DISTRIBUTED COLLABORATIVE ROBOTIC MAPPING

by

DAVID BARNHARD

(Under Direction the of Dr. Walter D. Potter)

ABSTRACT

The utilization of multiple robots to map an unknown environment is a challenging problem within Artificial Intelligence. This thesis first presents previous efforts to develop robotic platforms that have demonstrated incremental progress in coordination for mapping and target acquisition tasks. Next, we present a rewards based method that could increase the coordination ability of multiple robots in a distributed mapping task. The method that is presented is a reinforcement based emergent behavior approach that rewards individual robots for performing desired tasks. It is expected that the use of a reward and taxation system will result in individual robots effectively coordinating their efforts to complete a distributed mapping task.

INDEX WORDS: Robotics, Artificial Intelligence, Distributed Processing, Collaborative Robotics

DISTRIBUTED COLLABORATIVE ROBOTIC MAPPING

by

DAVID BARNHARD

B.A. Cognitive Science, University of Georgia, 2001

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment

of the Requirements for the Degree

MASTERS OF ARTIFICIAL INTELLIGENCE

ATHENS, GEORGIA

2005

DISTRIBUTED COLLABORATIVE ROBOTIC MAPPING

by

DAVID BARNHARD

Major Professor: Dr. Walter D. Potter

Committee: Dr. Khaled Rasheed
Dr. Suchendra Bhandarkar

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
May 2005

DEDICATION

I dedicate this thesis to my lovely betrothed, Rachel. Everyday I wake to hope and wonder because of her. She is my strength and happiness every moment that I am alive. She has continually, but gently kept pushing me to get this project done, and I know without her careful insistence this never would have been completed. So, for all of her kind words, love and support, I dedicate this thesis, and all of the associated work to the love of my life, Rachel. I just hope that I can give back to her all that she always gives to me. I love you Rachel.

ACKNOWLEDGEMENTS

I want to acknowledge everybody who helped me with this thesis. Primary thanks and appreciation go to Dr. Potter. Over the years, I have never known anyone to be as supportive as he has been. Through all of my crazy and deranged ideas, Dr. Potter has always tried to guide me to be a better student and person. Without his support, through the years, this project never would have had a chance to be completed. I also want to acknowledge my colleagues, without whom, it would have been impossible to have completed this project. John McClain, B.J. Wimpey and Julian Bishop were major contributing members to all of the works below. Their hard work and countless hours deserve the highest regard and acknowledgements. This work would not have been possible without their hard work and inspirational ideas. Also, I think that it is important to acknowledge my family. There have been many times that I would have lost my way without their constant support over the years.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

1.0 INTRODUCTION

The practical application of robotics in the modern world requires several challenges to be overcome. It is true that there are robots of many forms already tightly integrated with modern life. For example, modern factories are full of examples of robots that perform many different types of functions. Personal and home robots are available that range in function from vacuum cleaners to entertainment. However, when Artificial Intelligence researchers think about robots, entirely separate types of problems are envisioned. Typically, within the field of Artificial Intelligence the selected robotic problems that are addressed are more directed towards increasing the autonomous abilities of robots. For example, increasing the abilities of computer vision platforms to allow for improved object tracking is one area that could significantly benefit a robot's ability to avoid potential obstacles. In general, the ability to navigate autonomously through environments is a challenging problem within current robotics research. The broader topic of autonomous robot navigation is composed of many smaller challenges. Within this set of challenges is the ability of a single robot to accurately map its environment. Another of the large areas of interest in current research is how to perform a complex task using multiple robots. There is a large amount of complexity associated with distributing tasks over multiple robots. This research intends to combine the challenges associated with mapping an unknown environment with the ability to distribute that task over multiple robots. In order to accomplish this overall task, a series of projects and robots were developed to incrementally develop the abilities necessary to solve this problem.

1.1 CURRENT WORK

This thesis seeks to address the many different types of issues associated with a distributive mapping task in separate phases. The first paper is the model that was used to prototype a group of robots and the interactions required to allow for multiple robot communication. Entitled "Using Bluetooth Communication for Coordinated Robotic Search," the paper is mainly concerned with developing a robotic platform and a basic multi-robot communications structure that can be adapted to allow for added complexity later. The two robots that were used in that task were simplified to allow them to demonstrate the usefulness of the Bluetooth communication network as the method to pass information between robots. Since this research was considered a prototype, the task that was chosen was also greatly simplified in size and scope. The "honeybee" task that was assigned to the group involves one robot finding a target, in that case a light bulb, and then relaying the location information to the other robot who then attempts to move to the communicated location. This verified the usefulness of Bluetooth and also served as the basis for later, more complicated efforts that used similar software to accomplish a much more complicated task.

The paper located in Appendix A, is entitled "Distributed Robotic Target Acquisition Using BlueTooth Communication," and serves to extend the work of the first paper. In this project, the architecture used in the first project was extended to include a custom robotics chassis along with expanded sensory capabilities. For this paper, a robot chassis was constructed using the shell of a heavily modified model tank equipped with 4 dual purpose sonar and light sensors. Additionally, the robot was capable of firing twelve foam projectiles approximately six feet. This robot platform added capability that allowed for faster target acquisition because of the superior drive train of the tank. For this project, there were two tanks that were built in order to allow for coordinated behaviors to be developed and demonstrated. Both tanks were built using the same overall architecture as the previous robots, Odin and Hodur, with the addition of a second Brainstem microcontroller in order to accommodate the enhanced sensor network and additional motor capabilities.

This approach validated the premise that a task could be distributed among a group of robots in a dynamic environment using Bluetooth as the communication protocol. Additionally, the production of a new robotics platform has enabled for greater customization to allow for easier future development. This outcome produced two viable, usable robotic platforms that can be utilized for future projects.

The third paper “Using Multiple Collaborative Robots for Terrain Mapping,” is an attempt to design a system that creates reinforcement based emergent behavior for a group of robots. This project outlines a system that uses rewards, in the form of information credits, to provide the robots with the incentives to perform the desired behaviors to complete the given task. For example, if the task that the robots were assigned to complete was mapping an unknown environment, then the robots would receive information credits, which function like rewards, for each block of the environment that they recorded. Since the overall goal of the project was to encourage multiple robots to collaborate and share their collected information, more information credits were granted for the action of sharing than the action of exploring. The information credits are important to the individual robots because of a survival pressure tax that reduces the amount of credits that a robot has over time. If a robot exhausts their information credits then they are placed into an undesired inactive state for the remainder of that particular mapping task. It is expected that the reward system coupled with the survival pressure tax will produce emergent robotic behaviors that can result in an efficient exploration of an environment and an increased level of collaboration between the individual robots to accomplish the overall distributed mapping task.

CHAPTER 2

USING BLUETOOTH COMMUNICATION FOR COORDINATED ROBOTIC SEARCH[1]

[1] "Odin and Hodur: Using BlueTooth Communication for Coordinated Robotic Search," by D.H. Barnhard, J.T. McClain, B.J. Wimpey, and W.D. Potter. Proceedings of the 2004 *International Conference on Artificial Intelligence (IC-AI '04)* Volume 1, pp.365-371. Las Vegas, Nevada, 2004.

2.0 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.

2.1 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.

2.2 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 802.11x 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.

2.3 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 2.1). 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.

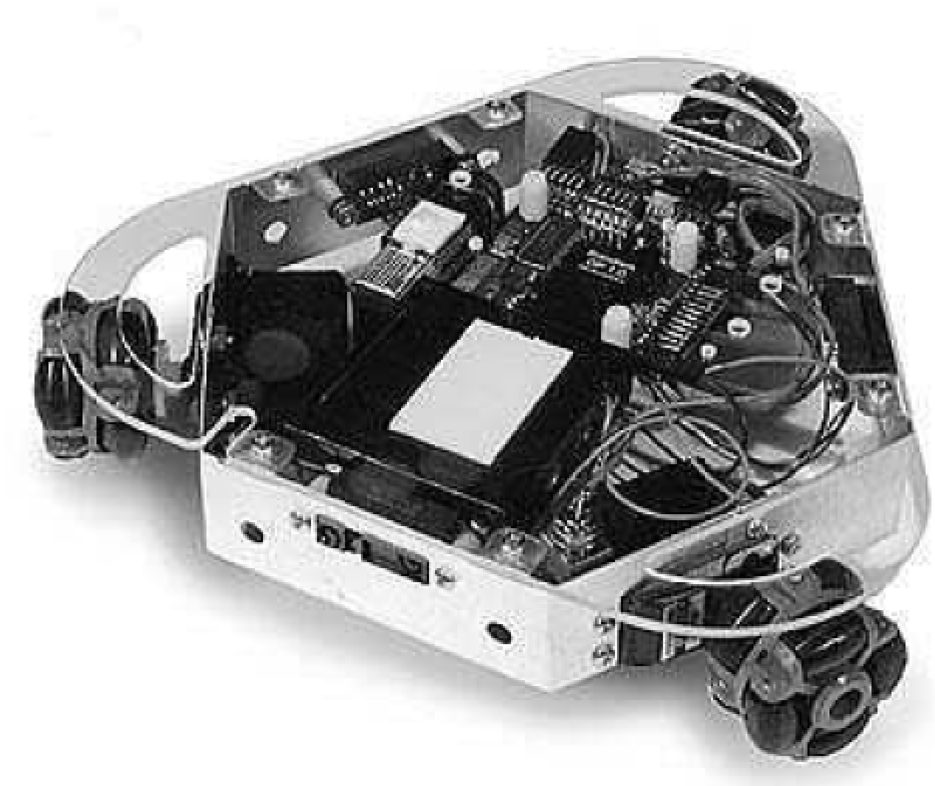

Figure 2.1 The Acroname Brainstem PPRK

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 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.

