

# Routing Protocol for Mobile Ad-hoc Wireless Networks

Ivris Marcelo BACARREZA NOGALES

Dept. of Radio Electronics, Czech Technical University, Technická 2, 166 27 Praha, Czech Republic

bacari@fel.cvut.cz, ivris.bacarreza@gmail.com

**Abstract.** Bluetooth is a cutting-edge technology used for implementing wireless ad-hoc networks. In order to provide an overall scheme for mobile ad-hoc networks, this paper deals with scatternet topology formation and routing algorithm to form larger ad-hoc wireless Networks. Scatternet topology starts by forming a robust network, which is less susceptible to the problems posed by node mobility. Mobile topology relies on the presence of free nodes that create multiple connections with the network and on their subsequently rejoining the network. Our routing protocol is a proactive routing protocol, which is tailor made for the Bluetooth ad-hoc network. The connection establishment connects nodes in a structure that simplifies packet routing and scheduling. The design allows nodes to arrive and leave arbitrarily, incrementally building the topology and healing partitions when they occur. We present simulation results that show that the algorithm presents low formation latency and also generates an efficient topology for forwarding packets along ad-hoc wireless networks.

## Keywords

Bluetooth, mobile ad-hoc wireless networks, routing algorithm and performance.

## 1. Introduction

The Bluetooth system operates in the worldwide unlicensed 2.4 GHz ISM frequency band [1]. To make the link robust to interference, it employs a Frequency Hopping Spread Spectrum technique, with carrier frequency changed at every packet transmission. To minimize complexity and to reduce the cost of the transceiver, it adopts a simple binary frequency shift keying modulation (Gaussian FSK), with a bit rate of 1 Mbps.

Two or more Bluetooth units sharing the same channel form a piconet. Within a piconet a Bluetooth unit can have either of two roles: master or slave. Within each piconet there may be only one master and up to seven active slaves. Any Bluetooth unit can become a master in a piconet. Furthermore, two or more piconets can be interconnected, forming what is called a scatternet, Fig. 1. The connection point between two piconets consists of a Bluetooth unit that is a member of both piconets. A Bluetooth unit can simultaneously be a slave member of multiple

piconets, but master only in one, and can transmit and receive data only in one piconet at a time, so participation in multiple piconets has to be on a time division multiplex basis.

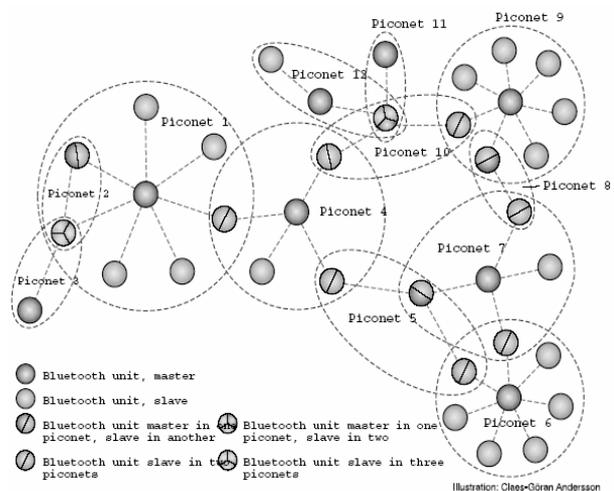


Fig. 1. Scatternet formation.

The Bluetooth system provides full-duplex transmission built on slotted time division duplex (TDD), where each slot is 0.625 ms long. Master-to-slave transmissions always start in an even-numbered time slot, while slave-to-master transmissions always start in an odd-numbered time slot. An even-numbered time slot and its subsequent odd-numbered time slot together are called a frame. There is no direct transmission between slaves in a Bluetooth piconet, only between a master and a slave, and vice versa. The communication within a piconet is organized such that the master polls each slave according to some polling scheme. A slave is only allowed to transmit immediately after the master has polled it. The slave will then start its transmission in the slave-to-master time slot following the packet received from the master. It is possible to send packets that cover multiple slots. These multi-slot packets may be either three or five slots long.

