Distributed Construction and Maintenance of Bandwidth and Energy Efficient Bluetooth Scatternets

Metin Tekkalmaz, Hasan Sözer, and Ibrahim Korpeoglu, *Member*, *IEEE*

Abstract—Bluetooth networks can be constructed as piconets or scatternets depending on the number of nodes in the network. Although piconet construction is a well-defined process specified in Bluetooth standards, scatternet formation policies and algorithms are not well specified. Among many solution proposals for this problem, only a few of them focus on efficient usage of bandwidth in the resulting scatternets. In this paper, we propose a distributed algorithm for the scatternet formation problem that dynamically constructs and maintains a scatternet based on estimated traffic flow rates between nodes. The algorithm is adaptive to changes and maintains a constructed scatternet for bandwidth-efficiency when nodes come and go or when traffic flow rates change. Based on simulations, the paper also presents the improvements in bandwidth-efficiency and reduction in energy consumption provided by the proposed algorithm.

Index Terms—Bluetooth, scatternet formation, ad hoc networks, network topology, algorithm design, distributed computing.

1 Introduction

Bluetooth [1] is a short range wireless RF technology designed initially for cable replacement at indoor places, but also supports usage scenarios for personal area or local area networking. Its low cost, low power consumption, and ad hoc connectivity features make it a good wireless connectivity choice for mobile devices due to their anytime-anywhere connection requirement and limited battery power.

Bluetooth devices are able to form small networks, which are called piconets, with up to eight nodes. A piconet consists of a master node and one or more slave nodes. The scheduling of packets into the common FHSS radio channel of a piconet is coordinated by the master node via a polling based TDMA scheme. No direct communication between any two slave nodes is allowed within a piconet. Data traffic among slave nodes has to go through the master node.

Bluetooth standards also specify some set of mechanisms to construct larger networks called scatternets. A scatternet is actually nothing but a network consisting of multiple piconets with common nodes between them, called bridges, to forward traffic between those piconets. Since all members of a piconet follow the same pseudorandom hopping sequence over some set of predefined RF channels specified in Bluetooth standards, and since each piconet has a different hopping sequence, a bridge node follows the hopping sequence of different piconets, which it belongs to at different times.

Although piconet construction is a well-defined process described in current Bluetooth standards, scatternet formation algorithms and policies are not specified with enough detail. Therefore, this is an open research area and many methods have been proposed so far for the scatternet formation problem. Once the scatternet is constructed, maintaining it in case of new node arrivals and node departures is another issue, which is not addressed much in the literature. Additionally, despite numerous works on scatternet formation, there is only a few studies ([2], [3], and [4]) that focus on constructing a scatternet with the aim of having the bandwidth capacity of the resulting scatternet utilized as efficiently as possible considering the traffic demands of the nodes constituting the scatternet. These studies all propose static (i.e., do not handle new node arrivals and node departures) and centralized solutions.

In this paper, we provide an algorithm that dynamically constructs a scatternet and concurrently modifies it to make it more bandwidth-efficient. In order to reduce the traffic load on a scatternet and thereby effectively utilize the capacity of the scatternet, our approach is based on reducing the path lengths between highly communicating nodes, as the approach followed in [4]. Different from [4], however, our algorithm works in a distributed manner with no central coordination and without requiring the availability of global topology and traffic demand information in advance. Furthermore, it is dynamic (i.e., new node arrivals and node departures are handled appropriately) and it can adapt to changes in the number of traffic flows and their rates, consequently preserving bandwidth-efficiency. Moreover,

- *M. Tekkalmaz is with ASELSAN A.S. MST-YMM-REH, Mehmet Akif Ersoy mh., 16. cad. No:16, Yenimahalle, Ankara, Turkey. E-mail: metint@cs.bilkent.edu.tr.*
- *H. Sözer is with the Software Engineering Research Group, University of Twente, Faculty of Electrical Engineering, Mathematics, and Informatics, PO Box 217 7500 AE, Enschede, The Netherlands. E-mail: sozerh@ewi.utwente.nl.*
- *I. Korpeoglu is with the Department of Computer Engineering, Bilkent University, TR-06800, Ankara, Turkey. E-mail: korpe@cs.bilkent.edu.tr.*

Manuscript received 15 Feb. 2005; revised 31 May 2005; accepted 2 Sept. 2005; published online 26 July 2006.
Recommended for acceptance by I. Stojmenovic, S. Olariu, and D. Simplot-Ryl.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDSSI-0128-0205.

1045-9219/06/$20.00 © 2006 IEEE Published by the IEEE Computer Society

messaging overhead imposed by the algorithm is kept as low as possible.