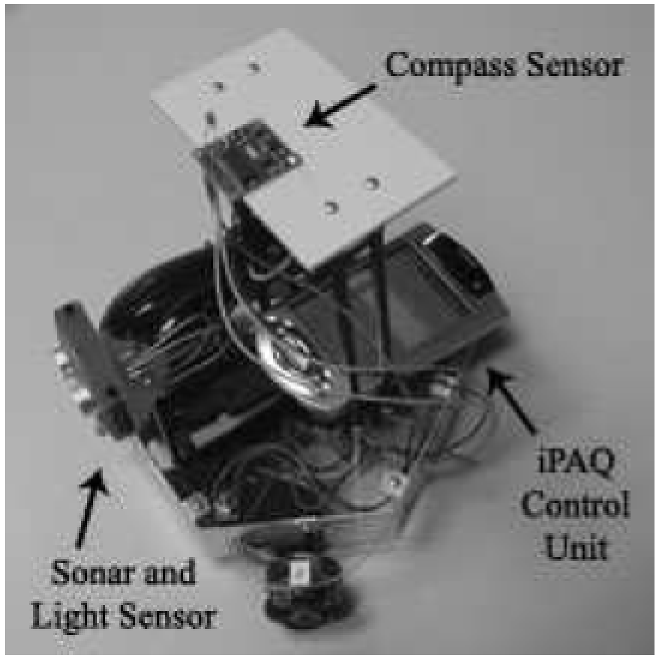

Figure 2.2 Modified PPRK

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 2.2).

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.

2.4 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.

2.4.1 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.

2.4.2 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.

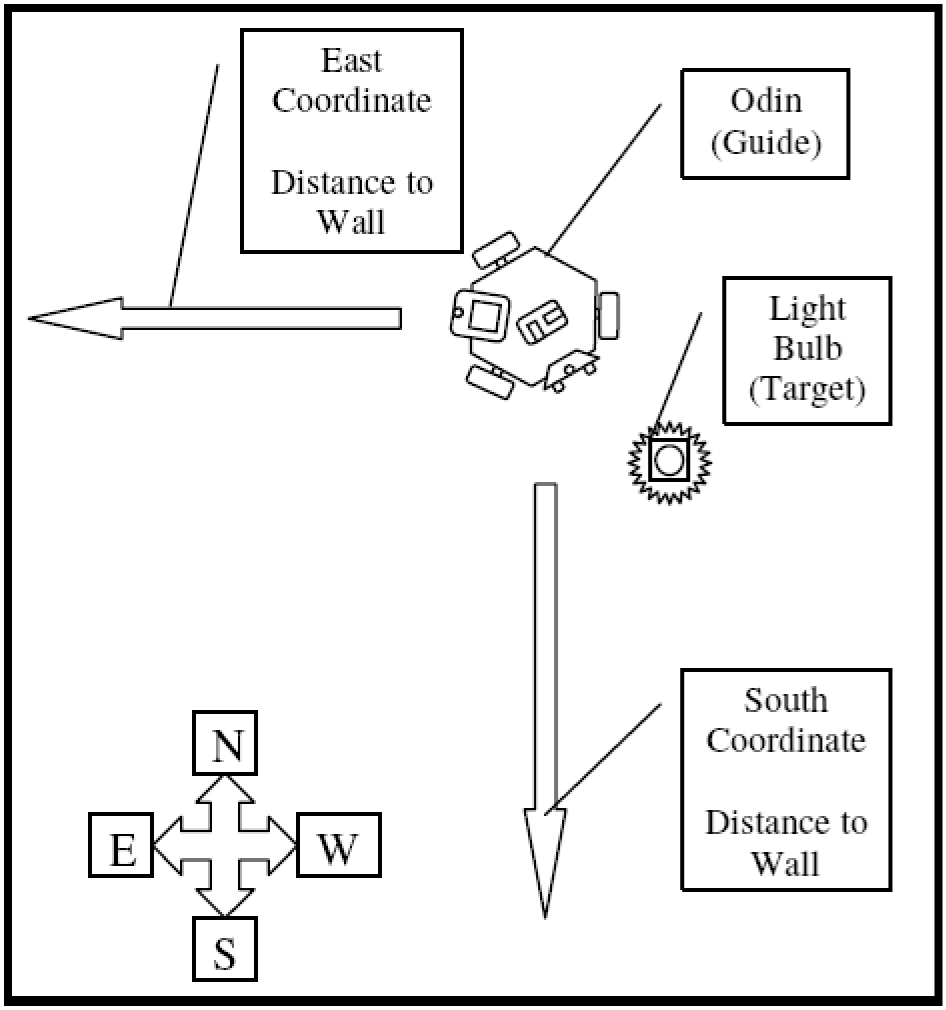

Figure 2.3 The Localization Process

2.4.3 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.

2.5 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 802.11x approach. In their design of a robotic 802.11x 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.

2.5.1 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 2.4 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.

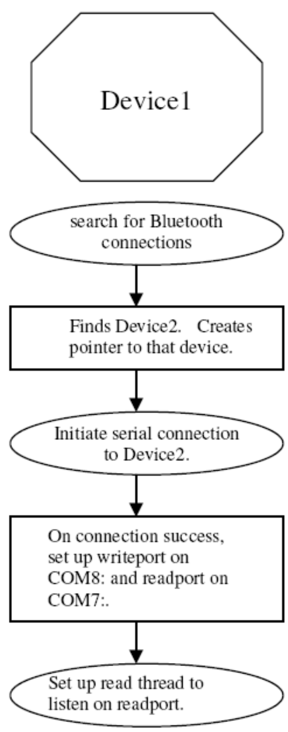

Figure 2.4 The Connection Process for One Device

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 801.11x 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.

2.5.2 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 2.5. 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.

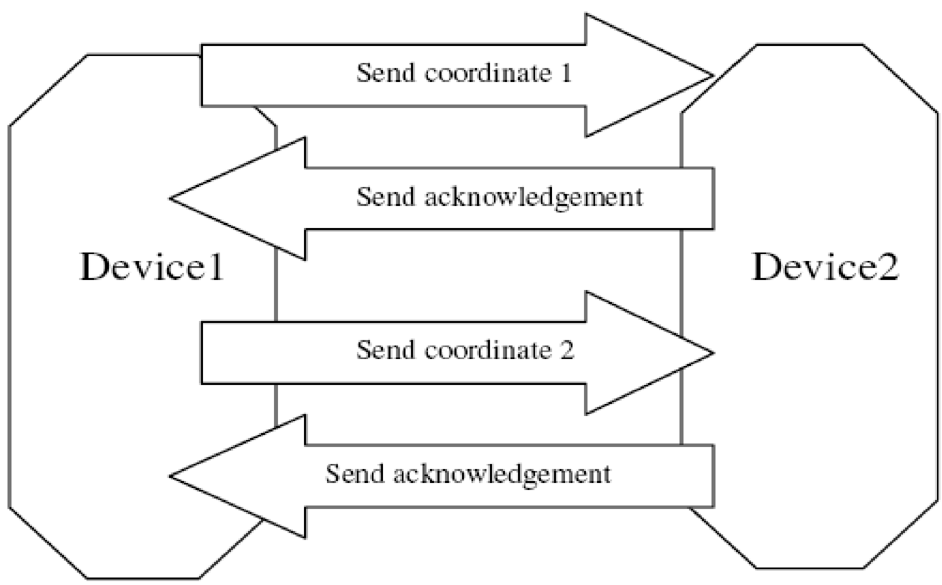

Figure 2.5 Coordinate Transmission Between Devices

2.6 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 2.6 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.

2.7 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.

CHAPTER 3

USING MULTIPLE COLLABORATIVE ROBOTS FOR TERRAIN MAPPING[2]

[2] To be submitted to the 8th IASTED Conference on Intelligent Systems and Control (ISC-2005)

3.0 INTRODUCTION

Intelligently distributing a complex task across multiple agents is a problem that has become a center piece of Artificial Intelligence. The term “agent” can refer to a number of different systems and implementations. There can be intelligent software agents that cooperate together to perform a complicated search task. Agents can also be defined in terms of actual hardware. In this case, robots represent the physical implementation of agents. There are several reasons that distributing a complex task to multiple robots represents an attractive area of current research. The current robotics platforms have limitations in speed, mobility and available processing power. An analogy can be made between a single robot and an ant. Just as a single ant cannot accomplish any great task, an entire colony all engaged towards a single purpose can be quite effective at achieving larger, more complex objectives. While a single robot can accomplish certain tasks, a group of robots collaborating together forms a much more effective force that is able to perform more complicated tasks.

3.1 BACKGROUND

Distributing a task across multiple agents is a problem that began to be addressed by Artificial Intelligence in the 1970s. Early work on this problem began by using blackboard systems that attempted to link together different agents across a network into a single problem solver [1]. Other early attempts at solving the problem of distributed systems used the idea of a contract-net protocol to link agents with problems to be solved with agents that have the ability to solve those problems [2]. The system necessitates the agents arrive at a mutually acceptable contract for how this work is to be done, but provides no specificity as to how to accomplish this.

The contract-net protocol method was left open ended, for example, how to bid on a solution and how to negotiate between the agents are all left to the individual implementations [3].

Eventually the notion of distributed artificial intelligence evolved into the Multi-Agent System. A Multi-Agent System (MAS) can be defined as a "loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver" [5, p80]. In general, a MAS can be thought of as containing the following characteristics [5]:

1) Each agent has incomplete information or capabilities for solving the problem and thus, has a limited viewpoint;
2) There is no global system control;
3) Data are decentralized;
4) Computation is asynchronous.

The use of multiple robots to solve a task in a decentralized manner is a form of a Multi-Agent System. There have been several successful attempts to implement robotic systems that use some of these ideas. Zlot et al. [6] attempted to solve this problem by implementing a form of the contract-net protocol by using a market economy for a multi-agent exploration strategy. This effort focused on individual agents maximizing their internal profit by entering into contracts with other agents to share information. This sharing of information results in profits to the individual agents which motivate them to acquire additional information.

Other approaches to this problem have implemented systems that use a multi-robot strategy coupled with a frontier-based search and bidding method. Simmons et al. [7] suggested that the individual robots create a frontier of cells that represent unknown areas that each robot then bids on to go explore. This method requires the use of a central bidding controller that assigns the winner the task to go complete. While these approaches represent valid advances in the field of robotic task distribution, our approach proposes a decentralized system that is based on reinforcements to motivate the emergence of robotic collaboration in a terrain mapping task.

3.2 DISTRIBUTIVE COOPERATIVE ROBOTS

The exact type of solution implemented to distribute robotic tasks is dependent upon the type of task and the environment that the robots are utilized in. In one scenario, for example, a group of robots might be tasked with determining where a particular object was within a known and discrete environment. If it was possible to allow for constant communications, it might be feasible to coordinate the robots activities and information through a centralized point. In this type of solution, a blackboard style architecture could be used to store all of the individually acquired data in a globally accessible structure. "In a robotic system, the blackboard [contents] could be seen as a representation of the world state, through sensor input, actuator positions, world maps, and other pertinent information [8. p.12]." If all of the robots had constant access to this information, then all of the robots would know the state of the world at all times. Correspondingly, they would all recognize, at the same time, when the object that they were seeking was found. This architecture has a number of benefits for performing tasks of this nature. Centrally controlling and coordinating all of the resources allows for a planning capacity that can reduce the amount of time to complete the given task. Increasing the efficiency in this manner is achieved by allowing all of the individuals to have access to the global store of information and providing the ability to easily decide how to allocate the resources to better accomplish their task. However, this architecture is not appropriate for all types of domains.

It may not always be possible to maintain continual communications with a central point in all environments. Additionally, in highly dynamic environments it may not be desirable to require constant coordination with a centralized point. In these situations, the need for a centralized control system could hinder the ability of a group of robots to complete their task or react quickly enough to their environment. One possible solution to this type of situation involves decentralizing the control structure of the individual robots. A decentralized control structure has many different potential implementations, but for this effort, the meaning is that there is no single point of control, coordination or global data storage inherent in the system. At first inspection, a system that operates as independent robots may not seem to have the ability to coordinate a complex task. There are several problems inherent in a decentralized control system that must be addressed prior to implementation.

The primary problem in a system that has no centralized and global method for data storage is determining how the individuals in the group assemble their acquired information into a concise package. In a centralized, blackboard style system, the overall global picture of the worldview is assembled at one point by combining all of the views into a single picture. As one agent gathers information, it can be dispersed to all of the agents within the range of the communication. This allows all of the agents that are connected with the centralized data store to instantly have updates to the global view. In a system that has no centralized location for data storage, sharing information between all of the agents can be a challenging problem. It relies on methods of communication that can send the appropriate data, and leaves it to the individual agent to assemble a consistent world view when updated and new information is available.

Within a mapping task that has a global coordinate system available, it is possible for each individual robot to allow their unique world view to be assembled into a global view as they come in contact with other robot's data. This is possible because individual chunks of mapping data can be assembled together by overlapping the known components into a larger map that each robot can benefit from (Figure 3.1). Once it is possible to merge individual maps, then there is the potential for each robot to construct an increasingly larger global view of the surroundings by interacting with other robots that have made different observations. It is possible to allow for this same type of functionality by using other systems as well. As long as the individual robots can assemble each other's information into a global overall view, it does not necessarily matter if there is a global coordinate system available (see next section). In a distributed decentralized robotic task, the interactions that each individual robot has with other robots are the key to developing a more complete world view.

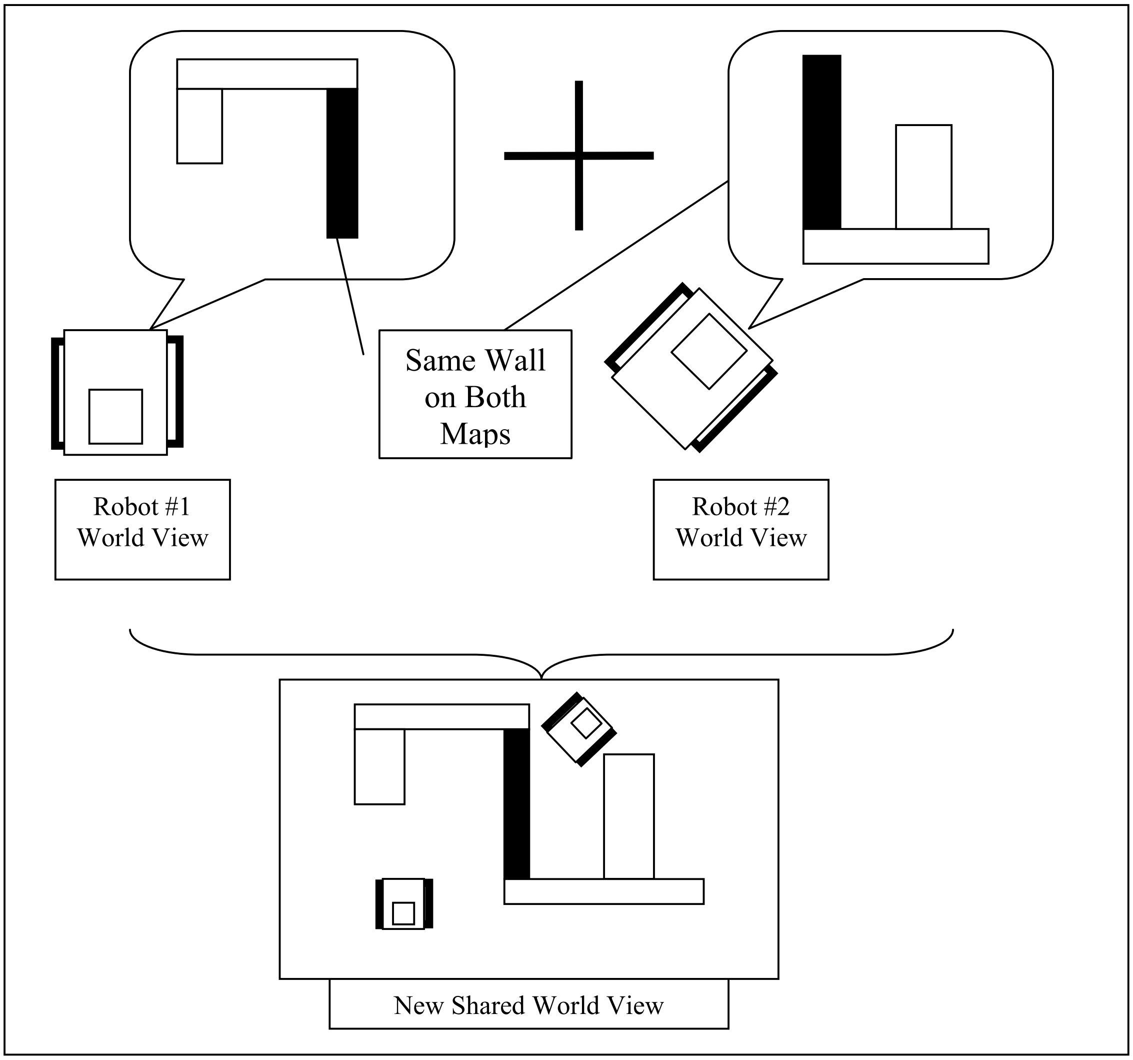

Figure 3.1 Knowledge Sharing

Regulating and controlling communications between individuals within a decentralized system is not relevant to this type of problem. The very nature of a decentralized system precludes the ability to directly control the communication within a group. So, alternatively, there must be a method that can be designed to allow a system with no direct control structures to exchange information and build a collective world view that is significantly better than the individual's view alone. Because the method of data exchange within a distributed decentralized system is based upon the direct interaction of the individuals within the system, it is logical to develop a method that encourages the single robots to communicate in a collaborative manner.

3.3 DESIGNING A COLLABORATIVE ROBOTIC SYSTEM

When designing a robotic system where there is no central control structure and data storage, there are many potential problems that can arise. Primarily, if the space under exploration is large enough relative to the communications range, the agents might be too distant to assure that one agent will ever be able to interact with another directly. However, while it is possible that the size of the environment might preclude robot communications altogether, this is an extremely unlikely possibility. Secondarily, even if the robots come in contact with one another, it may be difficult to form any solid conclusions about the level of task completion for the worldview as a whole. However, there can be attempts made to construct behaviors and frameworks that will make collaboration towards accomplishing a common goal much more likely to occur. By encouraging the individual robots to communicate, more information is shared between each robot. The encouragement of each robot to communicate with the others is a type of collaborative behavior. It is reasonable to assume that given enough time and communications, the entire environment could eventually be explored and shared with all of the robots within the system.

The primary motivation behind the system that will be proposed in this research is to create a framework that encourages a reinforcement emergent behavior that results in increased collaboration between individual robots. This is to be accomplished by providing each robot with a reward system that serves as the motivational force to accomplish the desired action. It is expected that the individual robots will learn and perform behaviors that serve to maximize the amount of reward received while simultaneously accomplishing their assigned task. In this particular case, a terrain mapping example performed by multiple robots will be used to exemplify this process.

While it is true that the principles that are constructed in this project can be applied to other problem domains, it is necessary to specify a more exact description of an individual problem to facilitate an understanding of the solution. In this scenario, there are at least four autonomous robots that have adequate sensory capabilities to sense and record observations about their environment. Another assumption is that these robots have the ability to intelligently navigate through their environment by having obstacle avoidance methods already designed. Each robot also has access to a global coordinate system that is used for navigation and mapping recorded sensor data. While a global coordinate system is not explicitly required, it facilitates the data sharing between robots and makes this example easier to explain. Other methods of localization and recording are possible; this is just one method to accomplish those tasks in a simple, concise manner.

The task that this group of robots is designed to complete is a distributed mapping exercise. This requires the robots to go out and explore the environment and record their results individually. As the results are recorded locally in each robot's individual memory, it must be possible for each robot to share their results in order to increase the collective awareness of the global state of information. So, it must also be assumed that the robots have the ability to distinguish one another and establish communications on an ad-hoc basis. This means that when one of the robots comes into range of another, they would both be able to understand which robot it is and be able to communicate information in a bi-directional manner.

In order to facilitate the sharing of information a method needs to be developed to help guide this communication. If the problem were to be abstracted to a more human level of cooperative task distribution it might become easier to understand the robotic solution that this work proposes. If a group of people were asked to explore an unknown area with only simplistic measuring devices and radio communications, it would be instructive to describe how the solution might evolve. When the explorers first begin their process they record their observations wherever they wander. As they make contact with other explorers, either over the radio or in person, they give descriptions of where they are and what they have found. These observations are then recorded by the receiving party, and then the receiving party attempts to share any unknown knowledge with the other explorer. Each of the explorers is not in constant radio communications with all the others because the radio's range is not sufficient to guarantee constant contact, but whenever they can talk to another person they try to. Here there is no global data store for all of the explorer's observations, but it is reasonable to assume that as each of the explorers record more information and share it with others, each person has a better understanding of the global picture of the world around them. They accomplish this task through their random communications whenever there is another person in range, so they attempt to synchronize all of their world views whenever they can.

In the same manner as with the humans, a mapping scheme can be distributed to a group of robots. However, in the case of the robots, to create the necessary framework, more formalization of the method of task distribution is required to design the robotic behaviors. The behaviors are designed around the central premise that every individual within the group should collaborate in order to effectively map an unknown area. Just as the human explorers in the scenario described above, the robots will need to be given the desire to perform two types of actions. First, they must be equipped to go and explore the environment in order to make the observations necessary to complete their task. Second, they must be sufficiently motivated, by using reinforcements of some kind, to want to share their individual observations, in a collaborative manner, with the other members of their group. However, the critical part of developing this framework is designing an implementation that seeks to balance these two types of actions, exploration and collaboration, into a system that can effectively map a large unknown area.

The method that is being proposed within this work to help balance the exploratory behavior with the collaborative behavior is the creation of a reward system that is based on information credits that are granted for the different actions that each robot can perform. The method of distribution of these credits will be detailed later; however, currently it is sufficient to understand that the information credits are artificial monetary units that are granted as rewards to each robot for specific actions performed. The desire of each robot to earn these information credits is motivated by a survival pressure tax that is created by decrementing the amount of credits per unit of time each individual has and simultaneously requiring that each robot have credits to maintain itself in an "active" state. The survival pressure tax is an artificial force within the group that takes away some of the credits that each individual robot has. Once all of the credits of a robot have been eliminated, the robot is considered to enter an "inactive" state for that particular mapping task. It is possible that if a new mapping task is desired, the robot can be re-activated to perform a subsequent task.

One method by which a robot can earn information credits is by exploring the environment. An assumption that will be true through the remainder of this paper is that the environment can be divided into discrete units, or blocks that the robot can differentiate between.

The size and shape of the blocks that are created might vary for each individual robot, but are directly related to the sensory capabilities of the platform chosen for implementation. For example, if the chosen robotics platform had four sonar sensors directed in every cardinal direction from the chassis, and each had a reliable range of 18 inches, then the block size of each unit of measure would be approximately 36 inches by 36 inches (Figure 3.2). However, as the chosen robotics platform sensors might be different from the example above, it is necessary to create a general rule governing the creation of the internal view of the environment that each robot has. This general rule states that the size and shape of the blocks used can be no larger than what can be accurately measured by the sensory equipment and recorded on board each robot and that the size and shape of the discrete block must be the same for all of the robots.

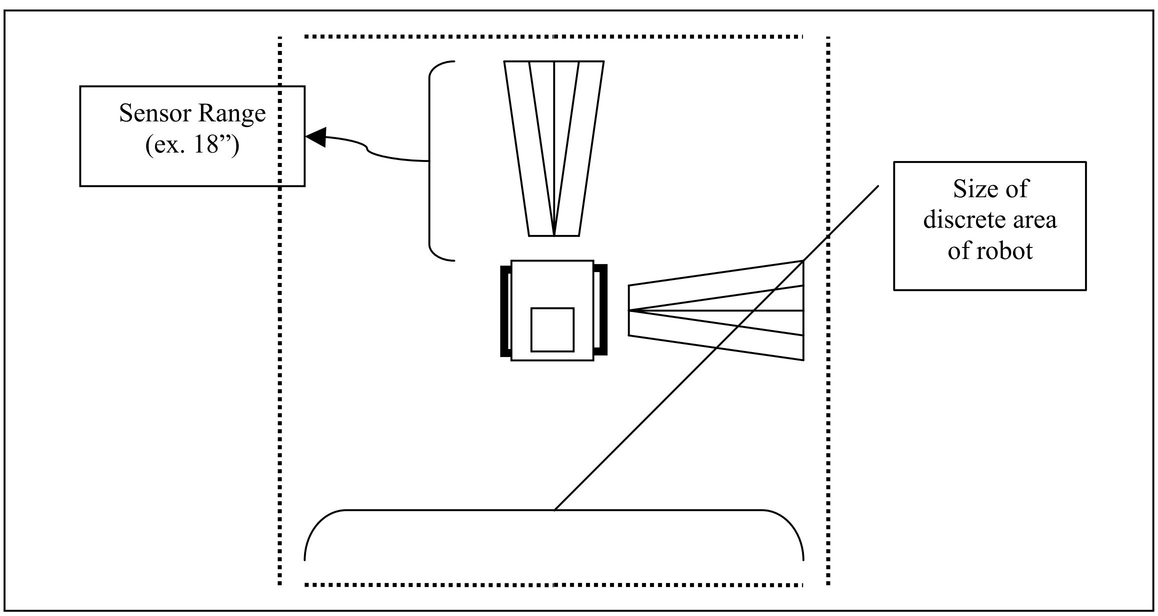

Figure 3.2 Sensor Area

It is necessary to make this assumption about creating a discrete environment because there are two essential components that rely on this premise. First, the ability to share information is made much simpler if each robot can communicate square blocks of data to correlate sensory information. So, the communications component relies on the ability to make the environment discrete. Secondarily, each robot is granted credits per block of explored territory. If a determination of the boundaries of each block of territory cannot be made, then it is impossible to assign credits to each individual. In this case, the information crediting component also relies on the discrete environment.

There are two separate cases for exploratory credits that need to be differentiated:

- The first case is when a robot is exploring a new area that it has never been in before. This is a very important operation because it yields the most valuable information to the individual, and is the method by which each robot accomplishes the given task. So, the action of discovering a previously uncharted area and the successful mapping of this area, for example, yields 20 information credits per block explored. The amount of credits assigned to each robot per block of area explored is an artificial number and is unimportant. When designing the actual system, the exact numbers themselves are not as important as the relationship and prioritization of how they are assigned.
- The second case that is relevant to the exploration behavior is moving through a block that has been previously explored, but realizing that the status of the block has changed in some manner. For example, as a robot moves through a dynamic environment, it is reasonable to assume that there are objects that will have moved around. Because the robots themselves have the ability to understand which block they are currently located within, it is possible to calculate the differences that have occurred since the block was previously explored. The ability to calculate these differences in positions requires that all of the recorded sensory data have a timestamp attributed to it. This allows the possibility to compare and differentiate older events with more recent discoveries. This

type of information is also extremely valuable to the individual and the overall task. However, the value of the information is directly related to how much the area has actually changed. If there is an object in a block that was mapped before, and that object has only slightly changed position, this is not as valuable as ten objects moving within the same area. Also, it is possible for objects to leave the area that they were first discovered in. In that case, the amount of credits awarded still relates to the percent difference in any particular block. So, as such, the value for the information must also be related to the amount of change in the given area[3]. In this case, the amount of information credits given to a robot for exploration of a previously known area is given as follows:

Information Credits = integer value[4] of (N% (range of 0 – 1) * 100)

The ability to calculate the credits is predicated on the assumption that the robots will be able to compare the sensor picture that they currently have with the one that they did have before and extract the percentage that is different. However, given the above equation, suppose that the robot has noticed that the area it is currently in is 13% different now than when it previously explored the area. In that case, 1.3 information credits are earned, but since, in this example, credits can only be granted in whole values, only 1 information credit is granted to the robot. In the previous example, because a non integer value was generated, a rounding of the numbers is done to assure an integer result. If the value is at or above 0.5, then the credit is rounded up, otherwise it is rounded down. If the entire contents of the area have changed, the maximum value granted would be 10 credits. Once again, the actual numbers that are used here are to serve

[3] The method described to calculate the change in a block might require further experimental refinement.
[4] The use of integers is not required in this system, but was only used to more easily illustrate this system.

as a guideline for the actual implementation, the relevant part is the relationship between the values. By granting credits for area explored by the individuals, and by linking those credits to the survival pressure tax that will be discussed later, it is reasonable to assume that the individuals will be motivated to explore their environment.

The second type of behavior that is imperative for a distributed task to work is the collaboration and communication of the data that has been acquired by all of the other robots. In this case, it is a desired attribute for the robots to be extremely cooperative with one another; that is, they should share data on a regular basis to improve the overall understanding of the environment. To accomplish this increased level of collaboration, there is a method that needs to be developed to broker the transaction of data between two individuals. As the individuals come into contact with one another, they can begin the process of sharing information. In order for this behavior to become important to the individuals, a process similar to the one above must be defined for the valuation of the information being shared. Since each robot has access to a global coordinate system or has another method to standardize their data into shareable knowledge, it is a simple matter for the individual robots to determine if they have different data to exchange. The robots will then initiate a swap of all of the unique information that they have about the world.

For any given block of recorded sensor data, there are three possible outcomes for the exchange of shared information:

- First, the block that is being shared is new and previously unmapped by the individual. In the case of a previously unknown block being shared, both robots will benefit greatly from the exchange. Both robots receive 1.5 times the value for an unexplored block by initiating the exchange. This would be the case when both robots make contact and there is a determination made that one of the robots does not contain any information about a particular block. The robot that has that information transfers the data for that block to the other robot. In this case, both robots will receive 30 information credits for the exchange. This number is derived from the unexplored block credit of 20 credits listed above multiplied by a factor of 1.5, which is the shared data value multiplier. This gives both robots equal incentive to both give and receive information.

