Odin and Hodur:
Using Bluetooth Communication for Coordinated Robotic Search

D.H. Barnhard, J.T. McClain (Presenter), B.J. Wimpey, and W.D. Potter

Artificial Intelligence Center
Room 111, Boyd Graduate Studies Research Center
The University of Georgia
Athens, GA 30602-7415
Phone: (706) 542-0358
(jmcclain@uga.edu)

Abstract

This paper describes a multi-robot system designed to use Bluetooth wireless communication to solve the "Honeybee" task. The Honeybee task is a simple search and navigation problem that requires a "guide" robot to lead another "blind" robot to a specific target within the environment. Topics discussed include advantages and disadvantages of using Bluetooth in robotics, as well as a variety of behaviors required to solve the Honeybee task.

Keywords: Robotics, Bluetooth, Wireless Communication, Coordinated Search

Introduction

This paper presents a multi-robot system designed to use short range Bluetooth communication to solve a simple search and navigation task. The system was designed as a preliminary step towards the more difficult task of coordinated target identification and tracking with multiple robots in a military-like scenario. The primary goal of this initial project was to create a robust communication system between two robots that would allow for the transmission of location information for coordinated movement. Sub-goals included exploring possible localization schemes as well as determining optimal placement of sensors and other components for future designs.

Although coordinating multiple robots using wireless communication protocols is relatively commonplace, the use of the recently developed Bluetooth protocol for these types of task is original. Thus, the main contribution of this paper is to outline the details of using Bluetooth communication for coordinated robotics.

This project made use of two modified Acroname PPRK robots named Odin and Hodur; Odin after the most prominent Norse deity, described as having given up one of his eyes in exchange for wisdom; and Hodur after the blind Norse god of winter. We chose Nordic names for the robots because Bluetooth is named after Harold "Bluetooth" I, the king of Denmark and Norway in the middle 900's. The architectures of these robots will be described in detail in a subsequent section of the paper.

The Honeybee Task

The "Honeybee" problem presents an interesting challenge to those involved in behavior-based robotics research. The problem is defined as follows. There are two robots, a "guide" robot and a "blind" robot. The guide robot (Odin) possesses the sensory apparatus to find a specific target in the environment; the blind robot (Hodur) does not. The task consists of two goals. First, Odin must explore the environment and discover the target, and then it must lead Hodur to it solely through the communication of its location.

This task is based loosely upon the behavior of the common honeybee. Each honeybee worker possesses the ability to find pollen-producing flowers in the area of the hive. However, when one bee discovers a source of food, it returns to the hive and performs a dance to communicate its location to all the other worker bees present (Koning, 1994). Thus, a large number of bees are able to find the food source without the added cost of a search of the environment. In this way, the food is retrieved much more quickly and efficiently.

Related Work

It is believed that this is the first time that Bluetooth has been used as the sole means of communication in a multi-robot system. Previous work in the area of multi-robot communication has focused primarily on the 802.11*x* wireless standard. Gerkey et al. (2001) developed Player, a network server that provided "transparent network access to all sensing and control" of multiple robots (p. 1226). Nguyen et al. (2003) developed a system for increasing the range of a wirelessly controlled robot by using small slave robots that followed the main robot and relayed the signal onward.

In order for a communication system to solve the Honeybee task, the information being communicated must be accurate. If both localization systems are inaccurate, even slightly, the error will compound and the accuracy of the blind robot will suffer tremendously. This system uses a much simplified version of landmark-based navigation as described by Uther et al. (2001). Their paper described the use of vision processing to find a robot's location and orientation by noting the location of various markers surrounding the task area. In a similar fashion, Odin and Hodur use landmark cues to determine location as well. This method is outlined further below.

