomes its slave. This simple "one-hop" network is called a piconet, and may include several slaves, no more than seven of which can be actively communicating with the master at the same time. A scatternet is finally formed by joining piconets. The inter-piconet connection is enabled by the possibility for a BT device to have multiple roles: a node can be a master in one piconet and a slave in multiple piconets. The devices with multiple roles will act as gateways to adjacent piconets.

Among the solutions proposed so far in the literature for solving the scatternet formation problem, some rely upon the unrealistic assumption that all BT devices are in each other transmission range ("single-hop" topologies, as in [1], [2], [3]). Among the solutions that apply to the more general case of multi-hop topologies, the scatternet formation protocol described in [4] requires that the protocol is initiated by a designated node (the blueroot) and generates a tree-like scatternet. The blueroot starts the formation procedure by acquiring as slaves its one hop neighbors. These, in turn, start paging their own neighbors (those nodes that are two hops from the root) and so on, in a "wave expansion" fashion, till the whole tree is constructed. To the best of the authors' knowledge, the only solutions for scatternet formation in multi-hop BT networks that produce topologies different from a tree are those presented in [5], [6] and [7]. The main aim of the protocol proposed in [5] is to build up a connected scatternet in which each piconet has no more than 7 slaves (i.e., the maximum number of active slaves that each master may have at the same time). To this purpose, degree reduction techniques are initially applied to the network topology graph so to reduce the number of wireless links at each node to less than 7. A scatternet formation protocol (which is left unspecified) is then executed on the resulting topology. These techniques require each node to be equipped with additional hardware that provides to the node its current (geographic) location (e.g., a GPS receiver). Beyond being potentially expensive, this solution is not feasible when such extra hardware is not available. The scatternet formation scheme proposed in [6], BlueNet, produces a scatternet whose piconets have a bounded number of slaves. After an (unspecified) device discovery phase, some of the nodes enter the page state randomly (they will be masters) trying to invite a bounded number of slaves to join their piconet. During successive phases, nodes that have not selected their role during the first phase try to connect to some already formed piconet. Finally, the master of each piconet instructs its slaves to set outgoing links to neighboring piconets to form a scatternet. The connectivity of the resulting scatternet is not guaranteed (i.e., not all the BlueNets are connected, even when the initial topologies are).

None of the three protocols outlined above is detailed in the papers, neither from an algorithmic point of view, nor from the perspective of BT implementation. Furthermore, no thorough performance evaluation is given.

In this paper we evaluate the performance of the scatternet formation protocol described in details (protocol and BT implementation) in [7]. The protocol produces connected scatternets that have a mesh-like topology in three successive phases: 1) When a BT device starts its operations, it enters the device discovery phase, in which it discovers neighboring BT devices. 2) Using the knowledge of its one-hop neighbors each nodes starts the piconet formation phase, which leads to the assignment of the roles of either master or slave to each node, resulting in a set of disjoint piconets that covers the entire network. 3) The final phase, scatternet formation, concerns the selection of gateway nodes for interconnecting adjacent piconets into a connected scatternet.

For the performance evaluation we have enhanced the VINT Project network simulator ns2 [8] and its IBM open-source extension BlueHoc [9] to implement the entire BT protocol stack and all those operations needed for performing the protocol. Simulations have been conducted which include 1. measuring the time needed for scatternet formation (the three phases); 2. assessing the effect of the duration of the device discovery phase on the entire scatternet formation process, 3. counting the average number of piconets; 4. counting the average number of slaves per piconets; 5. counting the average number of roles (either master or slave) assigned to each node, and 6. comparing the average length of the routes between any two BT devices in the scatternet with respect to the average shortest path length between any pairs of nodes in the original network topology.

The rest of the paper is organized as follows. In the next section we give some details on the protocol. In the following section, we describe the simulation environment and the results of our performance evaluation.

A SCATTERNET FORMATION PROTOCOL

In this section we describe the three phases of the scatternet formation algorithm introduced in [7].

The first phase, device discovery, aims at gathering at each device the knowledge of its one-hop neighbors (namely, their unique ID, other synchronization information, and information needed later for performing the protocol). For the correctness of the following phases of the protocol, neighbors' knowledge must be symmetric, in the sense that if node u knows its one-hop neighbor v, then v knows u as well.