- There is a second case that results from this information exchange process, where both robots contain information for the same block of data. Suppose that this information varies in its contents, resulting from an object being moved in that block. In that case, the timestamp that the robot set when that information was recorded becomes important. The information with the most recent timestamp is considered the most valid and overwrites the data on the robot with the older timestamp. The credits that are granted for this exchange are derived in the same manner as the credits for an updated session above. For example, if there is a 20 percent difference between the older recording of the block and the newer recording of the block, then 2 information credits are granted to both robots. However, because this exchange is a result of a sharing behavior, and these activities are very important, a multiplier of 1.5 is granted to the exchange to encourage the sharing. So, in that situation, 3 credits are given to each robot for updating each other's information. This is the result of the product of the 2 information credits by the 1.5 multiplier for this being a shared action. If there is a decimal formed as the result of the multiplication then the rounding described above occurs to keep all credit values as integers.

- The third case is that the robots have the same information about a particular block. Here no credits are exchanged, and nothing is updated. In this situation, the robot has taken time when it could have been exploring and gaining credits to attempt to exchange information with another robot. This results in the robot taking the risk of a negative reward because of the credits that are being deducted by the survival tax while no new credits are being earned. While this could result in the robots not wanting to share information because of the risk associated with not earning credits for new information, it is expected that a different behavior will emerge. Instead, it is assumed that in the long run, the robots will indirectly learn that taking a small risk is still better than not taking any risk at all.

In general though, by granting more information credits for exchanging new and updated knowledge between robots, it is expected that this will encourage the robots to be more collaborative, which will result in better dispersal of the individual's data to the rest of the group.

In order to understand the motivation that each robot has to seek these information credits, it is important to describe the survival pressure tax that is inherent in this system. The survival pressure tax is a regulated deduction of information credits per unit of time. So, suppose that in a particular task, the survival pressure tax is regulated at 20 credits per minute. With the rates of information credits given above, that would mean, that on average an individual robot would need to discover and record one new block of information per minute to maintain its "active" state. Of course, each robot would need to be given a certain amount of credits when the task is started to allow them time to disperse through the environment. The number of credits that are granted at the beginning of the task would be dependent on the type of task, and various other task related issues.

The notion of survival as related to the amount of sharing and discovery performed is important for two distinct reasons. First, it gives each robot the reinforcement to explore and share the information within the given environment. But, the survival pressure tax also serves a secondary, equally important purpose. It provides the stopping criteria for an otherwise uncontrolled distributed system of robots. Suppose that there have been no new discoveries in a given environment and all of the sharing that was possible for a given task has been already done. That means that there are no new information credits being given within the environment. No new information credits being given will ultimately result in all of the robots having an "inactive" state for that particular task. This is a desired quality because there is no need for a robot to continue to explore an environment that is mapped and unchanging. So in the case that the robots are all inactive, that particular mapping task is considered closed. It might be the case that as time continues, it may be useful to start another mapping task as the environment or objects within it have changed. After one task is closed, it is always possible for the robots to initiate another task if it is needed.

The survival pressure tax can also be used to regulate the urgency of behavior for the robots. For example, suppose that there was a building that has collapsed, and there is a great need for quick dispersal and recovery of information from the scene. In that case, by increasing the amount of information credits per minute deducted the robots are motivated to move faster through their environment and more quickly share their data to continue to maintain an "active" state. This urgency in an environment is not always advantageous. Suppose that there was an extremely hazardous environment that the group of robots was sent to explore. In that case, the survival pressure might be reduced to allow each robot to take more time in exploring an environment to avoid becoming disabled. The exact nature and rate of decline is then variable depending on the situation being explored.

3.4 PROPOSED ROBOT ARCHITECTURE

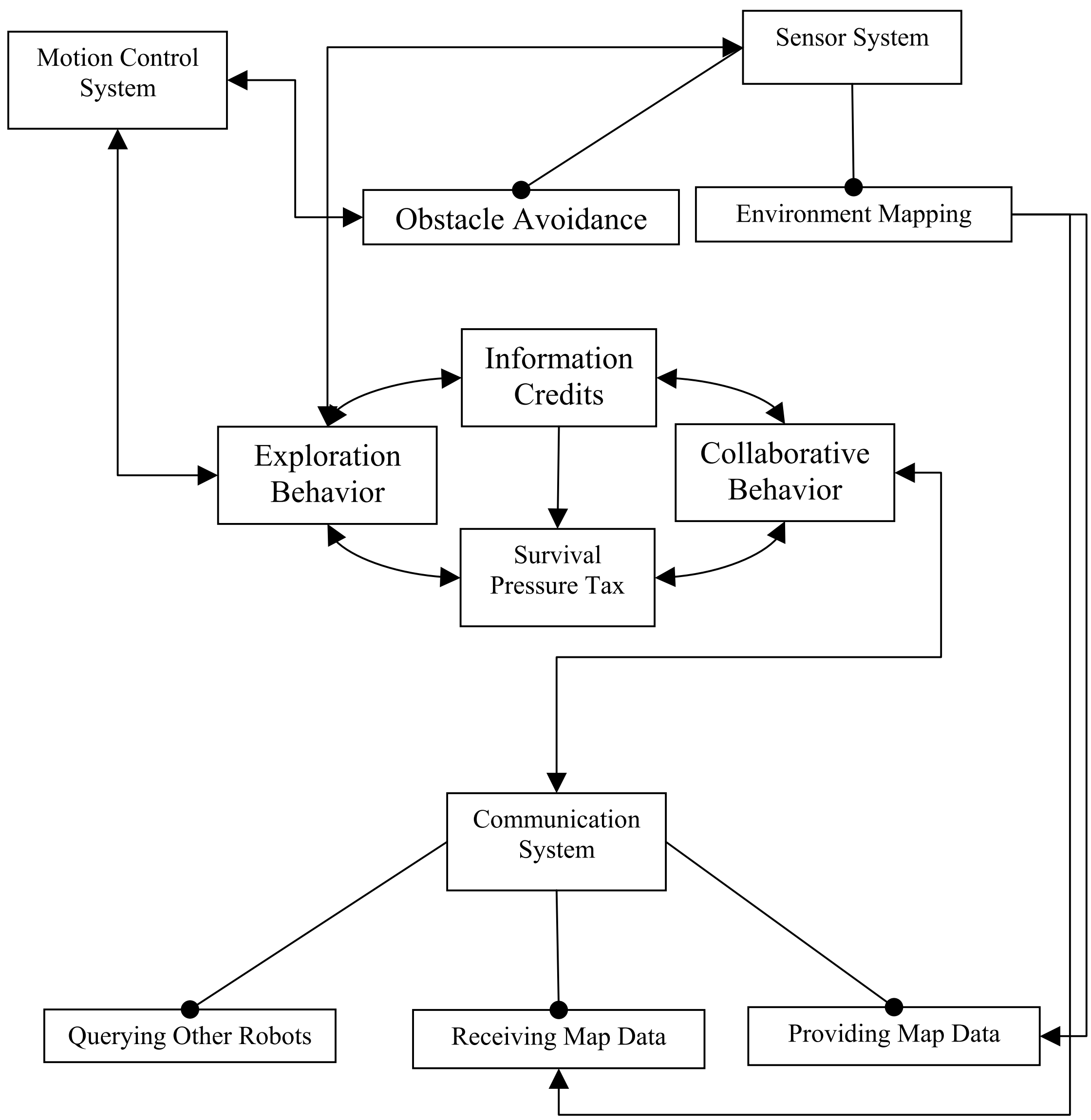

Figure 3.3 Robot Architecture

The architecture above (Figure 3.3) demonstrates the individual robot architecture described on an abstract level. Each robot will have three major systems on board:

1. Motion Control System: This system is responsible for moving the robot through the environment. The Motor Control System is directly connected to the Exploration Behavior Module because it is the mechanism by which the robot is physically able to move and discover the environment. There is also a direct connection between the Motor Control System and the Sensor system. This is required because in order to map the environment and avoid obstacles, coordination between these two systems is needed. For example, in the case of obstacle avoidance, if the robot's Sensor system recognizes an obstruction, it would provide an input to the Motor Control to move around that obstruction.
2. Sensor System: This system is responsible for coordinating all of the sensory capabilities on the robot platform. While this may include a variety of sensors, depending on the exact robot being used, this system is responsible for coordinating these inputs into a coherent, consistent view of the world. This view of the world is utilized by the Sensor system to provide two higher level functionalities.
    a. Environment Mapping: This module utilizes of all of the sensory inputs to provide the robot with the ability to accurately map and record the environment. Inherent in this functionality is the ability to localize the robot's current position and provide a timestamp because these are attributes needed to build a precise map of the world.
    b. Obstacle Avoidance: As the module would imply, this mechanism is required to utilize all of the robot's sensors to avoid obstacles in the environment. The exact

implementation of this function is unimportant, so long as the robot has the ability to avoid obstacles that are encumbrances within the environment. Also, because the result of avoiding an obstacle requires the robot to move in some manner, this module is also directly linked to the Motor Control System.

3. Communication System: This system is responsible for coordinating all of the robot's ability to make contact with other robots in the environment. Once again, the exact type of communication (Bluetooth, 802.11x, etc.) is unimportant, rather it is a description of a higher level system needed to coordinate and control how a robot talks with other robots.
    a. Providing Map Data: The primary purpose of this project is encouraging robots to share the information they have acquired. In this particular example, the robots are required to map an environment; so correspondingly, the type of information being shared is map data. However, this could easily refer to the type of data acquired in any other type of task. This module refers to the ability of the robot to make a connection through the Communication System and share the map data it has acquired with another robot.
    b. Querying Other Robots: It is not always the case that the robots that have made contact will actually have new information to share. This module refers to the ability of the individual robots to query other robots through the Communication System to determine if another robot has information to share.
    c. Receiving Map Data: If a robot has the ability to provide map data to another robot, it must also be able to receive this information. This refers to the ability of the individual robot to receive and integrate new map data from other robots through the communication system.

In addition to the major systems described above, there is another layer of functionality present on each of the robots. While the systems above refer to actions and physical systems on each robot, there is another layer of the architecture that provides the ability by which the robots can coordinate their efforts through the use of two different types of behaviors and the mechanisms that are used to balance those behaviors.

