

Bluetooth Scatternet Formation for Single-hop Ad Hoc Networks Based on Virtual Positions

Yu Wang[†], Ivan Stojmenovic^{*}, Xiang-Yang Li[#]

[†] Department of Computer Science
University of North Carolina at Charlotte
Charlotte, North Carolina, USA

^{*} School of Information and Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada

[#] Department of Computer Science
Illinois Institute of Technology
Chicago, Illinois, USA

ywang32@uncc.edu, ivan@site.uottawa.ca, xli@cs.iit.edu

Abstract

This paper addresses the problem of scatternet formation for single-hop Bluetooth based personal area and ad hoc networks, with minimal communication overhead. Recent scatternet formation schemes by Li, Stojmenovic and Wang [1] 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 the number of slave roles in one piconet to 7. In this article we 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 we use Delaunay triangulation instead of partial Delaunay triangulation proposed in [1], since each node has all the information needed. Likewise, we can also apply minimum spanning tree (MST) as the planar topology in our new schemes. Finally, we design experiments to study both the properties of formatted scatternets (such as number of piconets) 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.

Keywords: System design, graph theory, Bluetooth networks, scatternet formation, single-hop.

1 Introduction

The rapid adoption of the Internet and mobile wireless technologies is paving the way for high bandwidth to the mobile terminal. Local and personal area networks are also

increasingly becoming wireless, incorporated into seamless all IP wireless and mobile networks. Ad-hoc enabled consumer products will begin to form small-scale ad-hoc networks between a small group of people/devices. Communication between the devices (called nodes hereafter) in the ad-hoc network can be single hops or multiple hops. Bluetooth is well suited to provide ad-hoc networking for the consumer market. Bluetooth ad-hoc networking presents some technical challenges, such as scheduling, network forming and routing. User mobility poses additional challenges for connection rerouting and QoS services. It has been widely predicted that Bluetooth will be the major technology for short-range wireless networks and wireless personal area networks. This paper deals with the problem of building ad hoc networks using Bluetooth technology.

Bluetooth is an open specification for short-range wireless communication and networking, mainly intended to be a cable replacement between portable and/or fixed electronic devices. According to the standard, when two Bluetooth devices come into each other's communication range, one of them assumes the role of *master* of the communication and the other becomes the *slave*. This simple one hop network is called a *piconet*, and may include more slaves. The network topology resulted by the connection of piconets is called a *scatternet*. There is no limit on the maximum number of slaves connected to one master, although the number of active slaves at one time cannot exceed 7. If a master node has more than 7 slaves, some slaves must be parked. To communicate with a parked slave, a master has to *unpark* it, thus possibly parking another active slave instead. The standard also allows multiple roles for the same device. A node can be master in one piconet and a slave in one or more other piconets. However, one node can be active only in one piconet. To operate as a member of another piconet, a node has to switch its hopping frequency. Since each such switch causes delays (e.g., scheduling and

synchronization time), an efficient scatternet formation protocol can be one that minimizes the roles assigned to the nodes, without losing network connectivity.

While several solutions and commercial products have been introduced for one-hop Bluetooth communication, the problem of scatternet formation has not been dealt with until very recently. Several criteria could be set as the objectives in forming scatternet. First of all, the resulting network should be connected. Secondly, the number of piconets should be minimized to provide faster routing. Thirdly, the formation and maintenance of scatternet should have small communication overhead. Fourthly, the protocol should create degree-limited scatternets, to avoid parking any node.

In this paper, we focus on scatternet formation for single-hop ad hoc networks. In a single-hop ad hoc network, all wireless devices are in the radio vicinity of each other, e.g., electronic devices in a laboratory, or laptops in a conference room. A single-hop network can be modeled by a complete graph. Our scatternet formation solutions build or apply some geometric structures on the complete graph. We apply the same scheme recently proposed by [1] for multi-hop networks. In case of multi-hop networks, their scheme required that every node knows its *exact position* information. Obtaining the precise positions currently poses challenging technological tasks [2] for short-range Bluetooth devices, aimed primarily at home and office environments. However, we observe that, when the same scheme is applied to single-hop network, *virtual positions* (random position selected by each node independently and without any hardware requirements) are sufficient. The problem with virtual positions being applied in multi-hop networks is that two nodes which select virtual positions that are close to each other may physically be outside of each other's transmission range. On the other hand, in single-hop ad hoc networks, every node can communicate with each other directly, and the problem in multi-hop networks does not occur. Another advantage of using virtual positions for single-hop network is that our scatternet formation can be used for wireless nodes in three-dimensional space (such as a building) by just generating 2-dimensional virtual positions in a virtual plane.

Previous literature on scatternet formation assumed that devices are not able to communicate unless they have previously discovered each other by synchronizing their frequency hopping patterns. Synchronizing the frequency hopping patterns is apparently a time consuming and pseudo-random process [3]. In this paper we assume that the problem of discovering all neighbors within transmission radius of all neighbors is resolved by separate Bluetooth protocol such as those in [3,4].

The rest of the paper is organized as follows. In Section 2, we give preliminaries needed to describe our new

algorithms, and briefly review the literature on scatternet formation and related network topology design issues. Section 3 presents Bluetooth formation algorithms from [1], while Section 4 describes our new algorithms for single-hop ad hoc networks. The Bluetooth scatternet formation algorithm for single-hop networks limits the degree of each node to 7, keeps the connectivity of all the piconets, and does not park any node. Section 5 describes the experimental result of our algorithm. We conclude our paper in Section 6 by pointing out some possible future research directions.