Each Bluetooth unit has a globally unique 48-bit IEEE 802 address. This address is permanently assigned when the unit is manufactured. In addition to this, the master of a piconet assigns a local active member address (AM\_ADDR) to each active member of the piconet. The AM\_ADDR, which is only three bits long, is dynamically assigned and reassigned. The master uses the AM\_ADDR when polling a slave in a piconet. Scatternets may be used

for a number of reasons. First, this is a way to extend the rather limited coverage that Bluetooth provides and reach beyond a single piconet. In addition, the line of sight behavior of the 2.4 GHz radio-wave may be improved if multi-hops between piconets are provided, allowing traffic paths around obstacles.

Second, a scatternet may be the best solution to handle certain traffic and configuration patterns. For instance, when multiple Bluetooth PANs connect with a LAN through a Bluetooth LAN access point (LAP), the LAP is preferably chosen as the master. Since a node can be master of only one piconet, a simple architecture consists of connecting the remaining piconets to the "root" piconet in a tree-shaped scatternet.

Third, the partitioning of a network into several scatternets may give an overall gain in capacity for the same area. For instance, if two slave nodes in different piconets have a sustained traffic requirement, it may be a better choice to create a new piconet containing only the two nodes. In fact, the FHSS access system to the channel makes Bluetooth very robust against interference. Therefore, new overlaid piconets gain substantially more capacity than is lost due to the increased interference between piconets.

Bluetooth employs FHSS (Frequency Hopping Spread Spectrum) to avoid interference. There are 79 hopping frequencies (23 in some countries) each of them having a bandwidth of 1 MHz. Frequency hopping is combined with stop and wait ARQ (Automatic Repeat Request), CRC (Cyclic Redundancy Check), and FEC (Forward Error Correction) to achieve high reliability on the wireless links.

For real-time data such as voice, Synchronous Connection Oriented (SCO) links are used, while for data transmission, Asynchronous Connectionless Link (ACL) links are used. There are several ACL packet types, differing in packet length and whether they are FEC coded or not. Tab. 1 shows the different ACL packet types and their properties.

| Mode | FEC | Packet       |                | Symmetric Throughput (kbps) | Asymmetric Throughput (kbps) |       |
|------|-----|--------------|----------------|-----------------------------|------------------------------|-------|
|      |     | Size (bytes) | Length (slots) |                             |                              |       |
| DM1  | Yes | 17           | 1              | 108.8                       | 108.8                        | 108.8 |
| DM3  | Yes | 121          | 3              | 258.1                       | 387.2                        | 54.4  |
| DM5  | Yes | 227          | 5              | 286.7                       | 477.8                        | 36.3  |
| DM1  | Yes | 27           | 1              | 172.8                       | 172.8                        | 172.8 |
| DM3  | Yes | 183          | 3              | 390.4                       | 585.6                        | 86.4  |
| DM5  | Yes | 339          | 5              | 433.9                       | 723.2                        | 57.6  |

Tab. 1. Different Bluetooth ACL connection modes.

Previous algorithms that are proposed [5], [6], [7] lack the mobility support for ad-hoc wireless networks, and they can not excel in their ability to adjust and adapt to network

changes. Examples include device failure, device out-of-range, or the arrival of additional devices. Our proposal allows for dynamic reconfiguration of the network topology at any time, and overcomes limitations of previous implementations.

In our simulation experiments we have created a model that represents an ad-hoc network, capable to handle both kind of traffics asynchronous data communication and synchronous traffic as voice streaming. In this paper we will be interested only in the analysis of data flow throughput and delay measures, and in the behavior exhibited by the model.