1. Exploration Behavior: This is a broad term used to describe the robot's ability to explore the environment. This behavior describes the need for the robot to discover and record the environment. Because the proposed architecture rewards robots for exploration, this module refers to the method by which each individual can explore new areas and record information. In order to explore the environment, this behavior also must be linked to the Motor Control System. This link is needed because it represents how the robot actually moves through the environment. This topic is linked to both the information credits, which function as the reward mechanism, and the survival pressure tax, which reduces the amount of information credits.
    a. Information Credits: This module is the main motivational force that rewards the robots for the various positive actions that they can perform. For example, in the diagram, the information credits module is linked to both the providing and receiving of map information. This is because both of these actions result in more information credits being earned by the robot. Additionally, this heading is linked to the survival pressure tax described below because this tax reduces the total amount of information credits that each robot has.
    b. Survival Pressure Tax: As described above, this module refers to the action that reduces the amount of information credits that a robot has. In the proposed

architecture there is a constant reduction of information credits per unit of time. This value is variable, and can vary depending on the type of environment the robots are exploring, and the urgency that they need to explore it with. A more detailed explanation about determining the rate of the survival pressure tax is provided elsewhere in the text. However, because the survival pressure tax can determine the urgency and method that the robots use to explore their environment, it can directly influence the balance between the exploration and collaborative behaviors.

2. Collaborative Behavior: The primary purpose of this research is to encourage multiple robots to collaborate and share the information that they have collected individually with the entire group. The mechanism by which this is accomplished is represented by the collaborative behavior. This behavior is directly influenced by the ability of the robot to be able to communicate the desired information. For example, because it is important to have the robots share their map data, it is required that this behavior be linked to the Communication System, the mechanism by which this sharing occurs. This behavior is also directly influenced by the information credits being provided to the individual robot, as well as the rate of the survival pressure tax.

3.5 REINFORCEMENT BASED EMERGENT BEHAVIOR

The architecture serves as the basis for the following example of how the code for the individual robots might be designed. In the example, one possible implementation is outlined for how best to design the ideas presented above. It is expected that this overall structure best prioritizes the individual robot’s behavior and actions, but fine tuning of this example is best left for the actual implementation. As the robot begins the task of mapping an unknown area and collaborating with other robots, it is important to remember that the actions of the robot are designed to maximize the amount of information credits received. Since there is a constant survival pressure tax, the individual robots need to generate the most information credits possible for any given situation. The following is an example of how an individual robot might operate:

1. **Query the Communication System to determine if there is a robot in range:** If there is a robot in range, and either of them have information to share, then that is the most desired action because it results in the most information credits being earned by both robots.
    - **a. If there is a robot in range then establish a connection with that robot (Robot2).** (Robot1 is the robot being described here)
        - i. **Transmit Coordinates and Timestamps** of all data for Robot1 (Instead of sending the entire block of data over the communication, the robots first send just the coordinates and the timestamps. This allows for a quicker comparison in less time than sending the entire block of data.)
        - ii. **Receive Coordinates and Timestamps** for all data for Robot2
        - **iii. Compare data to determine if there is any new or updated information to exchange.**
            1. **If there is information to update then send all of the blocks of data for the coordinates that are required. (**This results in earned information credits for the transaction.)

2. **If there is no information to update, then disconnect. (**Since there is a survival tax being constantly assessed the robots want to minimize the amount of time performing activities that do not result in information credits being granted.)

2. **There is no robot in Range, so Explore the environment**

    a. **Explore unknown areas first.** (This action results in the highest reward of information credits possible, and therefore is the most desirable action.)

    b. **Explore known areas for changes. (**This action results in a proportionate amount of information credits being earned for the amount of change in a given area. This action is also worth information credits, but not as much as for unknown areas. If there are no changes in a block, then no information credits are earned.)

So, the example provided above is useful for a basic understanding of how an individual robot's architecture might be structured. However, the most important part of this project is to provide examples of how the structure of information credits and survival pressures ultimately can result in a reinforcement emergent behavior. The most important motivation that each robot has is their continual requirement to earn information credits to maintain their "active" status. Emergent behavior can result from the robot's attempt to maximize the amount of information credits by actively balancing the exploration behavior with the collaborative behavior. An example of how this might occur will serve to provide some insight into this process.

Suppose that a robot has a condition where the available information credits are considered to be low. A low condition might be variable depending on the situation, but in this case, with the current rate of the survival tax and assuming no new credits are earned the robot would be active for another two minutes. In those two minutes, the robot is very interested in gaining credits for any action taken, but the behavior that it demonstrates, exploration versus collaboration, depends on the conditions that surround the robot. For example, if the robot is in close proximity to an unexplored area of its map, and there is no robot currently in range, then it is less risky for the robot to explore that region than it is for it to attempt to find another robot to share information with. This is despite the fact that sharing information, if it could be accomplished in the given time frame, would be more valuable. It is expected that emergent behavior would be the result of the reinforcement generated by the information credits. This reinforcement could drive a machine learning component that could influence the types and frequency of the behaviors that the individual robots exhibit. So, the intention of the reinforcement system is to allow for the emergence of robot behavior that will both maximize their individual rewards while also efficiently accomplishing the task that the group has been assigned.

3.6 REAL WORLD ISSUES

In general there are several issues that can affect the real world implementation of any robotic system. First, the ability of a robot to move through the environment is a challenging problem. With most robot drive trains, whether it is wheeled, tracked, or uses some other method of locomotion, there are problems associated with enabling a robot to reliably move through the environment. For example, on tracked robots, a method called differential steering is often used to enable this type of vehicle to turn. This method requires a certain amount of slippage that is needed to make the robot turn. However, the exact amount of slippage is dependent on the traction of the surface that the robot is being operated on. So, if this robot is known to take 15 seconds to turn 90 degrees on carpeting, it might only take 10 seconds to make the same turn on a tiled surface. If the robot depends on the amount of time that each motor is running to judge the amount that the robot has turned, then it no longer becomes possible to be certain what direction the robot is headed and how far it has turned. This is problematic for systems that require this information to build and record information about the environment, because if it is not possible to know the current location of the robot, it may not be possible to map anything reliably.

Problems with the sensors on board robots also make it difficult to accurately map certain areas. For example, sonar sensors that are used for distance measurements are quite common on robotic platforms. However, these sensors can have difficulty mapping objects that are positioned at angles that are not perpendicular to the sensor. So, as a robot is mapping a corner of a maze, the distance measurements that the robot is recording as it navigates around the corner might not be accurate. If two robots map the same area from different angles using sonar sensors, it might be possible to generate two different maps as a result of these types of errors in the sensors. So, in a system that relies on the data for any given block being accurate and comparable from robot to robot, in the real world this can become problematic due to issues with the sensors.

The communication systems that the robots use can also be problematic. In reality, when two robots attempt to establish communications, there are quite frequently problems that may result in incomplete transmissions or premature loss of connections. This loss of connection can be caused by several different factors. Perhaps the robot is just at the range that communications are possible, and this causes an intermittent loss in the transmission. It might also be the case that a large object has moved within the environment that has blocked the robots ability to communicate. This lack of reliability with respect to communications can be problematic in a system that relies on the ability to exchange information as the primary method to complete the given task.

3.7 CURRENT IMPLEMENTATION

To effectively deploy a distributed robotics scheme based upon the ideas above a platform must be used that will allow for the correct level of functionality to both explore the environment, as well as provide the infrastructure required to implement the behaviors addressed in the previous sections. The robots chosen must demonstrate two key proficiencies:

- The ability to move through the environment and gather sensory data is primary to the task of robotic mapping.
- The ability of the robot to communicate and process the information being shared between the individual robots.

All of the interactions among the robots rely on the ability to communicate, so it is important that the communication system be both fast and reliable. As discussed above, the reliability of the communications systems for robots can be problematic. Additionally, since the robots will be moving through unknown environments it is also important that the communications scheme be able to function in an ad-hoc manner. For example, as one robot moves into range, it must have the ability to discover other robots, on the fly, within its current range. Since the group of robots could be dispersed over a large geographic area, they must have the ability to establish communications as they become available.

The Bluetooth protocol meets the requirements for communications for this implementation. Using this protocol, as an individual robot explores a selected area it is possible for it to continually query to discover if there are any other robots that are in the same vicinity. If an individual robot is able to discover a connection, then it is also possible to communicate on the fly with any other robot in range (Figure 3.4).

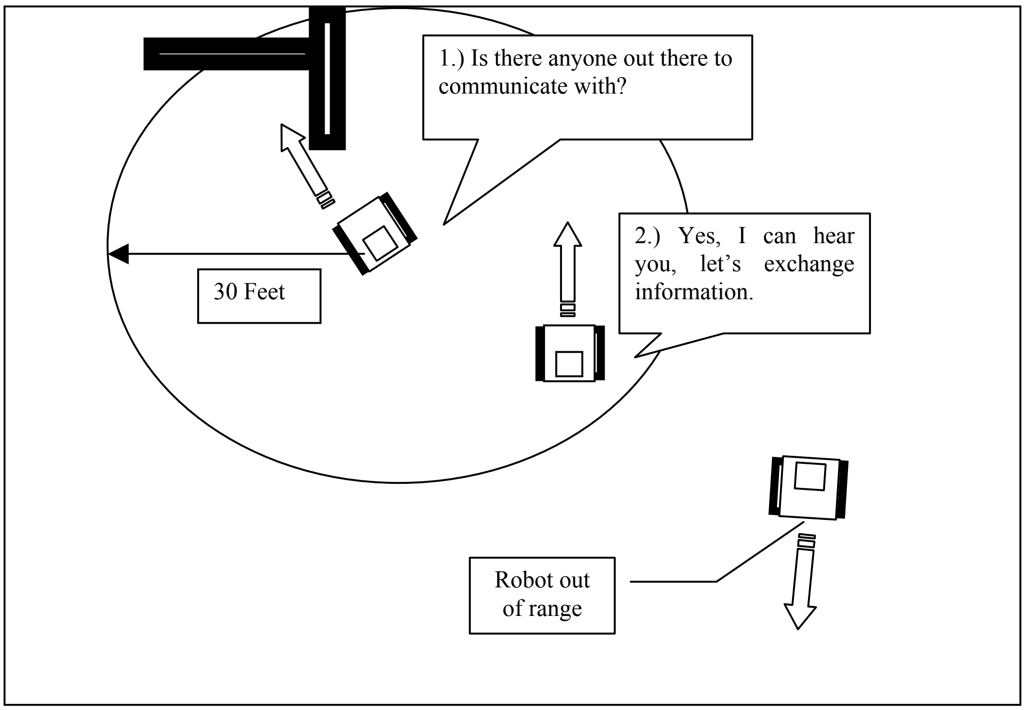

Figure 3.4 Ad-Hoc Communications

The tank-based robots that were designed for a previous project have sufficient flexibility to allow for their reuse in the implementation of this project as well [9]. This chassis has a handheld computer onboard that has already demonstrated the ability to record mapping data in sufficient detail for this project. Additionally, the handhelds each have a Bluetooth radio on board that allows them to discover and communicate with the other robots. Each robot has the ability to move through a dynamic environment because of the mobility and sensor functionality provided by the tank chassis. Each robot is built with four directional sonar sensors with integrated light sensing capabilities. This allows each robot to use their sensors to map each cardinal direction without the need to rotate to take individual measurements. On each robot there is also an onboard compass that allows the robot to continue to maintain directional bearings while on the move, addressing the concern above regarding tracked robotic motion. This compass can be used to provide localization by using a process that takes two sonar distance measurements at two cardinal directions separated by 90 degrees. The environment that the robots operated in had very few obstacles, and relied on the distance measurements to provide an X and Y position within the environment (Figure 3.5) [10].

