A Scalable Bluetooth Scatternet Formed by Autonomous Mobile Robots and Cell Phones

Tamás Kovács, Attila Pásztor and Zoltán Istenes

Kecskemét College, Department of Informatics, 6000 Kecskemét, Izsaki 10, kovacs.tamas@gamf.kefo.hu, pasztor.attila@gamf.kefo.hu

Eötvös Loránd University, Faculty of Informatics, Department of Software Technology and Methodology, Budapest Hungary, istenes@inf.elte.hu

Abstract: In the present work a novel communication scheme of an autonomous robot swarm via Bluetooth radio is investigated. In the presented solution an autonomous unit is equipped with two independent Bluetooth radios and so a relatively fast communication is possible in an arbitrary big swarm. Besides, the use of Bluetooth technology makes possible to directly involve cell phones in the communication network. A few elements of the possible scenarios were implemented by NXT robots and two cell phones. It was found that the reliability and the speed of such a communication scheme are satisfactory and give rise to future applications.

Keywords: Bluetooth scattenet, mobile robots, cell phones

1 Introduction

Wireless communication plays an important role in numerous swarm intelligence applications solved by autonomous mobile robots. In most of the cases the mobile robots at hand are equipped with Wi-Fi (IEEE 802.11), Zig-Bee (IEEE 802.15.4) or Bluetooth (IEEE 802.15.1) radio system, since they render cheap and yet satisfactory solutions for the communication between the mobile robots.

The Bluetooth radio system, which is the subject of the present work, is a cheap solution and compared to its relatively high data rate it is economic on power. Due to these advantages the Bluetooth technology is the most commonly used on small mobile devices, and therefore it is a good candidate also in the case of mobile robots. More detailed pros and cons on Bluetooth communication in mobile robotics can be read in [1] or [2]. The most serious limitation of a Bluetooth network is that it is not scalable, since a Bluetooth piconet can consists of a master and at most seven slaves [3]. In order to overcome this limitation the so called Bluetooth scatternet is invented shortly after the appearance of the original

Bluetooth standard [4, 5]. The basic idea of forming a scatternet is that a slave disconnects from its master and becomes only a passive member (park or hold mode) in its original piconet, and then asks for and gets admission into another piconet as a slave or a master. Thus there can be communication between the two piconets through this, so called, bridge unit: if there is a packet or a message directed to the other piconet the bridge takes it, changes piconet and passes the packet towards its destination. However, each of these bridging actions causes some delay and acts as a bottleneck [6]. In addition to this, the position of the bridge unit is more restricted than the others, since it must be in the radio coverage in both piconets. Due to this shortcomings this bridge based scatternet, although arbitrary scalable from theoretical point of view, is limited to not to big networks and low data rates in practice.

Sohrabi et al. [7] and later Leopold et al. [8] proposed a novel and simple solution that employed two independent Bluetooth radios in single autonomous host to form a large scale wireless sensor network. In this scheme the two Bluetooth radios are parts of two different piconets so that the host passes the information between its two radios. It is easy to see that this solution is free from the limitations of the former bridge based scatternet, although it has higher cost.

In the present work we employed the dual-radio scatternet scheme described above to form a scalable communication network of mobile robots and, besides, our implementation also involves Bluetooth equipped cell phones as parts of the network. In order to test our scheme in reality we used the microcontroller based NXT robot assembling set produced by LEGO. This set is based on Bluetooth communication and planned mostly for educational purposes, however, there are also numerous research applications using NXT.

2 The Dual-Radio-based Scatternet

The most significant part of the NXT robot building set is the main brick that contains the central microcontroller, several IO ports and the Bluetooth unit controlled by an ARM7 chip. By the means of the previous section our first goal is to create a communication unit equipped by two independent Bluetooth radios. The NXT is not a skeleton system i.e. it is not possible to extend the original electronics of the central brick. That is why we invented an unusual solution, the basic idea of which is that two central bricks are connected to each other through their RS485 ports. These ports regularly serves as a receiver of measured data from various sensors of NXT standard, however, they can not only receive but also send data via the NXT sensor cable, although these sensors do not require data of any kind. Using this capability of the RS485 ports, a fast half-dulpex communication link can be established between the two bricks. Thus, the two bricks form a new autonomous unit that has two independent radios (each in its own host brick), as it is seen in Figure 1. A Photograph of such an autonomous communication unit can be seen in Figure 2.