A preliminary conference version of this article appeared in [5].

2 Preliminaries

In this section, we first give some geometry definitions and notations that will be used in our presentation later. We then briefly review some related results on Bluetooth scatternet formation.

2.1 Geometry Definitions and Notations

We assume that all wireless nodes are given as a set S of n vertices in a two-dimensional space. Each node has some computational power. For single-hop ad hoc networks, we assume that all nodes are in the transmission ranges of each other. We model a single-hop network as a complete graph $CG(S)$.

The *Gabriel graph* [6] $GG(S)$ contains edge uv if and only if the disk with diameter uv contains no other points of S . The relative *neighborhood graph* [7], denoted by $RNG(S)$, consists of all edges uv such that the intersection of two circles centered at u and v and with radius $\|uv\|$ does not contain any vertex from the set S . $GG(S)$ and $RNG(S)$ are planar graphs (that is, no two edges cross each other). Obviously, they can be constructed in a localized manner. In other words, a node u can compute its incident edges in $GG(S)$ or $RNG(S)$ by using only 1-hop neighbors. It is easy to show that $RNG(S)$ is a subgraph of $GG(S)$. Both $GG(S)$ and $RNG(S)$ are connected and contain the Euclidean minimum spanning tree of S .

The Yao graph [8] is proposed by Yao to construct MST of a set of points in high dimensions efficiently. At given node u , any k equal-separated rays originated at u define k cones. In each cone, choose the closest node v within the transmission range of u , if there is any, and add a directed link uv . Ties are broken arbitrarily. The remaining edges are deleted from the graph. There are several variants on how this construction can be carried at each node in the graph. One choice is to carry it simultaneously on each node, with two options about keeping an edge uv : keep only if they mutually selected each other, or keep directional edges as well (one node selected other but not vice versa). The other

choice (considered in this paper), is to carry this process sequentially, first at node u , and then at node v . In this case, if u did not select v , then edge uv is considered deleted by v and is ignored when v makes its decision afterward.

We continue with definition of the Delaunay triangulation. We assume that there are no four vertices of S that are co-circular. A triangulation of S is a *Delaunay triangulation*, denoted by $DT(S)$, if the circumcircle of each of its triangles does not contain any other vertices of S in its interior. Obviously, $GG(S)$ and $RNG(S)$ are subgraphs of $DT(S)$.

Sparse geometric structures that can be defined locally have been applied in wireless networks for localized routing and broadcasting algorithms. Gabriel graph was used in [9,10] in order to define planar subgraph used for recovery routing to guarantee delivery, when simple heuristics fail. Gabriel graph was replaced in [11] by newly proposed restricted Delaunay graph, consisting of all the Delaunay edges with length up to transmission radius, possibly with some additional edges. However, the construction process requires additional nontrivial communication between nodes when they move or change activity. Relative neighborhood graph was used in [12] to provide efficient localized broadcasting for one-to-one models of wireless communications. Li *et al.* [13] proposed to use GGs, RNGs, and Yao graphs to construct sparse power efficient networks. To improve the graph connectivity of planar graphs, Li *et al.* [14] and [1] then proposed another two planar structures, localized Delaunay triangulation (LDT) and partial Delaunay triangulation (PDT), which can be constructed locally and efficiently. Both LDT and PDT contain GG as their subgraph, and themselves are subgraphs of DT.

2.2 Literature Review

Scatternet formation algorithms proposed in literature can be divided into single-hop and multi-hop solutions. In this paper, we only focus on designing scatternet formation algorithms for single-hop networks.

In [15,16,17], authors proposed several algorithms that create connected degree-bounded scatternets. The final structures are all tree like topologies, which limit efficiency and robustness. Note that the tree topology suffers from a major drawback: the root is a communication bottleneck, as it will be overloaded by communications between the different parts of the tree. A single-hop Bluetooth scatternet formation scheme based on 1-factors is described in [18]. However, piconets are not degree-limited in that scheme. Recently, Salonidis *et al.* [3] proposed a centralized topology construction algorithm which is not scalable and not localized. Moreover, how to assign the roles is not elaborated in [3]. They also assume up to 36 nodes in the

network. Another centralized solution for single-hop networks, where the traffic between any pair of nodes is known a priori, is described in [19]. Sun *et al.* [20] described a self-routing topology for single-hop Bluetooth networks. Nodes are organized and maintained in a search tree structure, with Bluetooth ID's as keys. It relies on a sophisticated scatternet merge procedure with significant communication overhead for creation and maintenance. Bluerings as scatternets are proposed in [21]. Ring structure for Bluetooth has simplicity and easy creation as advantage, but it suffers large diameter (i.e., the maximum number of hops between any two devices) and large number of piconets. Barriere *et al.* [22] described a connected degree limited and distributed scatternet formation solution based on projective geometry for single-hop networks. However, in their method, every node need hold information of the projective plane and the master node with "token" needs to know the information of the projective scatternet. In [22], they did not discuss in detail how to compute the labels for the new master and its slaves, and what will happen when the number of nodes reaches the number of nodes of a complete projective scatternets.