## 2. Bluetooth Model

A simple model has been devised which takes into consideration an environment consisting of a set of piconets. As can be seen from Fig. 1, interference is likely to occur between a set of piconets once the clock settings are not synchronized and, thus, packets from neighboring piconets overlap. This work gives special attention to this case; nevertheless it indicates the required changes in order to adapt it to the scatternet environment.

To be able to carry out our analysis, we did some assumptions. A Bluetooth simulation model was developed as an extension to Network Simulator 2 (NS-2) [8]. The simulation model implements most of the features of the Bluetooth baseband layer like frequency hopping, time division duplexing, automatic retransmission query, multi-slot packets, fragmentation and reassembly of packets. In addition, we extended the capabilities of the simulator in order to be able to support Bluetooth scatternets by the implementation of Bluehoc [9]. Finally, as Bluehoc does not support mobility, we have modified the Bluetooth node structure and implemented soft handoff techniques to the model in order to achieve seamless mobile communication.

Fig. 2 shows the overall architecture of a Bluetooth modified node, which shows us how new modules are integrated into the NS-2 framework, giving us the possibility to have nodes capable to act as master and slave simultaneously, something that in the basic model is not possible. Although a single Bluetooth stack is shown in the figure, a node can be configured to have multiple instances of the stack. Bluetooth AP node would be the case where the AP device needs to serve multiple Bluetooth nodes simultaneously. Tab. 2 describes the main stack components observed in the Bluetooth node structure.

Our algorithm is implemented in Baseband and L2CAP layers, and is divided into three phases, Device Discovery, Network Formation, and Network Maintenance. The Baseband is a simple frame-to-packet layer which creates packets to be sent for radio transmission, or receives radio messages to be sent to higher layers.

At the beginning each node in the network runs the same algorithm, and then when a device starts, the BTHost

shouts randomly the Host Controller Interface (HCI) which provides the convenient HCI\_Inquiry or HCI\_Write\_Scan\_Enables, that cause devices to enter to INQUIRY or INQUIRY\_SCAN, once the devices discover neighboring devices that are within range, a master-slave relationship is formed, then wait for the BASEBAND to accomplish communication among the nodes and then notify the BTAlgo according to benefit matter (INQ\_FINISH and PAGE\_SCAN\_FINISH). Then letting us know whether establish a connection with a distant device and decide to exchange data, or enter round procedures. Round procedures happen periodically, which find the links, and determine which ones have failed and which ones are alive.

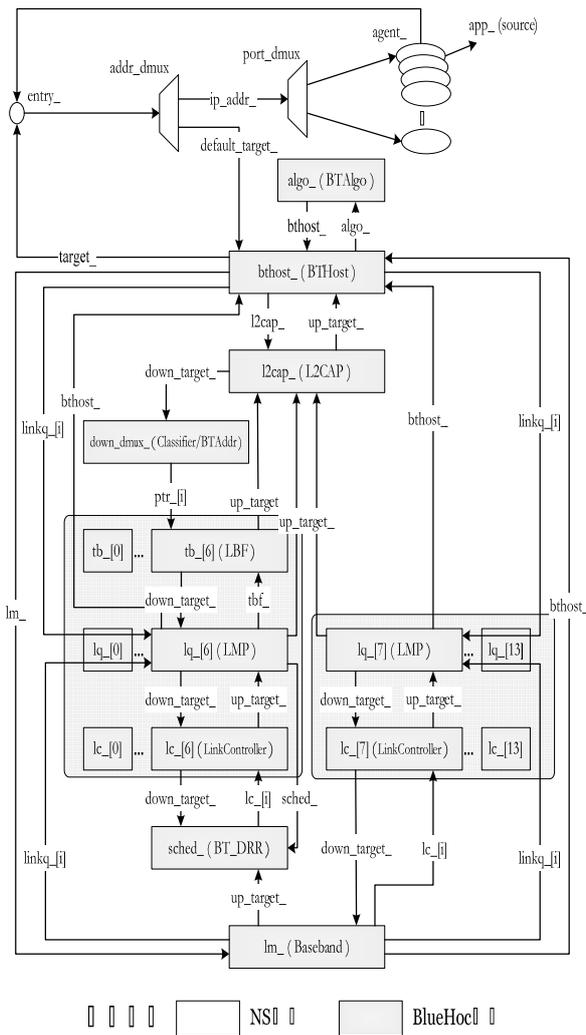


Fig. 2. New Bluetooth node structure.