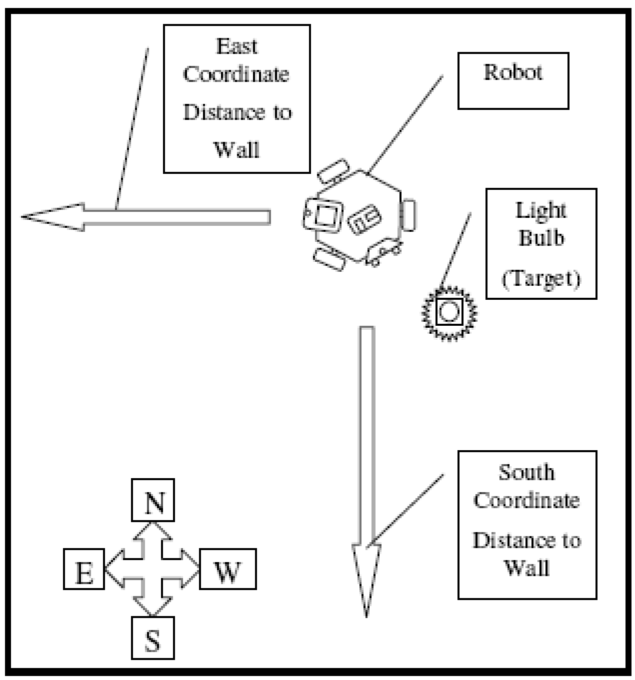

Figure 3.5 Current Robot Localization Process

Currently, only two of these robots have been built, but to create a collaborative group will require the completion of at least two more robots. This should be a sufficient number to demonstrate the behaviors required to validate the premises of using collaborative robotics for a distributed mapping task. Additionally, in order to use the robots that have been built, it would be very helpful to have a global positioning system in place. Currently, the robots use a combination of their sensors and compass headings to localize their position within their environment. In environments where there are many obstacles, this localization procedure has difficulty providing position information to the robots. This is because the current scheme requires that a robot have a clear view of two walls in two cardinal directions separated by 90 degrees. With obstacles in the environment, it is not always possible to have a clear view of walls, which results in the robot's inability to get an accurate position. So, it would be very beneficial to develop a localization procedure that is not affected by obstacles. The use of a global positioning system is not a requirement, but a suggestion that would aid these robots in being able to accurately determine their location.

3.8 CONCLUSION

Through the creation of the information credit system as a method of reinforcement, it is expected that the robots behavior will emerge to better accomplish the task of distributed exploration. It is also expected that the robots will develop an emergent behavior that will balance between an exploratory behavior and a collaborative behavior by using the system described above. However, the task does not necessarily need to be a distributed mapping exercise. This concept can be applied with a broader basis and could be useful in a variety of robotic tasks. In general, the modifications necessary to adapt this model to a different domain include understanding what is of value for that particular domain. For example, in our scenario, new map data were valuable, so the information credits were structured around that key attribute. However, there are other domains where robots are useful such as a military search and rescue mission. Perhaps in that domain, the amount of injured people found and returned to receive medical care could be the motivating factor. The credits would be based on something entirely different than mapping information, but the basic notion behind the concept remains. In the medical causality case, the information being exchanged might relate to where more individuals could be found to be rescued. The exact nature of the mission is not as important as the basis for achieving its success. In this case, the motivation is to create a system that is both distributed, and decentralized, but also one that creates a framework where cooperation between individuals is an important factor. By creating a system that relies on the collaborative communication of knowledge, a robust robotic framework can be created that is resistant to any single point of failure and can effectively complete a distributed task.

3.9 REFERENCES

[1]Edmund H. Durfee and Jeffrey S. Rosenschein. Distributed Problem Solving and Multi-Agent Systems: Comparisons and Examples. *In Proceedings of the Thirteenth International Distributed Artificial Intelligence Workshop*, pages 94-104, July 1994.

[2] Sandholm, T. An Implementation of the Contract Net Protocol Based on Marginal Cost Calculations. *In Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI)*, 256-262. Menlo Park. Calif.: American Association for Artificial Intelligence, 1993.

[3] S. Sen, "Multiagent systems: Milestones and new horizons." *Trends in Cognitive Sciences*, December 1997, vol. 1, no. 9, pp. 334-340(7)

[4] Durfee, E. H., and Lesser, V. Negotiating Task Decomposition and Allocation Using Partial Global Planning. *In Distributed Artificial Intelligence, Volume 2, eds. L Gasser and M. Huhns*. 229-244. San Francisco, Calif.: Morgan Kaufmann, 1989.

[5] Multiagent Systems. Katia P. Sycara AI magazine Volume 19, No.2 Intelligent Agents Summer 1998.

[6] R.M. Zlot, A. Stentz, M.B. Dias, and S. Thayer, "Multi-Robot Exploration Controlled By A Market Economy," *IEEE International Conference on Robotics and Automation*, May, 2002.

[7] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, S. Thrun, and H. Younes. Coordination for Multi-robot Exploration and Mapping. *In Proceedings of the National Conference on Artificial Intelligence, AAAI*, 2000.

[8] McClain, Jon. "A Behavior-Based Blackboard Architecture for Multi-Robot Control", Thesis, University of Georgia, Athens, Georgia, 2004.

[9] "Distributed Robotic Target Acquisition Using BlueTooth Communication," by J.T. McClain, D.H. Barnhard, B.J. Wimpey, and W.D. Potter. *In the Proceedings of the 42nd Annual ACM Southeast Conference,* pp.291-296, Huntsville, Alabama, April, 2004.

[10] D.H. Barnhard, J.T. McClain, B.J. Wimpey, W.D. Potter. Odin and Hodur: Using Bluetooth Communication for Coordinated Robotic Search. Proceedings of the 2004 *International Conference on Artificial Intelligence (IC-AI '04)* Volume 1, pp.365-371. Las Vegas, Nevada, 2004.

[11] Feddema, J. and Schoenwald, D. Decentralized Control of Cooperative Robotic Vehicles. *Proceedings of SPIE Vol. 4364*, AeroSense, Orlando, 2001.

[12] Parunak, H. and VanderBok, R. Managing Emergent Behavior in Distributed Control Systems. *Presented at ISA-Tech*, Anaheim California, 1997.

CHAPTER 4

CONCLUSIONS

The papers presented serve to demonstrate the possibility to design systems that are capable of distributing a complex task to multiple robots. The task of distributed mapping involves many challenges that can be difficult to overcome. The primary challenge of this task is creating a system that results in the effective coordination of many individual robots to complete a complex task. The first paper "Odin and Hodur: Using BlueTooth Communication for Coordinated Robotic Search," served to establish Bluetooth as a valid means of communication. Additionally, the paper demonstrated that it was possible for simple robots to use conventional means to find a specified target, localize their position, and send that information to another robot. This work paved the way for developing more complex interactions that make it possible to accomplish more difficult tasks by demonstrating a primitive collaborative ability.

The second paper, located in Appendix A, focused on applying the lessons that were learned building Odin and Hodur and extending that knowledge to include a new robotics platform with enhanced capabilities. In this paper "Distributed Robotic Target Acquisition Using BlueTooth Communication," the main emphasis was on better sensor integration and increased ability to localize the position. For this task, distributed acquisition of a specific target, a new robot chassis was built that allowed for better sensory capabilities and increased mobility. This task was successfully completed, and the robotic chassis is now available for future projects.

The next paper, “Using Multiple Collaborative Robots for Terrain Mapping,” proposed a collaborative reward based system that allowed for task distribution over several robots. In this project, a method was proposed that used information credits as an incentive for individual robots to perform desired tasks. The information credits are reduced by survival pressure that functions like a tax on these credits for every robot. If a robot depletes its information credits then it is placed in an inactive state for that particular task, though it can always be reactivated for later tasks. In this specific example, mapping an unknown environment was chosen as the specific task. It was explained that within the task of mapping an unknown area with multiple robots, there are two required robotic behaviors. The first behavior that is required for the completion of this task is that of individual robotic exploration. It is necessary for the individual robots to explore their environment and record their observations. The second required behavior for a system attempting to distribute a task over multiple robots is that of communication between the individuals. The underlying premise is that by using a reward system, based on information credits that are granted for the individual robots demonstrating the desired behaviors, it is possible to distribute a complex robotic task across multiple robots without having a centralized control structure. It is expected that by providing this type of reward system based on information credits and by enforcing a survival pressure, the individual robots will develop reinforcement based emergent behaviors that tend to maximize their rewards. By maximizing their individual rewards, it is expected that the specific task will be completed by increasing the collaborative abilities of a group of robots.