3 Scatternet Formation Algorithms [1]

We now review the localized scatternet formation algorithms from [1], based on sparse geometrical structures. The algorithms have several phases that are shown in following algorithm.

Algorithm 1: Scatternet Formation Algorithms

- 1) Neighbor discovery and information exchange.
- 2) Planar subgraph construction (constructing RNG, GG, or PDT), if desirable.
- 3) Degree information exchange, if desirable.
- 4) Bounding degree and assigning roles (consisting of several iterations).

Initially all nodes are undecided. In each iteration, if an undecided node has the highest degree among its all undecided neighbors, it runs the following steps:

- a) Bounds its degree (applying Yao structure).
- b) Assigns role to itself (based on the information on each link or using cluster-based method).
- c) Marks itself decided, and notices the deleted edges and its status to its undecided neighbors.

Repeat the iterations, until all nodes are decided.

3.1 Neighbor Discovery and Information Exchange

Firstly, in the neighbor discovery phase, each node learns about its one-hop or two-hop neighbors. This

procedure is called *inquiry procedure* in Bluetooth specifications. One-hop neighbor discovery can be performed by a scheme described in [3,4].

3.2 Planar Subgraph Construction

This phase is optional. The remaining phases can be applied on the complete graph directly, but will result in non-planar graph. Planarity may be a desirable property in some cases, e.g., routing with guaranteed delivery. In this phase, each node computes which of its incident edges belongs to chosen planar sparse structure, RNG, GG, or PDT. Note that each node can make local decisions about each of its edges without any message exchange.

3.3 Degree Information Exchange

This phase is also optional. In our methods, master-slave relations are decided based on a key. Two different keys can be considered. If node's Bluetooth ID is used as a key, this phase can be omitted. If the key is selected as the record (*degree, ID*), where node degree is the primary key, we need to collect degree information from neighbors. The procedure is basically the same procedure needed to collect two-hop information. One such Bluetooth compatible procedure has been described in [23] and is applicable here.

3.4 Bounding Degree and Assigning Roles

In the next (mandatory) phase, the degree of each node is limited to 7 by applying Yao structure, and the master-slave relations are formed in created subgraphs. Each node applies Yao structure on all of its neighbors, where $k=7$. This will guarantee that the number of slaves assigned to any node is no more than 7. To simplify the explanation, we assume that Yao construction is applied to all nodes (each at appropriate iteration), even if it has less than 7 neighbors. An edge remains in the structure if and only if both endpoints selected it, otherwise it is deleted from the structure. The process of applying Yao structure is done in an *iterative* way. It works as follows.

Initially all nodes are undecided. In each iteration, undecided nodes with higher keys than any of their undecided neighbors (we shall refer to such nodes as *active nodes* in the sequel) apply Yao structure to limit the degree, decide master-slave relations on the remaining edges, and inform all neighbors about either deleting edge or master-slave decision. The active node then switches to a decided state. Note that, in order to avoid excessive information exchange between neighbors, the originally decided keys (that is, original degrees) are used in all comparisons. At the end of each iteration, an information exchange step is needed so that active nodes inform their neighbors in the applied structure

about its decisions. For eliminated edges, the other endpoint node is informed about the decision, and that node then deletes that edge from its own list. For the selected edge, active node makes master-slave decision for the edge and informs the other node on each edge about the decision. This information exchange step is very similar to the one-hop neighbor discovery phase. The difference is that communication can be restricted to edges remaining in the graph, so that the information exchange step is faster than neighbor discovery phase.

In each iteration, active nodes decide master-slave roles at each undeleted edge, and communicate the decision to the other node at each edge. We shall now describe two different ways to decide the roles: node with initially higher key is master, and cluster based. Both methods keep all links "saved" by Yao structure in the final Bluetooth topology but converts them to directed edges, so that one node on each edge is master node, and the other is slave node.

The first method assigns roles based on the information on each link. Two neighboring nodes u and v compare their keys (ID or $(degree, ID)$), and the one with higher key becomes the master node, and the other node is the slave node. The purpose of such role assignment is to avoid slave roles at high connectivity nodes.

In the cluster based approach, a dominating set of masters in the degree limited subgraph is constructed, and a piconet is added for each remaining edge between two nodes not selected in dominating set, to preserve connectivity. In a given iteration, there are three cases for an active node u to assign role: 1) u decides to serve as the master node if it has only master role or is unassigned. It notices its undecided neighbors to add a slave role. 2) If u has previously received only slave roles, it decides to serve as a slave on its remaining links. Thus, it notices all remaining undecided neighboring nodes to add a master role. 3) If u has previously been given both master and slave roles, it keeps master-slave roles and notices all its remaining undecided neighboring nodes to add a slave role on the link to u . Notice that each active node marks itself decided after the above operation. Also each node, when receiving a notice of adding role, will change its role correspondingly. For example, if a slave node receives a notice of adding a master role, it will change its role to a masterslave node. In next section, we will show an example with the detailed iterations of assigning roles.