Device discovery is performed in BT by using the inquiry procedures. For two nodes to discover each other, they must be in opposite modes, i.e., one must be in inquiry mode and the other one in inquiry scan mode. Two are the main problems to be solved here: 1) No indication is given in the specifications on how to guarantee that neighboring devices are in opposite inquiry modes. 2) The inquiry message broadcast by the source does not contain any information about the source itself. Thus, once two neighboring devices complete an inquiry handshake, only the source knows the identity of the device in inquiry scan mode, not vice versa (asymmetric knowledge).

To overcome these drawbacks and attain mutual knowledge, we use a mechanism similar to that introduced in [1]. Each device is allowed to alternate between inquiry mode and inquiry scan mode, remaining in each mode for a time selected randomly and uniformly in a predefined time range. The operations while in each of the two modes are those as described in the specifications. When two nodes in opposite inquiry modes handshake, they set up a temporary piconet that lasts only the time necessary to exchange their ID and other relevant information (thus achieving the required mutual knowledge). (The parameters relevant to this phase are discussed in the next section).

The second phase of the protocol partitions the network nodes into disjoint piconets. Master selection is performed locally based on a nodes's unique ID, its weight, i.e., a real number ≥ 0 that is computed by each node and indicates how suitable the node is to serve as a master, and the ID and weights of its one-hop neighbors (this information has been gathered during the first phase). All nodes which have the biggest weight among their neighbors become masters. Once a node decides to be a master, it tries to recruit its one-hop neighbors as slaves in its piconet. In particular, if a node u has some "bigger neighbors" (i.e., neighboring nodes with bigger weight) which decided to be masters and invite u to join their piconet, then u becomes a slave and joins the piconet of the first master that pages it. Otherwise, u itself becomes a master. This phase leads to the formation of piconets no masters of which are neighbors.

During the third phase masters collect from their slaves information about other masters at most three hops away (called "neighboring masters"). This information is needed for establishing with these masters interconnections which guarantee the connectivity of the resulting scatternet [10]. The interconnections are obtained via sharing one slave between masters that are two hops apart (these slaves are called gateway slaves) or by setting up an extra piconet between two slaves of masters that are three hops away (intermediate gateways). Pairs of neighboring masters adopt consistent rules of gateway selection and instruct their gateways to set up the appropriate connection toward neighboring masters [7].

PERFORMANCE EVALUATION

Simulation environment

To evaluate the performance of our protocol we have developed a Bluetooth extension to the VINT project network simulator ("ns2") [8]. We based our extension on BlueHoc, the ns2-based simulator released by IBM [9]. In particular, in order to implement the operations of our solution for scatternet formation, we have enriched BlueHoc with mechanisms for *a*) giving to each node the possibility to dynamically assume either the role of master or the role of slave; *b*) handling collisions that might arise during the establishment of a link, and *c*) having a node alternating between inquiry and inquiry scan. (These functions are not available in BlueHoc.)

In the simulated scenarios, n Power Class 3 BT nodes (i.e., nodes with maximum transmission radius of 10 meters) are

randomly and uniformly scattered in a geographic area which is a square of side L. We make the assumption that two nodes are in each other transmission range if and only if their Euclidean distance is $\leq$ 10m. We call visibility graphs the topologies generated by drawing an edge between each pair of BT devices that are in each other transmission range.
In our simulations, the number of BT nodes n has been assigned the values 30, 50, 70, 90 and 110, while L has been set to 30m. This allowed us to test our protocol on increasingly dense networks, from (moderately) sparse networks, where only 95% of the visibility graphs are connected ($n = 30$), to highly dense networks. Figure 1 shows the average degree of the nodes in the visibility graphs when varying the number of nodes in the network. The average degree ranges from 27.9 for $n = 110$ downto 7.4 when $n = 30$. As the density increases, the average shortest path length in the visibility graph slightly decreases (10%) from 2.37 to 2.14, as shown in Fig 2.
All results presented in this section were obtained by running the protocol over 300 generated visibility graphs.

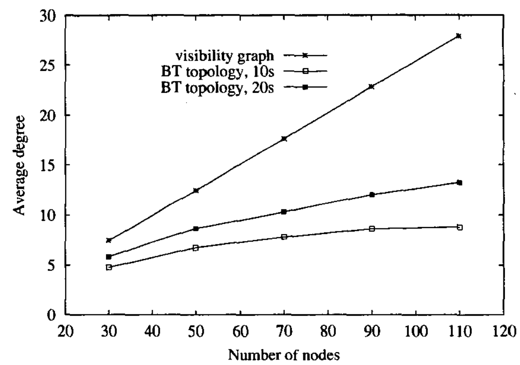

Figure 1. Average degree of visibility graphs and BT topologies after $10s$ and $20s$.