The goal of the Inquiry process is for a master node to discover the existence of neighboring devices and to collect enough information about the low-level state of those neighbors (primarily related to their native clocks) to allow it to establish a frequency hopping connection with a subset of those neighbors. The goal of the page process is to use the information gathered in during the Inquiry process to establish a bi-directional frequency hopping communication channel.

The routing protocol is an on-demand routing protocol that only discovers a new route when it is required, from this is clear that the protocol is more suited for mobile handheld devices as network bandwidth and battery power is limited, as is the case of Bluetooth technology. To find a route to the destination, the source broadcasts a route request packet (RRP), this message propagates through the network until it reaches an intermediate node that has recent route information about the destination. When intermediate nodes forwards the RRP, they record in their own tables which node the route request came from. This information is used to form a reply path for the route reply packet. All the nodes along the path enter the routing information in their tables. Whenever a link failure occurs, the source is notified and a route discovery can be requested again if necessary.

| Stack component | Responsibility of the item                            |
|-----------------|---|
| Baseband        | Implements basic operations of Bluetooth baseband     |
| BT_DRRScheduler | Deficit Round Robin (DRR) based scheduler             |
| LinkController  | Stop and wait ARQ                                     |
| LBF             | Leaky bucket filter                                   |
| LMP             | Implements LMP commands: signaling and QoS operations |
| L2CAP           | Basic L2CAP functionality: SAR                        |
| BTHost          | Host layered over L2CAP, controls device behavior.    |
| BTClassifier    | Classifies packets based on active mode address       |

Tab. 2. Description of Bluetooth node components.

Bluetooth has a slotted MAC protocol, where master-slave and slave-master communications take turns. Packets can be one, three or five slots long. As result, the amount of allocation and throughput seen in the two directions and the total overhead depend on the asymmetry of traffic on a Bluetooth link.

We have developed an analytical model for such a Bluetooth link. We assume that devices, from one or more piconets or a scatternet, are geographically distributed in a plane according to a Poisson process with  $\lambda$  stations/m<sup>2</sup> density. It is also assumed that a transmitting device is located at the centre of an imaginary hexagonal cluster and operates using an omni-directional antenna. For mathematical convenience, the hexagonal area is normalized to  $\pi$  and the clusters are approximated by circles of the unit radius.

L2CAP avoids packet collision within a piconet, but lack of synchronization among independent neighboring piconets conduces to packet overlapping within a time slot. The transmission power  $P_R$ , received by a receptor located at a distance  $r$ , is computed assuming a propagation model that takes into consideration signal attenuation, lognormal

shadowing due to surface irregularities, and a  $\eta$ -th power loss law, where the propagation loss exponent,  $\eta$ , is around 4. As a result, the received signal  $P_R$  is given by:

$$P_R = \alpha^2 e^{\xi} K r^{-\eta} P_T$$

where  $\alpha^2$  is an exponentially distributed random variable with unit mean,  $\xi$  is a Gaussian random variable with zero mean and variance  $\sigma^2$ ,  $K r^{-\eta}$  refers to the power loss law, and  $P_T$  represents the transmitted power. The same signal propagation model is assumed for all devices. Since the lognormal attenuation variable is given in dBs, the shadowing parameter,  $\sigma$ , is normally given in dB as well. Please note that the notation  $\sigma = 0$  means that the lognormal attenuation in dB is Gaussian with a null variance, in other words it is constant. This work uses a capture model which assumes that a receiver may correctly detect and receive a signal with power  $P_0$  if

$$\frac{P_0}{\sum_i P_i} > b$$

where  $P_i$  represents interference resulting from the  $i$  transmission of packets at the piconet  $i$ , and  $b$  represents the capture threshold. Since the ISM band is an open one, other interference caused by non-Bluetooth devices is outside the scope of this study.