In [1], Li *et al.* proved that the scatternet formed is a connected sparse subgraph such that each node has degree at most 7 after the iterative application of Yao structure and assigning roles. In addition, the constructed topology may be a planar graph, if we decide so, which makes possible to

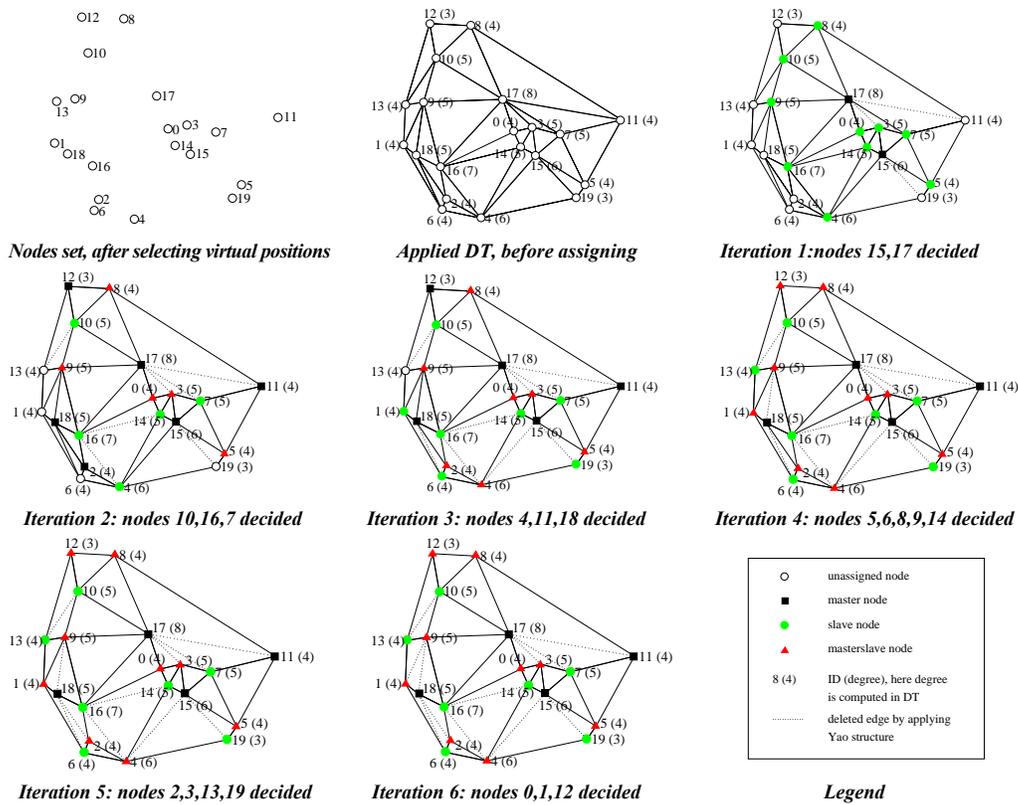


Figure 1 Assigning roles in single-hop networks: six iterations

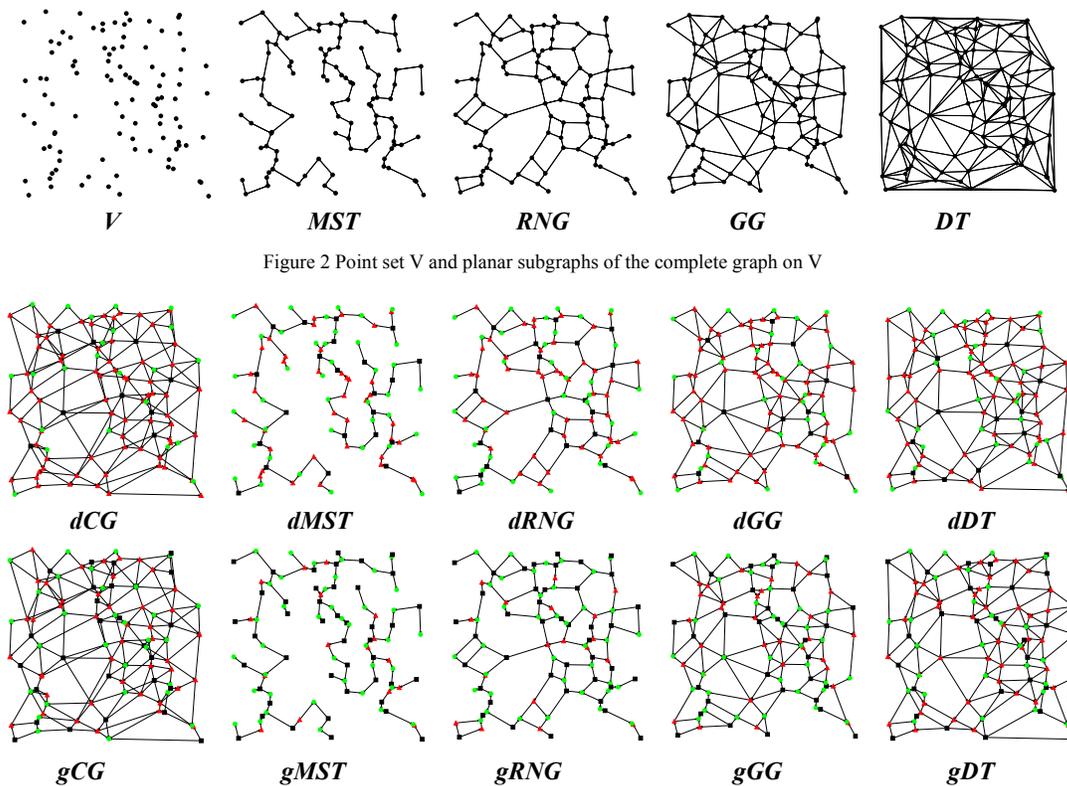


Figure 3 Geometric structures, bounding node degree, and assigning roles

implement some geometry-position based routing algorithms [10]. Recently, Basagni *et al.* [24] described some results of a ns2-based performance evaluation of our multi-hop scatternet formation method.