Simulation results

Our simulations concern the two main aspects of scatternet formation, namely, device discovery and piconet formation and interconnection.

• Device discovery in multi-hop networks.

We have run the device discovery phase for a predefined time T_{disc} over each visibility graph. Nodes alternate between inquiry and inquiry scan mode, spending a variable time, uniformly and randomly selected in the interval $(0.02s, 2s)$, in each mode. The resulting topology, which we call a BT topology, has links only between those pairs of BT nodes that were able to discover each other during the device discovery phase.

The effects of different discovery phase durations are shown in figures 1, 2, and 3. In Figure 1 we have depicted the average degree of the BT topologies discovered after $10s$ and $20s$. We notice a significant decrease of the average degree of the BT topologies with respect to the degree of the visibility graphs. When $T_{\text{disc}} = 10s$, the decrease ranges from 35% ($n = 30$) to 68.5% ($n = 110$). As T_{disc} increases, we notice a less marked decrease (from 21.4%, $n = 30$ to 52.6%, $n = 110$ when $T_{\text{disc}} = 20s$). The reasons of such behavior are mainly due to the difficulty of discovering all neighbors, especially in case of extremely dense networks. The higher the density of the network the more the probability of collisions of inquiry messages. As the length of the device discovery phase increases the number of discovered neighbors also increases. However, the rate of discovered neighbors decreases with time, since there is a larger probability that two neighbors that already discovered each other would do it again.

A similar reasoning explains the results depicted in Figure 2. The average shortest path length of the BT topologies is 20% to 33% (8% to 18%) higher than the corresponding measure in the visibility graph for $T_{\text{disc}} = 10s$ ($T_{\text{disc}} = 20s$) when n varies from 30 to 110.

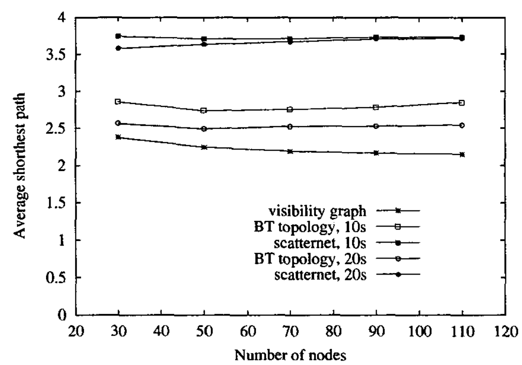

Figure 2. Average shortest path length.

Despite it may take a long time to discover all the neighbors of a node, a shorter time suffices for discovering enough neighbors so that the resulting BT topologies are connected. This is shown by Figure 3: When $T_{\text{disc}} \geq 8s$ all the BT topologies are connected in case of moderately dense to dense visibility graphs ($n \geq 50$ nodes). For $n = 30$, which corresponds to a sparse scenario (in which less than 95% of the visibility graphs are connected), the discovery of a very high percentage of neighbors is required to maintain connectivity. However, the neighbors discovered in 8 seconds are already enough to produce connected BT topologies the 95% of the times.

In general, device discovery for multi-hop BT networks is indeed a time consuming operation. This is mostly due to the following three limitations imposed by the technology: a) the need to adopt (stochastic) mechanisms to have neighboring nodes in opposite inquiry modes, so they can discover each other; b) the impossibility of identifying the inquirer, which demands the construction of a temporary piconet between neighbors that discovered each other already, and c) the

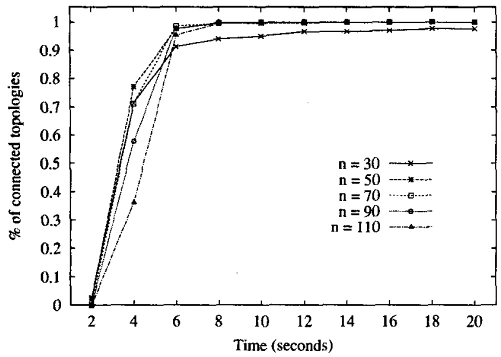

Figure 3. Percentage of connected BT topologies vs. T_{disc}.

overly long duration of the backoff interval as stipulated in the BT specifications (2048 clock ticks). In our performance evaluation we have quantified the impact of the backoff duration on the performance of device discovery. The results are shown in Figure 4. We observe that, by just decreasing the backoff duration to one fourth of the value specified by the standard, an impressive increase of the number of discovered neighbors is achieved that leads to connected topologies in less than $4s$.