Robot Architectures

The robots used in this project consisted of two modified Brainstem Palm Pilot Robot Kits (PPRK) from Acroname Robotics. The Brainstem PPRK is a small robotic platform that is characterized by holonomic motion and an easily modifiable chassis (see Figure 1). They are designed to be run by the Acroname Brainstem, a microprocessor with a 40 MHz RISC processor, 1 MBit Inter Integrated Circuit Bus (IIC) port, 5 digital input/outputs, and 4 high resolution servo outputs. The Brainstem PPRKs were interfaced with Compaq iPAQ 3970s running Pocket PC 2002 and Windows CE 3.0, which were used for primary control of both robots. One particular reason for the choice of the iPAQ as the controller was the ability to use Bluetooth wireless technology to establish a robust communication link between the two robots. Despite the possible use of the Brainstem as the primary controller, this project only used it to relay messages between the iPAQs and the robot's sensors and actuators.

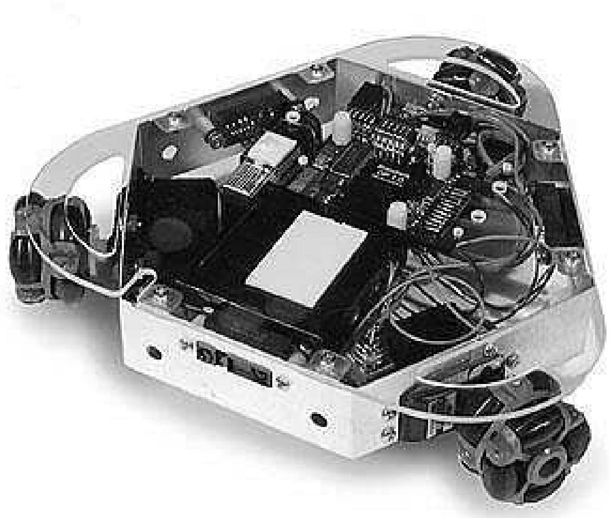

Figure 1: The Acroname Brainstem PPRK

The robot was modified from its original state by removing the infrared sensors from the chassis and adding a Devantech SRF08 Ultrasonic Sensor and a Devantech CMPS03 Magnetic Compass Sensor. Both robots were identical in that they both possessed the same modifications.

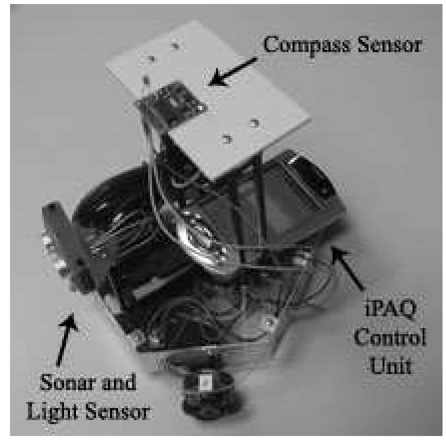

Figure 2: Modified PPRK

The SRF08 sensor is a unique sensor in that it is actually two sensors in one. Not only does it perform ultrasonic ranging that is accurate to the centimeter, but it also possesses a light-intensity sensor in the form of a photo-resistor. Taking a range reading proceeds as follows, the Brainstem sends a query to the SRF08, whereupon the ultrasonic "ping" is created using one of the speakers on the device. After the ping is released, a timer is set. Once the sound has bounced of an object and returned to the device, the time between the two events is returned in microseconds to the Brainstem. From there it is a simple calculation to convert from microseconds to a distance measurement.

The CMPS03 sensor is an electronic compass that works by measuring the Earth's magnetic field. It is advertised as being accurate to 1˚, but in practice we found that accuracy ranged between 1 and 5 degrees. In addition, it is quite sensitive to magnetic interference. In particular, we found that the iPAQ itself gives off an electro-magnetic field that was strong enough to effect the accuracy of the compass within a few inches. Thus, to ensure accuracy, we were forced to place the compass sensor up on stilts, away from any other electrical device on the robot (see Figure 2).

The PPRK chassis is also of note because it allows for holonomic motion, in particular it allows for the robot to rotate on its central axis, making movement of the robot much easier.

Behavior

To accomplish the task, the robots exhibit the following behaviors. First, Odin explores the environment in search of the target, which is in this case a 40-watt light bulb. Once the robot discovers the bulb, it finds its current location and communicates it to Hodur. Hodur then uses this information to navigate directly to the bulb without searching for it. In this case, the robot does not use any sensory apparatus to identify the light bulb; instead it assumes that the information obtained from Odin is perfectly correct. Thus, Hodur is completely dependent on Odin for accurate communication of the object's location.