4 Bluetooth Scatternet Formation for Single-hop Networks

In this paper, we adopt our multi-hop scatternet formation algorithms to single-hop ad hoc networks. Recall that, in a single-hop ad hoc network, all wireless devices are in the radio vicinity of each other, e.g., electronic devices in a laboratory, or laptops in a conference room. A single-hop network can be modeled by a complete graph. Our new scatternet creation solutions for single-hop networks apply the same schemes we described in last section for multi-hop networks. However, with the nice property of single-hop networks (each node knows all other nodes information), we can avoid the use of positions information in our algorithms. Also we can use some planar subgraphs of the complete graph, such as MST or DT, which can not be constructed locally in multi-hop networks.

4.1 Virtual Positions

In case of multi-hop networks, these schemes require *exact position* information. Obtaining the precise positions currently poses challenging technological tasks [2] for short-range Bluetooth devices. However, when the same schemes are applied to single-hop network, *virtual positions* (random position selected by each node independently and without any hardware requirements) are sufficient. The problem with virtual positions being applied in multi-hop networks is that two nodes which select virtual positions that are close to each other may physically be outside of each other's transmission range. However, in single-hop ad hoc networks, every node can communicate with each other directly, and the problem in multi-hop networks does not occur. Another advantage of using virtual positions for single-hop networks is that our scatternet formation can be used for wireless nodes in three-dimensional space (such as a building) by simply generating 2-dimensional virtual positions in a virtual plane. Thus, in our new scatternet creation solutions for single-hop networks, each node selects independently a random position in the neighbor discovery and information exchange phase.

4.2 Planar Subgraphs

In the second (planar subgraph construction) phase, following methods [1] proposed for multi-hop networks, a planar subgraph is constructed locally and efficiently. Since in single-hop networks every node has all the information

needed, we can apply some well-known global planar substructures in the second phase. For example, we can construct Delaunay triangulation (DT) directly instead of PDT. Notice that PDT is a subgraph of DT and DT is much denser than PDT. Thus, using DT may increase the delivery rate of routing methods on it. In addition, we can also use minimum spanning tree (MST) as the planar topology, which is suitable for broadcasting application. In summary, in the second phase, each node computes which of its incident edges belongs to chosen planar sparse structure, MST, RNG, GG, or DT. Note that each node can make local decisions (after completing neighbor discovery phase). Thus this construction only takes some computation cost without any communication cost.

4.3 An Example

Figure 1 illustrates the procedure of our algorithms (applying planar structure, bounding degree, and assigning roles) in detailed iterations for an example single-hop network.

5 Experiments

In this section, we present our experimental results that compare designed algorithms in terms of various characteristics. We did not include other existing schemes for single-hop networks since each of them has deficiencies (with respect to our scheme) such as significant maintenance overhead, possible disconnection, possible excess number of roles, or lack of termination proof. In the experimental results presented here, we choose total $n=100$ wireless nodes which are distributed randomly in a square area. Each node is specified by random X and Y coordinate values. These coordinates are virtual, used to mimic geographic position needed to establish Bluetooth scatternet structure. All results are the averages on total 20 wireless nodes sets.

5.1 Bluetooth Scatternet Formation

All nodes can be divided into several categories, according to the type and number of roles taken in the process. Thus a given node can be: (1) master only, denoted by M ; (2) slave only, denoted by S , possibly to a few piconets, this can be further divided as S_p , where p is the number of piconets where this slave node serves; (3) master of one piconet and slave in other piconets, denoted by MS or in general MS_p , where p is the number of piconets in which this node serves as slave.

Figure 3 illustrates the different Bluetooth structures using CG, MST, RNG, GG, or DT as topologies (shown in Figure 2), bounding degree by applying Yao structure, and assigning node roles by comparing end-nodes degrees of each link (denoted by d^*) or using cluster based method

Table 1 Number of master(M) nodes and number of slave(S)/master-slave(MS) nodes with p masters

graph	M	S_1 / MS_1	S_2 / MS_2	S_3 / MS_3	S_4 / MS_4	S_5 / MS_5	S_6 / MS_6	S_7 / MS_7	$S_{>7} / MS_{>7}$
dCG	9.40	0.45/15.9	3.70/24.2	10.8/20.7	7.35/5.45	1.60/0.40	0.00/0.00	0.00/0.00	0.00/0.00
gCG	22.9	0.00/15.8	1.45/16.4	5.70/10.3	7.85/5.60	6.70/2.45	3.30/0.50	0.95/0.00	0.00/0.00
dMST	22.3	23.7/32.7	20.3/0.95	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
gMST	46.8	7.05/10.3	23.6/2.45	9.20/0.00	0.45/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
dRNG	21.9	8.25/30.6	28.9/8.10	2.15/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
gRNG	41.7	2.30/13.9	19.3/4.65	16.5/0.45	1.05/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
dGG	13.3	2.10/27.1	13.7/24.6	13.2/4.95	0.90/0.05	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
gGG	32.5	0.60/17.0	7.25/11.4	13.9/4.15	10.4/0.50	2.10/0.00	0.15/0.00	0.00/0.00	0.00/0.00
dDT	11.4	0.45/19.7	5.95/27.3	14.1/14.0	5.55/1.15	0.25/0.00	0.00/0.00	0.00/0.00	0.00/0.00
gDT	26.4	0.00/16.5	2.55/16.0	8.50/8.75	10.5/3.20	5.70/0.15	1.50/0.05	0.05/0.00	0.00/0.00