### 3. Simulation Results

In this section, we present simulation results of a simple scenario under a typical network configuration. In our scenario, a mobile node transfers a data flow to its server node on the network through an associated Bluetooth Access Point node.

There are two possibilities by which an AP device and mobile devices determine their roles. Note that the AP device can act as the master or the slaves. First, the AP acts as the slave. When a mobile device comes in the range of the AP, it starts the page-inquiry procedure for a connection to the AP. These procedures form a new piconet where there are only two members. Consequently, the mobile device becomes a master device in the piconet. This approach brings time delay overhead taken by the AP node to switch among the different masters belonging to several unsynchronized piconets simultaneously.

In contrast, an AP can play the master role and mobile devices would become slaves associated with it. In this approach, a new-coming device initiates a page-inquiry procedure as in the first approach. On connection to the AP, it requests master-slave role-reversal by LMP\_switch\_req. Then, the AP becomes the master and the mobile device is added to the group of slaves associated with the AP.

This scheme enables the AP to control slave's access to the link, since only the slave addressed in the master-to-slave slot is allowed to use the following slave-to-slave

slot. But the number of slaves would be limited to seven in this approach. Once network formation is completed, then information can start being transmitted in new topology.

We ran a simple data flow scenario simulation. Fig. 3 shows the results of the experiments where 3 different size files, 100 Kbyte, 500 Kbyte, and 2.5 Mbyte, are transferred in 2 to 8 node piconet configurations. As shown ftp time increases significantly for a larger file, as the number of mobile nodes in the piconets increases. It is well matched to our expectation that the Bluetooth over-the-air link would be a bottleneck in an overloaded piconet. Ftp throughput averaging 3 different size experiments are shown in Fig. 4. The throughput reaches up to 640 kbps in 1-to-1 master-slave configuration, but it drops as multiple nodes contend to access the shared over-the-air channel. The maximum throughput is achieved after an initial ramping period of about 15 seconds. This is due to initial contention of the five mobile devices at the AP node.

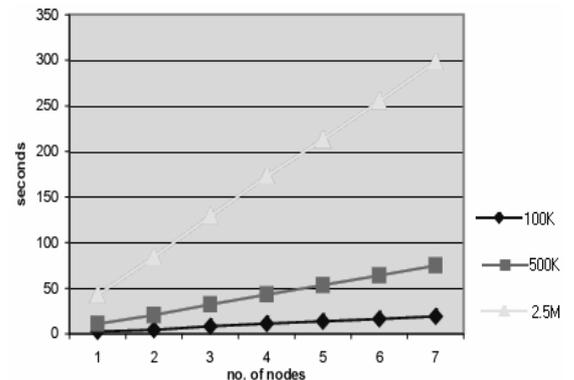


Fig. 3. Delay vs. Piconets when transferring different size files.

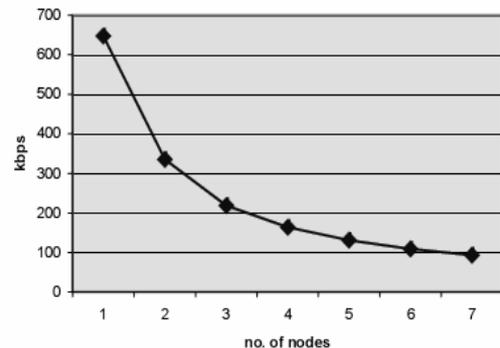


Fig. 4. Throughput vs. Piconets at average rate.