Exploration

The exploratory algorithm is very similar to the light attraction and repulsion behavior of W. Grey Walter's Tortoise (1953). That is, Odin uses a strictly greedy approach, moving only towards greater intensities of light until the light bulb is discovered. This technique begins with Odin taking an initial measurement of light intensity. Then the robot proceeds to move in a forward direction, continually taking new light measurements. During this movement, the light sensor is continually taking light measurements. As long as the light intensity is increasing, the robot continues to move forward, assuming that it is drawing closer to the light bulb. If the intensity ever decreases, the robot begins to turn, searching for a brighter intensity. To recover from local maxima of light intensity, which are encountered somewhat regularly, Odin moves in a random direction if it has been turning in place for too long and begins the search process again.

In addition to the greedy search, Odin also monitors the space in front of it using the ultrasonic ranging part of the SRF08 sensor. If an object is detected less than two inches in front of the robot, then it is possible that the target has been found. However, it is also possible that the robot has encountered another object in the environment, such as a wall. In this case, the robot does a quick check of surrounding light intensities to determine if it is indeed the brightest item in the environment. If it is not, then the robot takes a random turn and begins the search again. In practice, this situation is relatively rare, but did occur a few times during testing. This condition might become much more important in crowded or dynamic environments, where the robot is more likely to encounter other non-target objects. Once Odin has found the target, it begins the localization phase of the task.

Localization

In order to simplify the task, a simple landmark-based localization scheme was used. The environment in which the task took place was surrounded by a rectangular wall, with each wall aligned with the polar coordinates. Each of these walls was visible to the robots via the SRF08 ultrasonic sensor. In order to find its location, the robot simply needed to turn to two cardinal directions separated by a 90˚ angle (e.g. South and East), and take distance readings for each of the walls. In this way, unique coordinates were obtained that were subsequently transmitted to Hodur via a Bluetooth radio link.

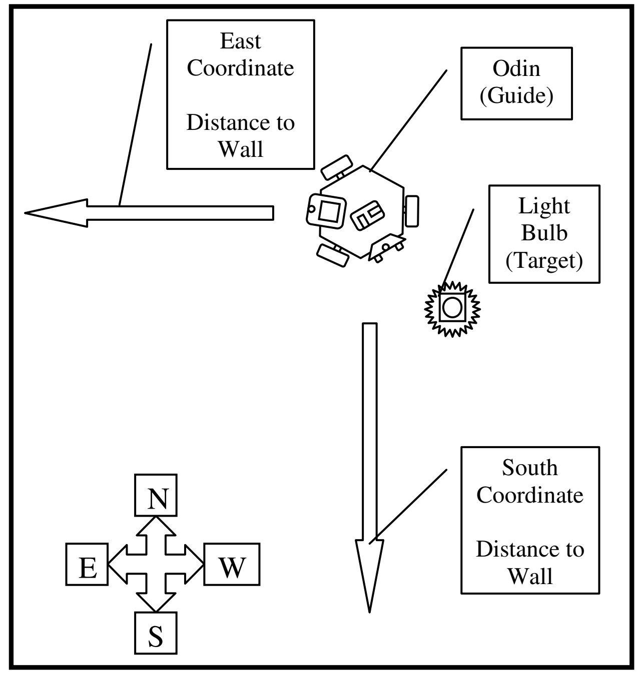

Figure 3: The Localization Process

Hodur's Behavior

Hodur's behavior is remarkably simple and does not depend on any sensory equipment to refine the target position. It waits until Odin's data transmission is complete. Once the transmission is received from Odin, Hodur ascertains its position by turning to the first cardinal direction. It then adjusts its position along that coordinate vector until it matches the distance supplied by Odin. It repeats this procedure for the other coordinate. After these two coordinate adjustments, Hodur has arrived at relatively the same point in the arena as Odin, and the task is complete.