We evaluated our algorithm by way of simulation experiments to observe how it affects the efficient use of bandwidth in a scatternet. For the evaluation, we used a metric called *weighted average shortest path* (WASP) which is introduced in [4] and that indicates the amount of bandwidth usage in a scatternet for a given traffic demand. Our simulation results show that the use of our algorithm in a scatternet results with up to around 45 percent improvement in the value of the WASP metric, implying a significant improvement in the bandwidth-efficiency in a scatternet. We also investigated how energy consumption in a scatternet is affected by our algorithm. Besides resulting with more available bandwidth and less delay in communication, which are immediate consequences of efficient bandwidth usage, the use of our algorithm also causes reduction in the total energy spent in a scatternet and the energy spent in the individual nodes. Hence, our algorithm also constructs and maintains a scatternet in an energy-efficient way.

The remainder of the paper is organized as follows: In the next section, related previous studies are summarized. In Section 3, a scatternet construction algorithm is described. In Section 4, evaluation criteria for bandwidth efficiency are explained and simulation results, based on these criteria, are presented. Finally, some future work issues are discussed and the paper is concluded in Section 5.

2 Related Work

There have been many solution proposals for the Bluetooth scatternet formation problem [5]. Most of these proposals focus on the efficiency of the scatternet formation algorithm, considering the duration of the construction process and number of messages exchanged during the construction, as it is the case in [6], [7], and [8]. Some of them also apply heuristics in order to make the constructed scatternet efficient in terms of some metrics. One such heuristic that is followed by some studies is to keep the number of piconets as small as possible in the resulting scatternet with the goal of decreasing the amount of interpiconet traffic and interference. There exist other studies, which focus on different aspects of the problem. In [9] and [10], for instance, tree-shaped topologies are formed for the ease of routing. Some studies, like [11], on the other hand, aim to form fault-tolerant topologies by means of constructing alternative paths between nodes.

Although various proposals have been made to form scatternets with various objectives, algorithms that favor the efficient usage of bandwidth are not widely studied. In [12], as one of the earliest studies, traffic patterns among the piconet members is taken into consideration while selecting the master of the piconet. Some studies on constructing efficient scatternet topologies are [2], [3], and [4]. All these three proposals are single-hop, centralized, and static solutions. Baate et al. [2] propose a graph theoretical solution which uses 1-factors to construct a scatternet. Marsan et al. [3] approach to the problem as a min-max optimization. Topal [4], on the other hand, uses a set of heuristics to designate the locations and the roles of the nodes, assuming that the traffic flow information is known a priori.

There are also proposals for the scatternet formation problem that are distributed and dynamic. They are dynamic in the sense that they can handle node arrivals and departures. However, they do not consider traffic dynamics and trigger maintenance procedures when traffic conditions change. In [13], where a distributed and multi-hop algorithm is provided, the focus is on reducing the scatternet formation time, and a scatternet is not maintained based on traffic conditions. Also, the scatternet topology has to be a tree, which is not a requirement in our solution. Chiasserini et al. [14] propose a distributed and dynamic algorithm that resembles our work in considering traffic conditions for re-arranging a scatternet. However, the goal is not increasing the bandwidth efficiency, but reducing the load on the most congested node. It also provides an optimal topology in that respect but achieves this through a centralized algorithm. As another difference, we base our solution on an estimation of traffic load at a finer granularity, whereas [14] assumes that the traffic load on a node is classified as either low or high.

Along with the scatternet formation proposals which consider the relation between topology and efficiency, there are studies that try to reveal the relation between them, such as [15], [16], and [17]. Kapoor et al. [16] apply an analytical approach, whereas Miorandi et al. [17] try to establish a mathematical foundation to determine the relations between the topology and network capacity. On the other hand, Miklos et al. [15] apply a statistical approach in order to investigate the effects of topology on performance.

3 The Construction Algorithm

We now describe, in this section, our distributed solution for bandwidth and energy efficient scatternet construction and maintenance. Our solution requires two main procedures, *Link Establishment* and *Maintenance* procedures, to be executed *concurrently* in a scatternet. The *Link Establishment* procedure handles new node arrivals (i.e., new connections) and node departures (i.e., disconnections), hence, incrementally constructing a scatternet. On the other hand, the *Maintenance* procedure works on the base topology constructed by the *Link Establishment* procedure and maintains the bandwidth-efficiency of the scatternet by modifying it.

The *Maintenance* procedure constantly collects information about the traffic flowing in the scatternet and runs a set of operations based on this information when *significant* changes in the traffic pattern and rates are observed. These operations are performed in order to modify the scatternet topology so that the node pairs whose communication demands between each other are relatively higher are placed closer to each other in the scatternet.

The *Maintenance* procedure depends on some assumptions. The first assumption is that all the nodes which are going to be part of the scatternet are in the communication range of each other. Another assumption is that the topology on which the *Maintenance* procedure works has the following properties: