

HMT: A Hybrid Mesh Tree Algorithm in Forming Bluetooth Networks

Chih-Min YU^{†a)}, Member

SUMMARY In this letter, a new scatternet formation algorithm called hybrid mesh tree for Bluetooth ad hoc networks was proposed. The hybrid mesh tree constructs a mesh-shaped topology in one dense area that is extended by tree-shaped topology to the other areas. First, the hybrid mesh tree uses a designated root to construct a tree-shaped subnet, and then propagates a constant k in its downstream direction to determine new roots. Each new root then asks its upstream master to start a return connection procedure to convert the first tree-shaped subnet into a mesh-shaped subnet. At the same time, each new root repeats the same procedure as the designated root to build its own tree-shaped subnet until the whole scatternet is formed. Simulation results showed that the hybrid mesh tree achieved better network performance than Bluetree and generated an efficient scatternet configuration for various sizes of Bluetooth scatternets.

key words: Bluetooth, ad hoc networks, scatternet formation

1. Introduction

A Bluetooth-based multihop ad hoc network brings a number of challenges. In addition to the methods of device discovery for a node to participate in multiple piconets, the scatternet formation algorithm is a major technical issue. The scatternet formation algorithm deals with the problem of how to construct individual piconets and connect them together into a scatternet.

Based on the purpose of scatternet, a number of different topology models [1] can be generated. Scatternet topology models can be generally classified as tree hierarchies (TH), master/slave meshes (MSM) and master/slave rings (MSR), among others.

Bluetree [2] was the first scatternet formation protocol to build a tree hierarchy (TH) for a Bluetooth ad hoc network. It adopts one or several root nodes to start the formation of a scatternet. The resulting topology is tree-shaped and uses master/slave nodes to serve as relays throughout the whole scatternet. Although its spanning tree architecture achieves a minimum number of connection links between any two nodes, its tree-shaped topology is not reliable under dynamic topological changes [3]. In addition, the algorithm can be implemented easily but the root node is likely to become a bottleneck.

Bluenet [4] and Bluestar [5] are good master/slave mesh (MSM) examples. Bluenet sets up a scatternet in a distributed fashion, and it shows that a mesh-like architec-

ture achieves higher information-carrying capacity than a tree-shaped one. With Bluestar, each node initially executes an inquiry procedure in a distributed fashion to discover its neighboring devices, and then a number of masters are selected based on the numbers of their neighbors. Finally, a number of gateways are selected by these masters and a mesh-like scatternet is formed. The MSM model may use master/slave or slave/slave nodes as relays to increase scatternet performance with additional protocol complexity.

To increase the scatternet reliability for the TH model [6] and simplify the formation protocol for the MSM model, a hybrid mesh tree formation method was proposed. This method uses a designated root to propagate constant k in its downstream direction to construct the MSM model and determine the new roots for its descendant TH nodes. Then, each new root starts to build its own tree subnet until the entire scatternet is formed.

The rest of this letter is organized as follows: In Sect. 2, the hybrid mesh tree scatternet formation algorithm is presented. In Sect. 3, computer simulations are used to compare the scatternet performance between hybrid mesh tree and Bluetree. Finally, conclusions are stated in Sect. 4.

2. Hybrid Mesh Tree Networks

At the beginning, a new root selection process is designed in the designated root to determine new roots on a tier-by-tier basis in the downstream direction (out from the designated root) during scatternet formation. Then, each new root constructs and coordinates its own local tree-shaped subnet. In addition, a return connection algorithm is used for the first root to convert its tree-shaped subnet into the mesh-shaped subnet. The hybrid mesh tree scatternet formation algorithm is described as follows.

With a new root selection process, the designated root sets constant k as a parameter. With the k parameter, the first root pages up to seven neighboring slaves to form a piconet. Each slave then switches its role to master (called M/S node) and pages additional slaves. These new masters decrease k by 1 and continue to propagate the k parameter in the downstream direction. Afterwards, the new masters begin to page up to seven neighboring slaves and connect their slaves to form their own piconets. Finally, each new master will switch to return mode and wait for the return signal notification.

In this method, when the (k) th master is reached, $k = 0$. The master becomes a new root and the k propagating pro-

Manuscript received November 10, 2010.

Manuscript revised February 23, 2011.

[†]The author is with the Department of Communication Engineering, Chung-Hua University, Hsinchu, Taiwan.

a) E-mail: ycm@chu.edu.tw

DOI: 10.1587/transinf.E94.D.1178

cess stops here. The root selection process continues until all new roots are selected. Then, the tree-shaped subnet of the designated root is created. After each new root is determined, it notifies its upstream masters to start a return connection procedure to connect with additional master/slave (M/S) nodes. During the return connection procedure, each returning M/S node alternately switches its state between page and page scan activities to connect with the other M/S nodes. This procedure is operated iteratively until the designated root is reached. As a result, the tree-shaped subnet of the designated root is converted into a mesh-shaped subnet.

At the same time, the new roots start to page and connect up to seven neighboring slave nodes to form their piconets. Then, the paged slaves switch their roles to master (called M/S nodes). Afterwards, the new masters begin to page up to seven neighboring slaves and connect their slaves to form their own piconets. This procedure is operated iteratively until the leaf nodes of the tree are reached. When the leaf nodes of M/S cannot page and connect any other slave nodes, the M/S nodes change their role to slave nodes and the algorithm stops here. As a result, the mesh-shaped topology of the designated root is formed and each new root manages its own tree-shaped subnet.

It is assumed that when two or more masters try to connect with a slave, this slave node will be affiliated with the master node whose page signal reaches it first. After finishing the scatternet formation, a hybrid mesh tree architecture is formed, the immediate master/slave (M/S) nodes function as relays, and only the leaf nodes play the role of slave.

Here, $k = 2$ was used in Fig. 1 as an example to describe the hybrid mesh tree scatternet formation process. At the beginning, the designated root R1 connects with the first tier masters, as shown in Fig. 1 (a). Then the first tier masters decrease k by one and continue to connect with their downstream masters. When the second tier masters are reached and the counter limit $k = 0$, these masters become new roots, as shown in Fig. 1 (b). The tree-shaped subnet of the designated root is created.

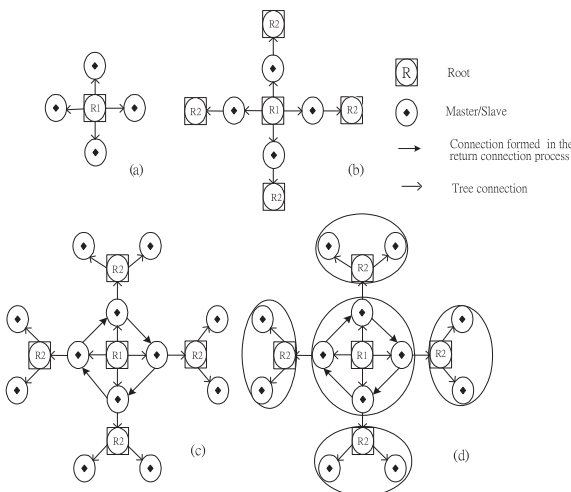


Fig. 1 Scatternet formation process of hybrid mesh tree.

These new roots ask their upstream masters to start the return connection procedure until R1 is reached. The topology of the designated root is finished and generates a mesh-shaped subnet. At the same time, these new roots start to page new slaves and connect with their immediate downstream masters (leaves in this example), as shown in Fig. 1 (c), to build their own tree-shaped subnets. Finally, the mesh-shaped topology of the designated root is formed and each new root manages its own tree-shaped subnet, as shown in Fig. 1 (d).

3. Scatternet Performance

In this section, the performance of the scatternet formation algorithms was simulated and compared for both the hybrid mesh tree and Bluetree. In the simulation of scatternet performance, it was assumed that Bluetooth nodes were randomly located within a rectangular area of $40 \times 40 m^2$, while a radio transmission range of 10 meters was assumed and the number of simulated nodes ranged from 40 to 160. A set of performance metrics was calculated by averaging over 50 randomly generated topologies for each simulated node number. The scatternet performance metrics included the average number of nodes in a piconet, the average path length and the average number of formation packets. The simulation results for both hybrid mesh tree and Bluetree are shown as follows.

Figure 2 shows the piconet efficiency (average number of slave nodes in a piconet) of the hybrid mesh tree and Bluetree. With Bluetree, the piconet efficiency was less than two, since all nodes played the role of M/S except the leaf nodes. With the same spanning tree topology, the hybrid mesh tree achieved better piconet efficiency than Bluetree, since each piconet could connect with more M/S nodes as its slaves during the return connection.

An average hop length between any two nodes was also calculated for these networks. A larger average hop length implies that it will take more time to deliver a packet. This metric usually provides a coarse estimation of the average packet transmission delay of the scatternet.

In Fig. 3, it can be observed that the hybrid mesh tree achieved significant performance improvements in the av-

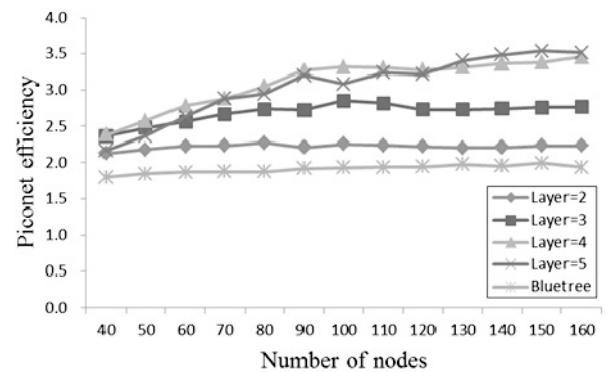


Fig. 2 Average number of slave nodes in a piconet.

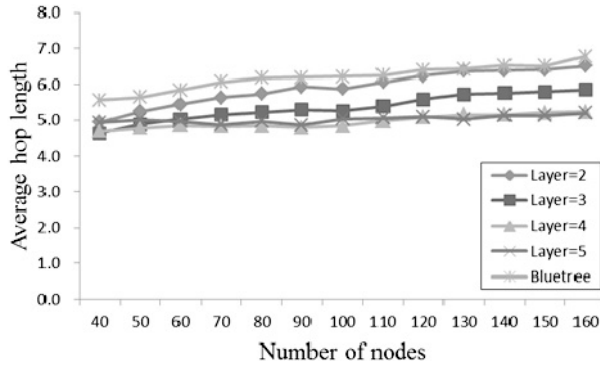


Fig. 3 Average path length in a scatternet.

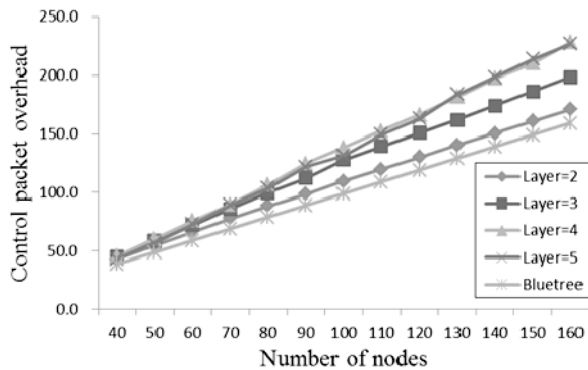


Fig. 4 Average number of formation packets.

average hop length, compared to Bluetree. More return links could be connected among the M/S nodes as parameter k increased, and thus the average path length among nodes could be reduced. In the hybrid mesh tree, $k = 4$ achieved the lowest hop length when the number of nodes ranged from 40 to 120, and $k = 5$ achieved the best performance when the number of nodes was greater than 130. As a result, the optimum number of k with the lowest average hop length was found to vary according to the size of the network.

In real network operations, the performance results of the average hop length can be preset as a lookup table in the designated root to determine the optimum k for different numbers of nodes. The design methodology is described as follows. Initially, the k parameter is set as 2, to form the scatternet of the hybrid mesh tree. After the designated root acquires the scatternet size from all the other nodes, it will determine the optimum k according to the lookup table shown in Fig. 3, and propagate the k in its downstream direction to determine the new roots, as described in Sect. 2.

The formation packets are counted as the total number

of page packets in terms of the number of connecting links during scatternet formation. The link connections represent the aggregated connections including the spanning tree links and return links of the network to improve the scatternet efficiency. In Fig. 4, the hybrid mesh tree spent more control packet overhead as a formation cost than Bluetree. However, this control overhead was only generated during the scatternet formation phase.

4. Conclusions

This letter presented a hybrid mesh tree. With non-uniform distribution applications, the hybrid mesh tree constructs a mesh-shaped topology in one dense area that is extended by tree-shaped topology to other areas. First, this approach used a designated root to construct a tree-shaped subnet and propagate constant k in its downstream direction to determine new roots. Then each new root asked its upstream master to start a return connection procedure to convert the first tree-shaped subnet into a mesh-shaped subnet. At the same time, each new root repeated the same procedure as the designated root to build its own tree-shaped subnet until the whole scatternet was formed. Simulation results showed that the hybrid mesh tree achieved better network topology metrics than Bluetree and generated an efficient scatternet configuration for various sizes of Bluetooth scatternets.

Acknowledgements

My heartfelt thanks to the support of the work by the National Science Council, Taiwan (NSC-99-2221-E-216-025).

References

- [1] K. Persson, D. Manivannan, and M. Singhal, "Bluetooth scatternet formation: Criteria, models, and classification," Proc. First IEEE Consumer Communications and Networking Conference, Jan. 2004.
- [2] G.V. Zaruba, S. Basagni, and I. Chlamtac, "Bluetrees-scatternet formation to enable Bluetooth-based ad hoc networks," IEEE International Conference on Communications, vol.1, pp.273–277, June 2001.
- [3] F. Cuomo, T. Melodia, and I.F. Akyildiz, "Distributed self-healing and variable topology optimization algorithms for QoS provisioning in scatternets," IEEE J. Sel. Areas Commun., vol.22, no.7, pp.1220–1236, Sept. 2004.
- [4] W. Zhifang, R.J. Thomas, and Z. Haas, "Bluenet — A new scatternet formation scheme," Proc. 35th Annual Hawaii International Conference on System Sciences, pp.779–787, 2001.
- [5] C. Petrioli, S. Basagni, and I. Chlamtac, "Configuring BlueStars: Multihop scatternet formation for Bluetooth networks," IEEE Trans. Comput., vol.52, no.6, pp.779–790, June 2003.
- [6] C.M. Yu, S.J. Lin, and C.C. Huang, "On the architecture and performance of Blueweb: A Bluetooth-based multihop ad hoc network," IEICE Trans. Commun., vol.E89-B, no.2, pp.482–489, Feb. 2006.