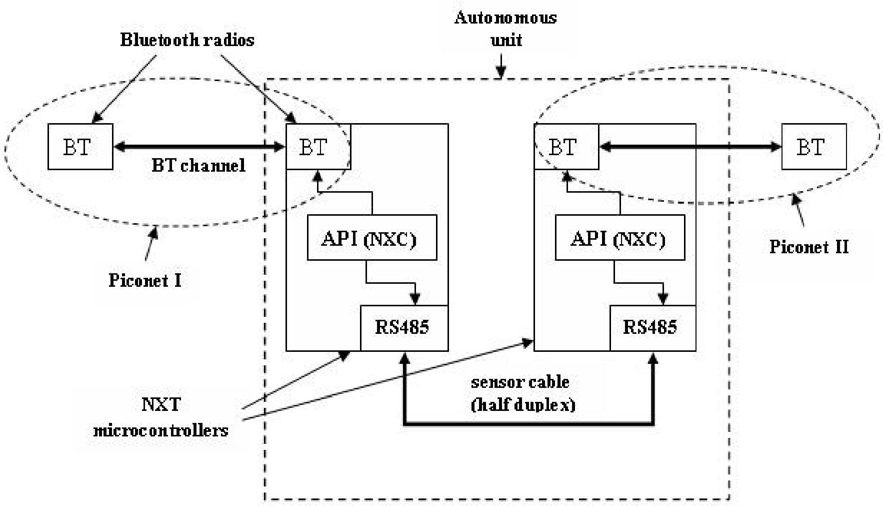

Figure 1
The architecture of the autonomous communication unit consisting of two NXT microcontrollers.

Figure 2
Two NXT central units connected to each other by a sensor cable through their RS485 ports.

The point to point communication or on the Bluetoth either on the RS485 link must be controlled by program codes in the application layer. In the present project we applied the C-like NXC programming language of the NXT microcontroller, which contains API functions to treat the ARM7 controller and the RS485 port. (The Java-like Lejos or the assembler-like NBC programming tools are also capable for the purpose.)

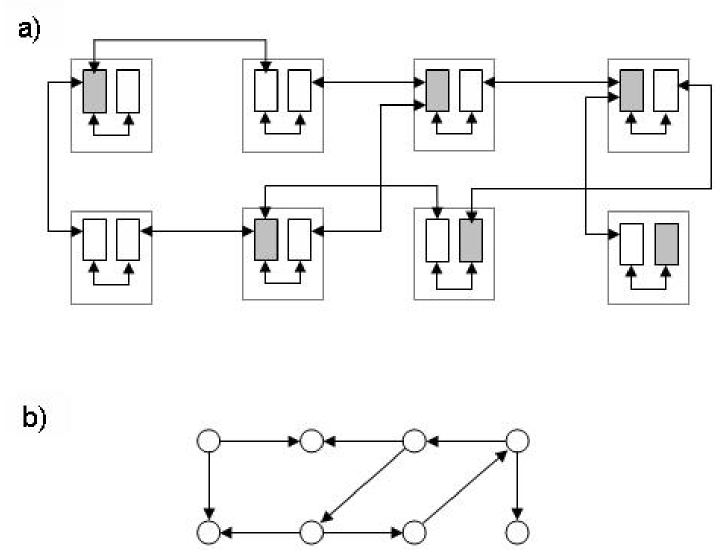

Figure 3
An example of mapping a dual radio based Bluetooth scatternet (b) into a directed graph (a). The master and slave units are marked by shaded or white rectangles, respectively.

Each of the radios of the autonomous unit can be either master or slave in the piconet it belongs to. Every unit with the two radios is a connection point between the two piconets, so a large connected network can be formed by such units. Let us represent this network by a directed graph, where the nodes correspond to the autonomous units and the directed links denote the Bluetooth connections so that the direction is from the master to the slave. The direction plays role in the data link layer and the MAC sub-layer in of the Bluetooth system. An example of mapping a real network into the directed graph can be seen in Figure 3.