Bluetooth Communication

The main goal of this project was to develop a robust system of communication between both robots for use in future projects. The communication scheme that was eventually chosen was Bluetooth, a proprietary wireless standard invented in 1994 by L.M. Ericsson. Bluetooth was named after Harold Bluetooth I, the king of Denmark and Norway in the middle 900s. Intended to create a short-range radio link between electronic devices, it is characterized by robustness, low complexity, low power, high data transmission speed, security, and low cost. Bluetooth has both advantages and limitations when compared with the more traditional 802.11*x* approach. In their design of a robotic 802.11*x* relay system, Nguyen et al. (2003) found that it was necessary to design a new

802.11x wireless modem, primarily because ordinary, proprietary modems were too large to be practical for robotics. However, the use of Bluetooth eliminates the problem of size restriction, since radios for this standard are small enough to fit into an iPAQ, which is the size of an adult hand.

Another aspect of the 802.11x approach is the need for infrastructure. In Gerkey et al. (2001) and Nguyen et al. (2003), the infrastructure took the form of a control server. While direct communication is possible, Nguyen et al. (2003) described how it is quite difficult. In general, it is at least necessary for a wireless router to be present. In contrast to this, Bluetooth makes it very straight forward to establish a direct serial link between two robots, making it possible to avoid extra infrastructure related to the communication medium. Without any added infrastructure, it is possible to move the robot system from one location to another. Another distinct advantage is the Bluetooth enabled robot's ability to communicate with any other Bluetooth technology, including printers, and wireless phones, to name a few.

A major disadvantage of Bluetooth is its limited range of approximately 30 feet, as opposed to the 802.11x typical range of about 300 feet. While this was not an issue with the current project because of the small size of the task area, when working in larger arenas, Bluetooth type communication would require the robots to remain much closer together. However, it is quite possible to extend the range of the system by using robots to relay messages between one another, much in the same way as was described in Nguyen et al. (2003).

The Bluetooth Protocol stack in Windows CE (WinCE) 3.0 is from a company called Widcomm. A cost-effective way to go about Bluetooth development in WinCE 3.0 is to download BTAccess from High Point Software. The demonstration version of this software development kit is free, and allows access to the Bluetooth stack via the BTAccess libraries. A third party solution is needed for Bluetooth development since software development capabilities did not become integrated within WinCE until version 4.0 (.NET). The communication system was developed with Microsoft Visual C++ 3.0 using Microsoft Foundation Classes (MFC). The iPAQs used were model 3970. The following sections outline the basic steps required to create a robust link between two iPAQs for the purpose of transmitting information.

Creating a Bluetooth Connection

A communication link to another iPAQ is set up in the following manner. First, a connection is established with the Bluetooth stack. This is done using the provided class, `CBtStack`, in the following example:

```
CBtStack  *theStack;
theStack = new CBtStack();
eBTRC eRCmsg= theStack->Connect(this);
```

To add a little more autonomy to the robots, it was necessary to adjust the security settings so that authorization was not required to make a Bluetooth connection between the two iPAQs. This was done by setting the `bAuthorizationRequired` variable to `FALSE` within the `SVC_SECURITY_OPTIONS` structure.

Below, Figure 4 outlines the process to connect to another device via Bluetooth. Once a connection is established from one device, the process must be repeated on the second device to connect to the first.

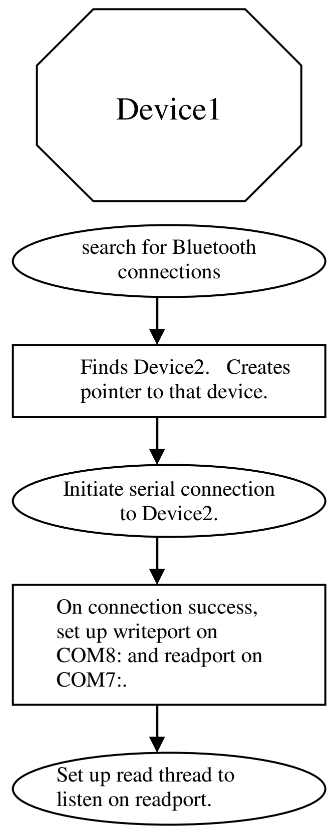

Figure 4: The Connection Process for one Device

Each iPAQ searches for Bluetooth capable devices in the local area to find and connect to each other. Since only the two Bluetooth iPAQs are in the area, they only detect each other. In order to make a connection, both iPAQs must search for and establish a connection to the other. Once another Bluetooth device is found, a `CBtDevice` pointer is created to refer to that device, thereby forming a Bluetooth connection in one direction. The other iPAQ must do the exact same process to form a bi-directional link.