Table 2 Number of piconets, bridge nodes, and size of piconets

graph	master	slave	masterslave	avg M of S	avg M of MS	avg S of (M+MS)
dCG/gCG	9.40/22.95	23.90/25.95	66.70/51.10	3.25/4.29	2.25/2.29	3.00/3.08
dMST/gMST	22.30/46.85	44.05/40.35	33.65/12.80	1.46/2.08	1.03/1.20	1.77/1.66
dRNG/gRNG	21.90/41.70	39.35/39.25	38.75/19.05	1.85/2.42	1.21/1.29	1.97/1.97
dGG/gGG	13.30/32.50	29.90/34.45	56.80/33.05	2.43/2.97	1.61/1.95	2.34/2.71
dDT/gDT	11.40/26.40	26.35/28.85	62.25/44.75	2.97/3.84	1.95/1.99	2.71/2.80

(denoted by g^*). The master and master-slave nodes are denoted by black squares and red triangles respectively, while the slaver nodes are denoted by green disks.

Table 1 lists the number of slave/master-slave nodes that serve as slaves of p piconets under different Bluetooth topologies. We conducted extensive simulations using different number of nodes (from 20 to 500). We find that the results are stable, i.e., the portion of the bridge nodes is stable. In addition, as we expected, the cluster based method generates smaller number of nodes with masterslave roles than the method comparing degrees of two end-points of a link.

Table 2 presents the average number of slave nodes assigned to a node with master role, i.e., a master node or a master-slave node. The fifth column represents the average number of piconets assigned to a node with slave roles only. The sixth column represents the average number of piconets assigned to a node with both master and slave roles. We found that assigning node roles based on the cluster based approach always produces fewer number of slaves to a node with master role. Moreover, it also generates less number of nodes with master-slave role than the other methods.

We found that the complete graph CG consistently performs the worst among all underlying structures: it has

less pure master node, has many slave nodes belonging to many piconets. The other structures (MST, GG, RNG, DT) perform at the same level in terms of the number of piconets generated and the number of piconets a slave node belonging to. We suggest to use DT since it has more edges than other three structures, thus, can sustain more link failures, and have shorter path for some pair of nodes. We also found that scatternets generated based on GG and DT are similar, due to the fact that DT has slightly more edges than GG.

5.2 Routing in Scatternet

Routing in Bluetooth received little attention so far. Bhagwat and Segall [25] proposed a routing method in Bluetooth based on a concept of route vector. They described protocols for route discovery and packet forwarding. Prabhu and Chockalingam [26] proposed battery power level based master-slave switch, distance based power control, and selecting route path with maximum cumulative battery power. Barriere et al. [22] also proposed a routing method for Bluetooth scatternets formatted by their method using their specific labels. An important problem for scatternet formation algorithms is to choose the structure that also provides efficient routing on the designed scatternet, in terms of hop count, power consumption, and delay in message delivery (the delay

depends on the amount of multiple roles performed by various nodes). Most designed structures are planar and therefore suitable for routing with guaranteed delivery [10], which is an additional benefit of proposed structures.

Table 3 Delivery rate

	sCG	sMST	sRNG	sGG	sDT
NN	83.8	10.5	33.8	63.3	80.3
FN	80.00	8.8	21.3	72.2	76.7
MFR	79.7	19.3	53.4	88.5	90.3
Cmp	76.6	4.2	18.9	46.0	65.5
RCmp	92.8	15.8	31.9	65.4	81.0
Grdy	100.0	31.3	68.8	100.0	100.0
GCmp	85.2	5.5	22.9	53.3	66.7

In this subsection, we study some well-known geometric localized routing methods on the new structures. Assume a packet is currently at node u , and the destination node is t . Several localized routing algorithms, i.e., find the next node v of u based on t and information of k -hop neighbors of node u , were developed. In our experiments, we test the following routing algorithms: compass routing (Cmp)[27], random compass routing (RCmp)[27], greedy routing (Grdy)[10], most forwarding routing (MFR)[28], nearest neighbor routing (NN), farthest neighbor routing (FN), greedy-compass (GCmp)[29,30]. The compass routing, random compass routing and the greedy routing guarantee to deliver the packets if DT is used as network topology [10,27,30].

In our experiments, again we choose 100 nodes distributed randomly in a square area. Figure 2 and Figure 3 illustrate the well known planar topologies and the final topologies after applying our method. We randomly select 20% of nodes as source; and for each source, we randomly choose 20% of nodes as destination. The statistics are computed over 10 different node sets.

Table 3 illustrates the delivery rates. We use s^* to denote the bounded degree structures after applying Yao structure, where $*$ denotes the name of the sparse topology from the second phase. For routing methods NN and FN, we choose the next node within $\pi/3$ of the destination direction. Because sDT is denser than sMST, sGG and sRNG, the delivery rates of many routing methods on it are higher. Recall that sCG is not a planar structure, while other three are. Since sMST, sGG and sRNG are planar graphs, we can apply right hand rule to improve delivery rate. More precisely, delivery can be even guaranteed following method described in [10], which applies the greedy routing on Gabriel graph and uses the right hand rule for recovery when greedy mode fails.

6 Conclusion

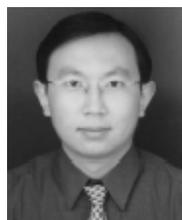
We have described a scheme that creates connected degree limited scatternets for single-hop Bluetooth networks. A number of issues remain for future study. One of major desirable properties of the proposed cluster based method is that the number of masters that serve as slaves in other piconets is minimized, in fact limited to gateway piconets. However, this property is not without a cost. The problem with clustering approach in multi-hop networks is that the maintenance of clustered graph structure is expensive, since a local change due to mobility may trigger global change in updating the scatternet. In single-hop networks, we assume that nodes remain within transmission range of each other, therefore cluster update procedure is not called due to mobility. Nevertheless cluster maintenance is needed when nodes are added or removed from the network. Cluster update scheme can be modified to achieve localized maintenance property, but at a significant cost of increasing the number of clusters. To address this problem, and still reduce the number of piconets, which is the main problem with the first proposed method here (where higher degree node on any remaining link is the master node), we intend to study alternative way of determining master-slave relations. Some other interesting problems include: fast schemes for the neighbors discovery, more suitable routing algorithms for the proposed scatternets, scheduling of Bluetooth piconets, and capacity assignment based on expected traffic load.

References

- [1] X.Y. Li, I. Stojmenovic and Y. Wang, Partial Delaunay Triangulation and Degree Limited Localized Bluetooth Multihop Scatternet Formation, *IEEE Transactions on Parallel and Distributed Systems*, Vol.15, No.4, 2004, pp. 350-361.
- [2] J. Hightower and G. Borriello, Location Systems for Ubiquitous Computing, *IEEE Computer*, Vol.34, No.8, 2001, pp. 57-66.
- [3] T. Salonidis, P. Bhagwat, L. Tassiulas and R. LaMaire, Distributed Topology Construction of Bluetooth Personal Area Networks, *Proc. IEEE INFOCOM*, 2001.
- [4] S. Basagni, R. Bruno and C. Petrioli, Device Discovery in Bluetooth Networks: A Scatternet Perspective, *Proc. IFIP-TC6 Networking Conf., Networking 2002*, Italy, May 2002, LNCS 2345.
- [5] Y. Wang, I. Stojmenovic and X.Y. Li, Bluetooth Scatternet Formation for Single-hop Ad Hoc Networks Based on Virtual Positions, *Proc. 9th IEEE Symp. on Computers & Communications*, Alexandria, Egypt, Jun. 2004.
- [6] K.R. Gabriel and R.R. Sokal, A New Statistical Approach to Geographic Variation Analysis, *Systematic Zoology*, Vol.18, 1969, pp. 259-278.
- [7] G. Toussaint, The Relative Neighborhood Graph of a Finite Planar Set, *Pattern Recognition*, Vol.12, No.4, 1980, pp. 261-268.
- [8] A.C.C. Yao, On Constructing Minimum Spanning Trees in k -dimensional Spaces and Related Problems, *SIAM J. Computing*, Vol.11, 1982, pp. 721-736.
- [9] L. Barriere, P. Fraigniaud, L. Narayanan and J. Opatrny, Robust Position Based Routing in Wireless Ad Hoc Networks with Unstable Transmission Ranges, *Proc. ACM DIALM01*, Rome, Italy, July 2001, pp. 19-27.
- [10] P. Bose, P. Morin, I. Stojmenovic and J. Urrutia, Routing with Guaranteed Delivery in Ad Hoc Wireless Networks, *ACM/Kluwer Wireless Networks*, 7, 6, November 2001, pp. 609-616.
- [11] J. Gao, L.J. Guibas, J. Hershberger, L. Zhang and A. Zhu, Geometric Spanner for Routing in Mobile Networks, *Proc. ACM MobiHoc*, 2001.
- [12] M. Seddigh, J. Solano and I. Stojmenovic, RNG and Internal Node Based Broadcasting Algorithms for Wireless One-to-one Networks, *ACM Mobile Computing and Communications Review*, Vol. 5, No. 2, 2001, pp. 37-44.
- [13] X.Y. Li, P.J. Wan, Y. Wang and O. Frieder, Sparse Power Efficient Topology for Wireless Networks, *Proc. Hawaii Int. Conf. on System Sciences*, 2002.
- [14] X.Y. Li, G. Calinescu, P.J. Wan and Y. Wang, Localized Delaunay Triangulation with Application in Ad Hoc Wireless Networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 14, No. 10, 2003, pp. 1035-1047.
- [15] G.V. Zaruba, S. Basagni and I. Chlamtac, Bluetrees - Scatternet Formation to Enable Bluetooth Based Ad Hoc Networks, *Proc. IEEE ICC*, 2001.
- [16] G. Tan, A. Miu, J. Guttag and H. Balakrishnan, Forming Scatternets from Bluetooth Personal Area Networks, *First Annual Student Oxygen Workshop*, Gloucester, MA, July 2001.
- [17] C. Law, A.K. Mehta and K.Y. Siu, Performance of a new Bluetooth Scatternet Formation Protocol, *Proc. ACM MobiHoc01*, October 2001, pp. 183-192.
- [18] S. Baatz, S. Bieschke, M. Frank, P. Martini, C. Scholz and C. Kuhl, Building Efficient Bluetooth Scatternet Topologies from 1-Factors, *Proc. IASTED Wireless and Optical Communications*, Banff, Canada, July 2002.
- [19] D. Miorandi and A. Zanella, On The Optimal Topology of Bluetooth Piconets: Roles Swapping Algorithms, *Proc. Mediterranean Conf. on Ad Hoc Networks*, Sardinia, Italy, Sept. 2002.
- [20] M.T. Sun, C.K. Chang and T.H. Lai, A Self-routing Topology for Bluetooth Scatternets, *Proc. ISPAN*, Philipini, May 2002.
- [21] F. Cgun-Choong and C. Kee-Chaing, Bluerings - Bluetooth Scatternets with Ring Structure, *Proc. IASTED Wireless and Optical Communications*, Banff, Canada, July 2002.
- [22] L. Barriere, P. Fraigniaud, L. Narayanan and J. Opatrny, Dynamic Construction of Bluetooth Scatternets of Fixed Degree and Low Diameter, *Proc. ACM SODA*, 2003.
- [23] C. Petrioli and S. Basagni, Degree-constrained Multihop Scatternet Formation for Bluetooth Networks, *Proc. IEEE GLOBECOM 2002*, Taipei, Taiwan, November 2002.
- [24] S. Basagni, R. Bruno and C. Petrioli, A Performance Comparison of Scatternet Formation Protocols for Networks of Bluetooth Devices, *Proc. IEEE PerCom*, Texas, Mar. 2003.
- [25] P. Bhagwat and A. Segall, A Routing Vector (RVM) for Routing in Bluetooth Scatternets, *Proc. IEEE Int. Workshop on Mobile Multimedia Communications MoMuC*, Nov. 1999.
- [26] B.J. Prabhu and A. Chockalingam, A routing Protocol and Energy Efficient Techniques in Bluetooth Scatternets, *Proc. IEEE ICC* 2002.
- [27] E. Kranakis, H. Singh, and J. Urrutia, Compass Routing on Geometric Networks, *Proc. 11th Canadian Conf. on Computational Geometry*, 1999, pp. 51-54.