There are a several simple rules regarding the directed graph, which arise from the architecture of our double microcontroller unit and the limitations of the Bluetooth system:

- There can be at most two arrows pointing to a node; and if there are two arrows pointing to the node then there cannot be arrow that starts from the node. In this case this node is a slave-slave double radio unit.
- If there is an arrow pointing to a node then there can be at most seven arrows starting from that node in the case of a full capability Bluetooth

radio. This number in case of the NXT Bluetooth radio is three. This rule holds because there can be at most seven (three) slaves in a piconet.

- If there is no arrow pointing to a node then there can be at most 14 arrows starting from that node in the case of a full capability Bluetooth radio. This number in case of the NXT Bluetooth radio is six. In this case this node is a master-master double radio unit.

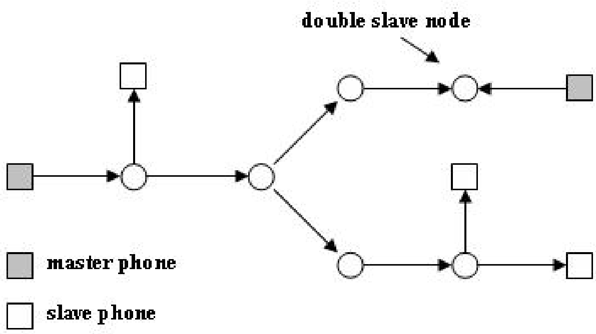

Figure 4
An example of a robot - cell phone hybrid network, that is capable of communication by broadcasting.

It is also possible to involve Bluetooth equipped cell phones into this network, however the cell phones must be end points (leafs) of the graph since they have only one Bluetooth radio. With the help of this multi-hop network the cell phones can communicate with each other so that in this scenario the cell phones are the clients and the mobile robots render service for them. As an example, Figure 4 shows a tree structured hybrid (cell-phone and mobile robot) network, where the phones can communicate with each other by simple broadcasting.

Various self-assembly algorithms to create such networks are detailed by Sohrabi et al. [7] for sensor nodes with fixed spatial positions. The situation is much more difficult when the nodes are mobile robots or cell phones carried by their owners, since the topology is continuously changing. In that case a continuous maintaining the network connectivity and routing information is necessary. Working out such connectivity preserving and routing algorithms is beyond the scope of the present work. Our aim is to implement a real communication capability test for a simple topology consisting of several autonomous NXT robots assembled by the scheme above and two cell phones. This communication test is described in the next section.

3 The Communication Tests and Results

The test presented here is based upon variable number of robots and two cell phones arranged in a linear graph topology shown by Figure 5. The phones were Java MIDP 2.0 enabled ordinary devices equipped with Bluetooth. In order to get a picture about the communication speed and reliability a simple ping application was written for the cell phones using the Sun’s Java Microedition tool, which is capable to control the Bluetooth radio by its JSR-82 packet. In this application the cell phones initiated a serial port connection to the a priori appointed neighbouring robot and the sender phone began to send small (several characters) ping packets into the chain, while the other phone replied automatically the ping packets. Both cell phones were master so one robot at one of the chain ends had to be of slave-slave type. Actually, in our scheme (i.e with the NXT robots) it is quite a tricky task to set up the cell phones’ Bluetooth as slave, because there is no service discovery protocol implemented in the NXT Bluetooth system, therefore, the NXT unit cannot initiate a serial port connection. This problem is to be solved yet.

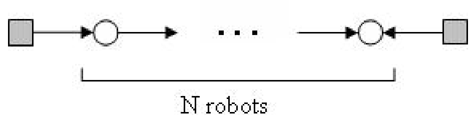

Figure 5
The testing scenario. The cell phones are denoted by rectangles.

A ping procedure sent ten packets in one series. The next packet was sent as soon as the acknowledgement arrived. The success rates and the average reply times were measured and recorded for various robot chain lengths of N = 1, 2, 3, 4 and 5. For each chain length five ping procedures were executed. The success rate was 100 per cent in any cases, and the average reply times are shown in Figure 6.