When the iPAQs are connected, a virtual serial connection is initialized for the transmission of data. When the two devices are both connected, read and write ports for communication are established in the form of virtual COM ports (`COM7` for reading and `COM8` for writing). The following example illustrates how to create the virtual COM port for writing. A similar procedure is used to set up the read port; however a new handle other than `sendPort` must be used. For purposes of this example, the write handle will be called `sendPort`.

```
HANDLE sendPort;
sendPort = CreateFile(T("COM8:"),
    GENERIC_READ | GENERIC_WRITE,
    0,
    NULL,
    OPEN_EXISTING,
    0,
    NULL);
```

Now that both the read and write handles have been created, reading and writing is possible. Next, a specific thread must be created for reading. This is done using `AfxBeginThread`, which takes a function, in this case `ReadThread`, and runs it as its own process. `ReadThread` is a user-defined function that uses `ReadFile` to continually monitor `COM7` for new transmissions. Writing is done by calling the function `WriteFile` on `sendPort`. `CreateFile`, `WriteFile`, and `ReadFile` are available in the API for Windows CE OS versions 1.0 and later. The `AfxBeginThread` function is available from the Microsoft Foundation Class library for Windows CE 2.0 and later.

One restriction on the current implementation does not take into account the possibility that Bluetooth enabled products other than the second robot might also be present. In such a situation, it might be possible that the robot would find and connect to the wrong Bluetooth device. This problem was not resolved in order to allow the robots more autonomy when connecting with each other. A better solution to this problem would be to require the robots to authenticate themselves at the application layer.

Another issue with the current implementation that does not have a direct effect on the present project, but will have an impact on future efforts, is that the present approach only allows for two robots to be connected at a time. This is because the present version of the Bluetooth stack does not allow for multiple serial connections using the SPP connect method as described. The obvious solution to this problem would be to create a Bluetooth local area network, similar to those used in the 801.11*x* approaches. However, because of restrictions with the Bluetooth stack implemented on the iPAQ 3970s, this approach is not a viable option for the present architecture. Newer versions of Windows CE (e.g. Windows CE .NET) provide a more robust Bluetooth stack and developer interface allowing for the implementation of more advance connection strategies such as a local area network. Another solution using the present architecture would be to alternate connections between robots using the serial connection approach. This idea will be explored in subsequent phases of the project.

Transmitting Coordinates

Once a connection is established, it is used to share the location of the first robot with the second in the form of X and Y coordinates. These are represented as numerical values corresponding to the distance in inches to a given wall along a cardinal direction.

The data transfer proceeds through a series of sent messages followed by acknowledgments, as depicted in Figure 5. Upon obtaining the coordinates, Odin converts the first coordinate from numerical data into a string of characters for transfer. Upon receiving the transfer, Hodur decomposes the first coordinate into numerical form and sends an acknowledgment reply. When Odin receives the acknowledgment, the second coordinate transfer begins. In this way, any number of messages could be transmitted between the two robots.

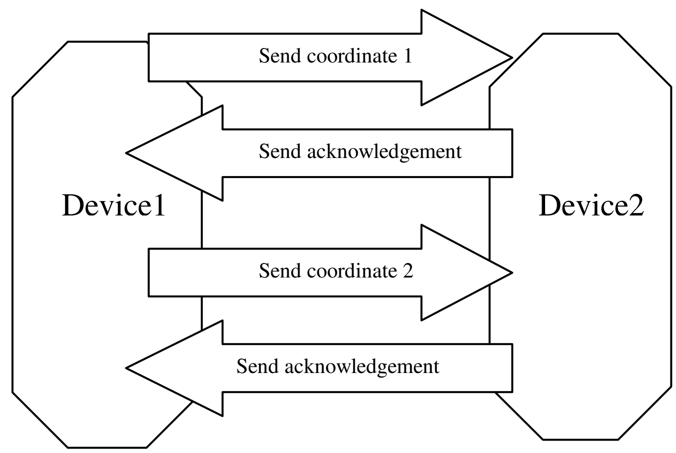

Figure 5: Coordinate Transmission between Devices

Conclusions and Future Directions

This paper has presented a unique solution to the Honeybee task. Of particular note was the implementation of a robust communication link between two mobile robots using Bluetooth wireless technology. In addition, simple solutions to problems of localization and coordinated search were also discussed.

The advantages of the robust communication system described in this paper are directly applicable to any multi-robot system engaged in coordinated search. In many instances, the sensory apparatus required to discover a particular item may be quite expensive, in which case it may only be possible to have one robot that possesses these sensors. The ability to direct other robots to a resource for collection, however, would greatly increase the speed of its retrieval in many circumstances, making communication essential. Even when all robots involved in a search possess the correct sensors to detect an object, it would be advantageous if as soon as one robot discovered it, all other robots were aware of its location.

This system was designed as a prototype for a subsequent project that will use a tank chassis rather than a PPRK. Two of the main purposes of this project were to determine sensor performance as well as provide basic algorithmic design for later models. From this experiment, it has been determined that subsequent designs should include a more robust chassis. Additionally, valuable information was developed in regards to proper sensor placement that will prove to be critical in successive models.

The next project will use model tanks that have been converted into robotic platforms. Advantages of this chassis are that it is extremely robust and allows for greater speed. In addition it allows for better sensor placement. This platform also has a projectile launcher that will allow the robot to actually fire upon the target. The launcher assembly consists of ten foam missiles that are electrically launched and it is hoped that the robot will be accurate enough to hit targets within a two foot radius.

Figure 6: The Tank Chassis

The behaviors of the robots in the next project will be roughly analogous to the current behaviors with one critical difference. In this new strategy, both robots will seek out the target simultaneously in a much larger environment. As soon as the target is found, its coordinates will be sent to the other robot that is still searching. The robot that found the target will continue to track it until the other robot arrives. Once both robots are in close proximity to each other, they will proceed to fire on the target in a coordinated fashion.

In order to accomplish these more advanced behaviors, several sensors need to be added. First, the addition of one more SRF08 sensor will be used to allow for better target acquisition and tracking. By mounting both of these sensors on the left and right forward portions of the chassis the robot can also perform crude motion tracking. Additionally, there will be one infrared sensor that will be mounted on the rear of the tank. This will be used in the obstacle avoidance routine, so it is not possible to run backwards into a wall. The compass sensor will be mounted directly at the center point of rotation for the tank, and will be positioned high enough that there will be no magnetic interference from any other part of the robot.

The last addition will be an improvement to the Bluetooth communication scheme to enable better bi-directional communication. In contrast to the unidirectional flow of information in the current project, it will be critical for the tanks to be able to process information both ways in order to coordinate movements and attack patterns. It is also hoped that through communication, the robots will be able to distribute the processing of the firing solution, and thus increase their overall intelligence.

References

Gerkey, B.P.; Vaughan, R.T.; Stoy, K.; Howard, A.; Sukhatme, G.S.; and Mataric, M.J. 2001. Most Valuable Player: A Robot Device Server for Distributed Control. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1226-1231, Wailea, Hawaii, October 29th – November 3rd, 2001.

Koning, R.E. 1994. Honeybee Biology. *Plant Physiology Website*. http://plantphys.info/plants_human/bees/bees.html (visited 10/20/03).

Nguyen, H.G.; Pezeshkian, M.R.; Raymond, M.; Gupta, A.; and Spector, J.M. 2003. Autonomous Communication Relays for Tactical Robots. *11th Int. Conf. on Advanced Robotics*. Coimbra, Portugal, June 30th – July 3rd, 2003.

Uther, W.; Lenser, S.; Bruce, J.; Hock, M.; and Veloso, M. 2001. CM-Pack'01: Fast Legged Robot Walking, Robust Localization, and Team Behaviors. *RoboCup-2001: The Fifth RoboCup Competitions and Conferences*.

Walter, W.G. 1953. *The Living Brain*. New York, NY: Norton.