- [28]I. Stojmenovic and X. Lin, Loop-free Hybrid Single-path/flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 12, No. 10, 2001.
- [29]P. Bose and P. Morin, Online Routing in Triangulations, *Proc. of 10th Annual Int. Symp. on Algorithms and Computation ISAAC*, 1999.
- [30]P. Morin, Online Routing in Geometric Graphs, Ph.D. Thesis, Carleton University, 2001.

Biographies



Yu Wang received the PhD degree in computer science from Illinois Institute of Technology in 2004, the BEng degree and the MEng degree in computer science from Tsinghua University, China, in 1998 and 2000. He has been an assistant professor of computer science at the University of North Carolina at Charlotte since 2004. His current research interests include wireless networks, mobile computing, algorithm design, and artificial intelligence. He is a member of the ACM, IEEE, and IEEE Communication Society.



Ivan Stojmenovic received Ph.D. degree in mathematics. He is Editor-in-Chief of *Ad Hoc & Sensor Wireless Networks*, *International Journal of Parallel, Emergent and Distributed Systems*, and *Multiple-Valued Logic and Soft Computing*, and editor of several journals including *IEEE TPDS*. He guest edited special issues in several journals including *IEEE Computer Magazine* (February 2004), *IEEE Networks* (July 2004), and *Wireless Communications and Mobile Computing* (Wiley). His current research interests include wireless ad hoc, sensor and cellular networks. He received Fast Breaking Paper in Computer Science for October 2003, Faculty of Engineering's 2004-2005 Award for Excellence in Research at the University of Ottawa, and other awards. He edited 'Handbook of Wireless Networks and Mobile Computing' (Wiley, 2002), 'Mobile Ad Hoc Networking' (IEEE/Wiley, 2004), and 'Handbook of Sensor Networks' (Wiley, 2005). He served as program co-chair at the IEEE MASS 2004, WONS 2005, workshop co-chair at IEEE ICDCS 2002-2005; HICSS 2000, 2002, 2003; ICPDS 2002; ICPP 2000 etc., and program committee member at a number of conferences (e.g. about 30 in 2005).



Xiang-Yang Li received the MS (2000) and PhD (2001) degrees in computer science from the University of Illinois at Urbana-Champaign. He received the Bachelors degree in computer science and business management from Tsinghua University, People's Republic of China, in 1995. He has been an assistant professor of computer science at the Illinois Institute of Technology since 2000. His research interests span the computational geometry, wireless ad hoc networks, optical networks, and cryptography. He is a member of the ACM and the IEEE.