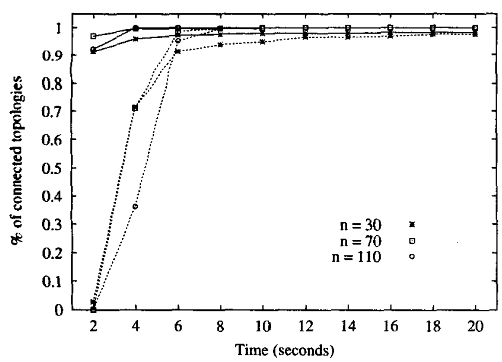

Figure 4. Effect of the backoff interval length on the percentage of connected BT topologies.

• Scatternet formation and interconnection.
In this part of the performance evaluation we have investigated the time needed to complete the protocol, and a set of metrics identified in the literature as measures of the "quality" of the resulting scatternet: *a*) the average number of piconets in the scatternet, *b*) the average number of roles assigned to each node, *c*) the average number of slaves per piconet, and *d*) the average length of the routes in the scatternet, and its comparison to the average length of the routes in the BT topologies. We also evaluate the impact of the device discovery phase duration on the quality of the scatternets produced by the last phases of the protocol.

The duration of the piconet formation and interconnection phases is depicted in Figure 5. We have plotted the average duration over the last two protocol phases and the 95th percentiles when T_{disc} is set to $10s$ and $20s$. As expected, the duration increases with the number of nodes and with the duration of the device discovery phase. When any of these parameters increase the number of neighbors to contact (page) and from which to wait for a response also increases. In any case, the total duration of the last two phases of our protocol is always below $1.8s$ when $T_{\text{disc}} = 20s$ and below $1.2s$ when $T_{\text{disc}} = 10s$. These figures refer to the 95th percentile, the average being well below.

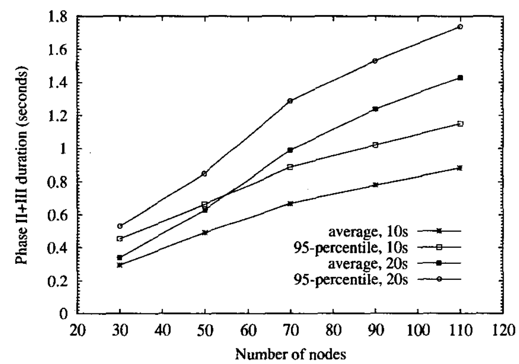

Figure 5. Duration of the second and third phases.

Figure 6 shows the average number of piconets generated by our protocol during the phase of piconet formation as well as the average number of piconets in the generated scatternet. We have considered BT topologies at $10s$ and $20s$. The two cases show similar trends. The number of piconets generated in each of the two phases, and totally, increases with the number of nodes. The number of piconets generated in the second phase ranges from 28% to 33% (23% to 28%) of the nodes for $T_{\text{disc}} = 10s$ ($T_{\text{disc}} = 20s$). As the number of nodes (and thus the number of piconets) increases, a higher number of extra piconets is needed to interconnect neighboring piconets via intermediate gateways. The number of the needed extra piconets fall short of the number of piconets generated in the second phase for highly dense networks. While T_{disc} increases, we observe a general decrease in the number of generated piconets. This is due to the fact that the BT topologies are increasingly denser (more neighbors are discovered), which leads to a more efficient partitioning of the network into piconets (second phase) and to a higher probability that a node between two masters can be selected as a gateway slave.
The average number of slaves per piconet is depicted in Figure 7, together with the 95th percentile of the number of slaves. In this case, the increased number of links obtained with a longer device discovery leads to a slight increase both in the average number of slaves per piconets, and, more importantly, in the size of the "bigger" piconets. However, in

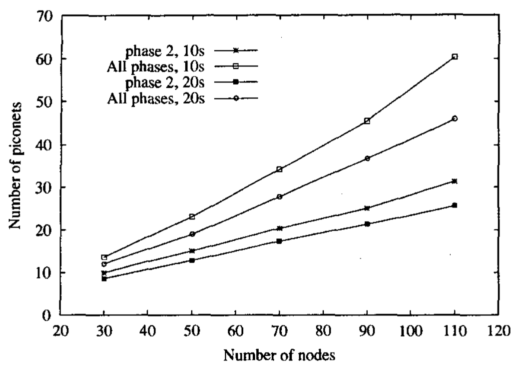

Figure 6. Average number of piconets.