The proposed criteria lead to a good improvement in network performance. The results presented may be seen as a first attempt to design efficient mobile Bluetooth networks topologies, where several nodes attempt to transfer data when they are on the move. We will be working to find a mathematical framework that will allow us the design of a mobile ad-hoc routing protocol.

### 4. Conclusions

In this paper, we presented a novel model for mobile ad-hoc routing protocol, which ensures efficient, seamless,

and provide support for users on the move. The extensions that were implemented in the Bluetooth model demonstrate the flexibility offered at lower layers of the Bluetooth stack. We have found that two characteristics, namely the amount of bridging overhead and the number of established Bluetooth links, have a major impact on system performance. Namely, there is a link number where the throughput is maximized. This implies that for good performance, it is fundamental to decrease bridging overhead as much as possible, and that the number of established Bluetooth links has to be well controlled by any scatternet formation protocol.

There are still many challenges facing mobile ad-hoc wireless networks. However, because of their inherent advantages, wireless ad-hoc networks are becoming more and more prevalent in the world.

Our next steps are directed to get better understanding of the full impact of Bluetooth technology on mobile ad-hoc wireless networks, by eliminating some simplifying assumptions that have been considered in the past, and to find the mathematical model mentioned in the previous section.

## References

- [1] Bluetooth SIG, Specification of the Bluetooth system, Core, Version 1.1. *Bluetooth SIG Homepage: [www.bluetooth.com](http://www.bluetooth.com).*
- [2] LAW, CHING, SUI, KAI-YEUNG A Bluetooth scatternet formation algorithm. In *Proc. IEEE GLOBECOM 2001*. 2001, vol. 5, pp.2864-2869.
- [3] MISIC, V. B., MISIC, J. Minimizing end-to-end delays in Bluetooth scatternet with a slave/slave bridge. In *Proc. 7th Intel. Conf. on Computer Communications and Networks*, 2002, pp.634-639.
- [4] SATO, T., MASE, K. A scatternet operation protocol for Bluetooth ad hoc networks. In *5th Intel. Symp. on Wireless Personal Multimedia Communications*. 2002, vol. 1, pp.223-227.
- [5] ZARUBA, G. V., BASAGNI, S., CHLAMTAC, I. "Bluetrees—scatternet formation to enable Bluetooth-based ad hoc networks. In *Proc. IEEE International Conf. on Communications, ICC2001*. June, 2001, vol. 1, pp.273-277.
- [6] TAN, G., MIU, A., GUTTAG, J., BALAKRISHNAN, H. "Forming scatternets from Bluetooth personal area networks. *MIT Technical Report*, no. MIT-LCS-TR-826, October 2001.
- [7] JOHANNSON, P., KAZANTZIDIS, M., KAPOOR, R.; GERLA, M. Bluetooth: an enabler for personal area networking. *Network*, IEEE vol. 15, Issue 5, Sep/Oct 2001, pp. 28 - 37 Digital Object Identifier 10.1109/65.953231
- [8] NS-2, Network Simulator ns-2, *web page: [www.isi.edu/nsnam/ns](http://www.isi.edu/nsnam/ns).*
- [9] BlueHoc, IBM Project web page: [www-124.ibm.com/developerworks/projects/bluehoc](http://www-124.ibm.com/developerworks/projects/bluehoc)

## About Authors...

**Ivris Marcelo BACARREZA NOGALES** was born in La Paz, Bolivia. He received the Bc. degree in 2001 and the M.Sc. degree in 2003, both in Computer Science and Engineering from the Czech Technical University in Prague. Actually he is a candidate for a Ph.D. degree at the same University. His research interest includes modeling and performance analysis of network protocols, media access control, and quality of service for IP and wireless network technologies. He received several international networking certifications as ITIL, CCNA, CCDA, CCNP, CCDP, CCIP, IP Communication specialist and Wireless specialist; and he is a member of the IEEE.