It can be seen that each robot-robot link raises the hopping time by approximately 0.5 seconds. This seems to be a bit long, but it should be noted that the program governing the packet hopping was written so that acknowledgements were applied at every hop in order to increase the reliability. By giving up some amount of reliability it is possible to optimize the program to get higher speed.

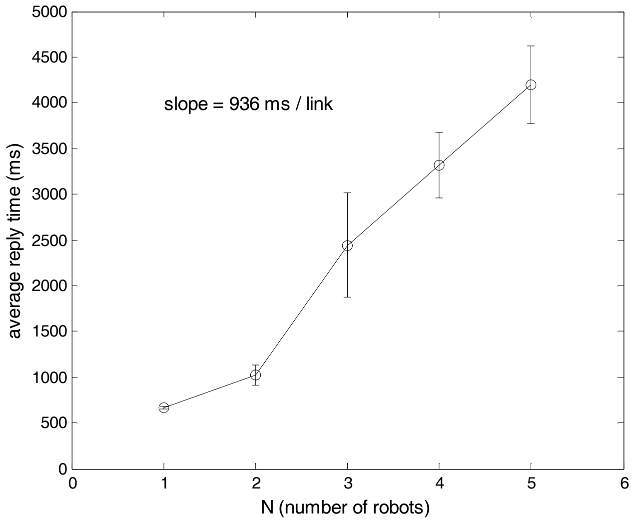

Figure 6
The average reply time as function of the number of the robots in the multi-hop chain.

It is also important that the experiments were executed in a closed room with solid walls the diameter of which was less than ten meters that is each Bluetooth radio was within the coverage of all of the others. In spite of this closeness the possible interference did not corrupt the packets.

Conclusions

Based on the implemented test of our linear scenario the following conclusions can be drawn:

The LEGO's NXT robot builder set is capable of creating a Bluetooth dual radio unit, which can serve as a basic element in building up scalable multi-hop communication networks.

This network can easily connect to Bluetooth equipped cell phones and render an alternative communication service for them.

The multi-hop communication can be made perfectly reliable with an acceptable hopping time delay, which was 0.5 seconds in our experiment.

Acknowledgement

We would like to thank our scientific worker students, Ádám Mészáros and Péter Vigh for their devoted help in the software developing work.

References

[1] J. Bae, R. M. Voyles: Wireless Video Sensor Networks over Bluetooth for a Team of Urban Search and Rescue Robots. ICWN 2006, pp. 409-415

[2] U. Witkowski, M. El Habbal, S. Herbrechtsmeier, A. Tanoto, J. Penders, L. Alboul, V. Gazi: Ad-hoc Network Communication Infrastructure for Multi-robot Systems in Disaster Scenarios. In: Proceedings of IARP/EURON Workshop on Robotics for Risky Interventions and Environmental Surveillance (RISE 2008), Benicassim, Spain 2008

[3] J. C. Haartsen: The Bluetooth radio system, IEEE Personal Communications 7, 2000, pp. 28-36

[4] R. M. Whitaker, L. Hodge, I. Chlamtac: Bluetooth scatternet formation: A survey, Ad Hoc Networks 3, 2005, pp. 403–450

[5] O. Al-Jarrah, O. Megdadi: Enhanced AODV routing protocol for Bluetooth scatternet, Computers and Electrical Engineering 35, 2009, pp. 197–208

[6] A Pasztor, T. Kovacs; Z. Istenes: Swarm intelligence simulation with NXT robots using Piconet and Scatternet, Applied Computational Intelligence and Informatics (SACI '09), 2009, pp.199 - 204

[7] K. Sohrabi, W. Merrill, J. Elson , L. Girod, F. Newberg , W. Kaiser: Scalable Self-Assembly for Ad Hoc Wireless Sensor Networks, IEEE Transactions on Mobile Computing, 2002, pp.317-331

[8] M. Leopold, M. B. Dydensborg, P. Bonnet: Bluetooth and Sensor Networks: A Reality Check, in Proc. 1st ACM Conf. Embedded Networked Sensor. Systems (SenSys 2003), pp. 103–113