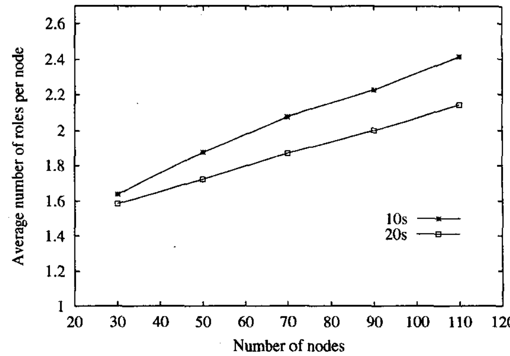

Figure 8. Average number of roles per node.

all the simulated scenarios, the 95th percentile of the number of slaves per piconet remains below 12, ranging from 6 ($n = 30, T_{\text{disc}} = 10s$) to 12 ($n = 110, T_{\text{disc}} = 20s$).

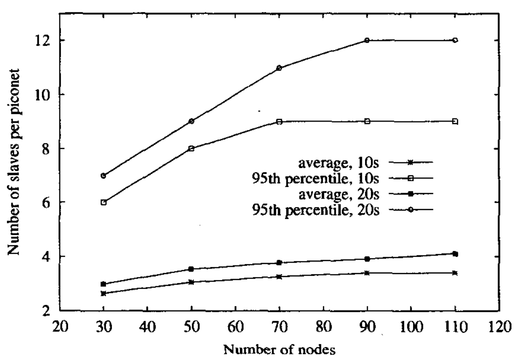

Figure 7. Number of slaves per piconet.

A critical performance measure for Bluetooth Scatternets is the (average) number of roles assigned to each node. A high number of roles per node translates into reduced throughput performance since nodes can be active only in one piconet at a time.

In Figure 8 we show the average number of roles assumed by BT devices in the scatternet produced by our protocol. The average number of roles per node slightly increases with n and decreases with T_{disc} to take into account the higher number of adjacent piconets that need to be joined (and thus the number of gateways). However, the average number of roles assumed by each node is always very low, and never exceeds 2.4.

Figure 2 shows the increase of the average shortest path length in the final scatternet with respect to the same measure in the BT topologies. We notice an increase of the average length of the shortest paths in the scatternet that ranges from 31% to 36% ($T_{\text{disc}} = 10s$) and from 40% to 46% ($T_{\text{disc}} = 20s$). Increased route lengths are mostly due to the piconet-based network organization (which may force nodes which are close in the visibility graph to communicate through a much longer interpiconet route) and to the need for all communications in a BT piconet to pass through the master (two neighboring slaves that belong to the same piconet cannot communicate directly).

REFERENCES

[1] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. LaMaire, "Distributed topology construction of Bluetooth personal area networks," in Proceedings of the IEEE Infocom 2001, Anchorage, AK, April 22–26 2001, pp. 1577–1586.

[2] C. Law, A. K. Mehta, and K.-Y. Siu, "A new Bluetooth scatternet formation protocol," ACM/Kluwer Journal on Mobile Networks and Applications (MONET), 2002, to appear.

[3] G. Tan, "Interconnecting Bluetooth-like personal area networks," in 1st Annual Student Oxygen Workshop, Gloucester, MA, July 16 2001.

[4] G. Záruba, S. Basagni, and I. Chlamtac, "Bluetrees—Scatternet formation to enable Bluetooth-based personal area networks," in Proceedings of the IEEE International Conference on Communications, ICC 2001, Helsinki, Finland, June 11–14 2001.

[5] I. Stojmenovic, "Dominating set based Bluetooth scatternet formation with localized maintenance," in Proceedings of the Workshop on Advances in Parallel and Distributed Computational Models, Fort Lauderdale, FL, April 2002.

[6] Z. Wang, R. J. Thomas, and Z. Haas, "Bluenet—a new scatternet formation scheme," in Proceedings of the 35th Hawaii International Conference on System Science (HICSS-35), Big Island, Hawaii, January 7–10 2002.

[7] C. Petrioli, S. Basagni, and I. Chlamtac, "Configuring bluestars: Multihop scatternet formation for Bluetooth networks," IEEE Transactions on Computers, special issue on Wireless Internet (Y.-B. Lin and Y.-C. Tseng, eds.), 2002, in press.

[8] The VINT Project, The ns Manual, http://www.isi.edu/nsnam/ns/, 2002.

[9] IBM, BlueHoc: Bluetooth Ad Hoc Network Simulator, Version 1.0, http://www-124.ibm.com/developerworks/projects/bluehoc, June 2001.

[10] I. Chlamtac and A. Faragó, "A new approach to the design and analysis of peer-to-peer mobile networks," Wireless Networks, vol. 5, no. 3, pp. 149–156, May 1999.