These papers represent a progressive model for how robot development should continue. From the proof of concept models of Odin and Hodur to the creation of a unique robot chassis using model tanks, this has been an evolutionary process that has taken years to complete. While it is certainly not meant to be the final task to implement the structure proposed in the "Multiple Collaborative Robotics for Terrain Mapping," experimental verification is the next logical step.

APPENDIX A

DISTRIBUTED ROBOTIC TARGET ACQUISITION USING BLUETOOTH COMMUNICATION[5]

[5] "Distributed Robotic Target Acquisition Using BlueTooth Communication," by J.T. McClain, D.H. Barnhard, B.J. Wimpey, and W.D. Potter. In the Proceedings of the 42nd Annual ACM Southeast Conference, pp.291-296, Huntsville, Alabama, April, 2004.

A.0 INTRODUCTION

This paper presents a multi-robot system designed to use short range Bluetooth communication to solve a team-based search and destroy task. This project represents an extension to an earlier project described in [1] that used Bluetooth communication to solve a simple multi-robot search and communication task. The primary goals of this extended project include the development of a cheap and easily produced platform with Bluetooth capabilities, development of a more robust Bluetooth communication scheme, and the development of behavioral models that allow efficient team-based search and destroy tactics in a military-like scenario.

A.1 RELATED WORK

Previous work in the area of multi-robot communication has focused primarily on the 802.11x wireless standard. Gerkey et al. [3] developed Player, a network server that provided “transparent network access to all sensing and control” of multiple robots (p. 1226). Nguyen et al. [5] 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 multi-robot system to solve any task which requires coordinated movements through communication, the information being communicated must be accurate. Location data are the primary information to be communicated. If both localization systems are inaccurate, even slightly, the error will compound and the accuracy and efficiency of the system will suffer tremendously. Barnhard et al. [1] described a simplified version of landmark-based navigation similar to the one presented by Uther et al. [6], which outlined 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, the present system uses landmark cues to determine location as well. This method will be described in detail in the following section.

A.2 PRELIMINARY WORK

Because of the inherent complexity of the system, it was decided that development should be divided into two phases. The first phase consisted of two robots, Odin and Hodur1, which were designed to coordinate their movements to solve the “Honeybee” task. The primary goal of the first phase was to create a robust communication system for the transmission of location information to enable coordinated movement between two robots.

The “Honeybee” task is based on the behavior of the common honeybee, in that each worker bee, upon the discovery of a food resource, is able to use communication to direct other workers in the hive to the resource for collection [4]. In the task, there are two robots, a “guide” robot and a “blind” robot. The goal of the task is for the guide robot to locate a simple target (a small light bulb for example) within the environment and lead the blind robot to it solely through the communication of its location.

The robots used in the first phase consisted of two modified Brainstem Palm Pilot Robot Kits (PPRK) from Acroname Robotics. A Brainstem is a micro-controller with a 40 MHz RISC processor, 1 MBit Inter Integrated Circuit Bus (IIC) port, 5 digital input/outputs, and 4 high resolution servo outputs. It is useful for controlling basic robotic functions. The PPRK is a robot kit developed by Carnegie-Mellon University that is characterized by a holonomic drive system and a chassis that is easy to modify. These 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. The Bluetooth protocol allows for the creation of a short-range radio link between electronic devices and is primarily used for wireless desktop applications. It is characterized by a 30 foot range and a data transmission speed of 1 Mbps. Despite the possible use of the Brainstem as the primary controller, both phases only used it to relay messages between the iPAQ and the robot's sensors and actuators.

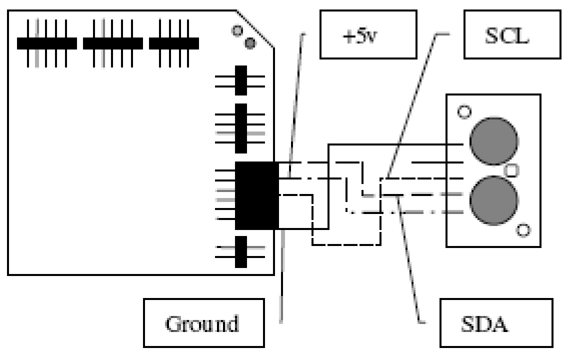

Figure 3.1 The Brainstem/SRF08 Interface

Each PPRK 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. The SRF08 utilizes sonar to detect objects in the environment up to approximately 20 feet away. The CMPS03 is an electronic compass that uses the Earth's magnetic field to detect orientation to an accuracy of 1˚. Figures 3.1 and 3.2 illustrate the interface between a Brainstem and each of the respective sensors. Both the SRF08 and the CMPS03 are capable of being connected directly to the IIC Bus on the Brainstem and referenced as addressable devices. In both Figures 1 and 2, the SCL and SDA connections represent the communication lines for the IIC Bus. Both robots were identical in that they both possessed the same modifications.

The behaviors developed in the first phase consisted of both search and localization behaviors. Since the SRF08 sensor also has the ability to take light intensity readings, we chose a light bulb as the target. Thus, the search strategy in both phases consists of a greedy algorithm designed to maximize the light intensity reading.

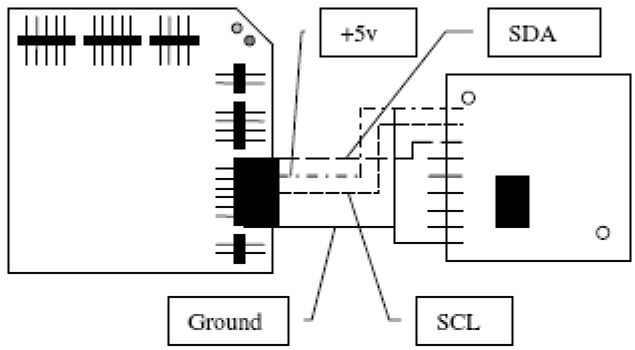

Figure 3.2 The Brainstem/CMPS03 Interface

In order to simplify the task, a simple landmark-based localization scheme was used (see Figure 3.3). 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 via a Bluetooth radio link.

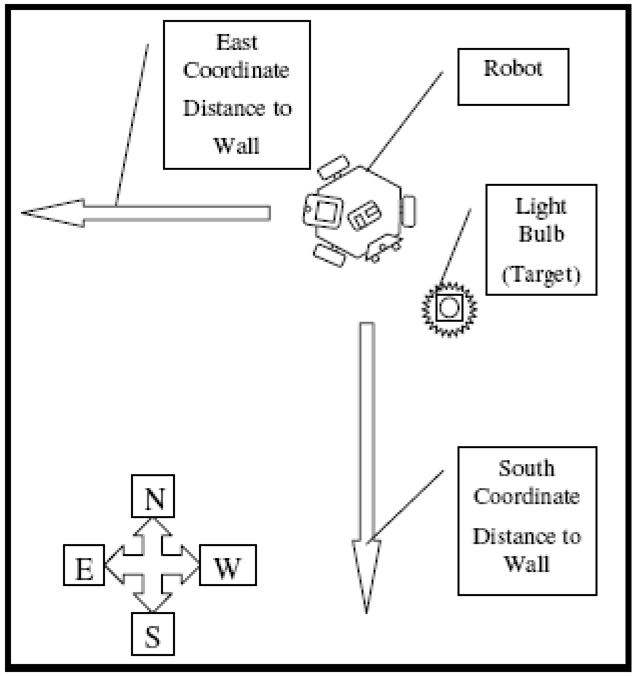

Figure 3.3 The Localization Process

A.3 CURRENT WORK

A.3.1 TASK DESCRIPTION

The task for the current phase uses the same environment as the first phase; a playing field surrounded by four walls placed at cardinal directions with a light bulb placed somewhere within the environment to serve as the target. The current task differs from the first, however, along a number of factors. First, all the robots participate as a team in the search, allowing for a quicker search time. This also reduces the possibility of a system failure through the failure of one robot. Second, the main goal in the first task of leading the blind robot to the target has become a sub-goal in the present task, with the main goal now being to find the target and destroy it in a coordinated fashion. Finally, because this is a military-like scenario, the robots must exercise greater caution throughout the execution of the task.

When the target is found, the robot that found it must communicate the target' s location to all the other robots involved while at the same time keeping a safe distance from the target (depending on the current attack strategy). While the other robots are en route to the target, a decision must be made regarding the attack formation that is to be used. The final requirement of the task is that the system must be able to adapt quickly to the loss of any team member, in order to carry on the mission.

A.3.2 ROBOT ARCHITECTURE

One of the primary goals of the current phase was to develop a more robust robot platform. The PPRK robots used in the first phase, while being quite well suited to the Honeybee task, had a number of design limitations that made them unfit for the current task. First, a PPRK cannot react quickly enough to succeed in a fast paced environment; and second, it is difficult to modify the chassis of the PPRK to accommodate additional sensors as well as a projectile launcher.

Figure 3.4 Projectile Launcher

It was decided that the new platforms would consist of model tanks (see Figure 3.4), modified to be autonomous robotic vehicles with Bluetooth radio communication capabilities. Each tank runs on a standard 9.6 volt battery, has two individually driven treads, and also has a rotating turret with a built in projectile launcher. The firing mechanism is capable of individually firing 12 foam missiles, allowing for multiple attack sequences without reloading. In addition, it is operated by a small DC motor, allowing for simple programmatic operation. This chassis is also able to move very quickly, allowing for the quick reaction times required in the task. The tank is able to climb up to a 30˚incline, as well as cross small ditches and other obstacles.

The control architecture for the new chassis is similar to the architecture used in the PPRK in that it still consists of the iPAQ/Brainstem interface. However, to allow for more sensory apparatus and control, the current architecture consists of two Brainstems (per tank) networked together across the IIC Bus. Figure 3.5 presents the wiring diagram for networking two Brainstems across the IIC Bus. Adding a second Brainstem presents a distinct advantage, because it allows each Brainstem to work on a task in parallel. The current system uses two Brainstems because we needed more input/output ports than were available on one Brainstem. For greater ease in design, we allocated one of the Brainstems completely for motor control, and the other for sensor control. The iPAQ remains the master controller of the system; all instructions from the iPAQ are routed through a single Brainstem, which then proceeds to route instructions by address across the IIC Bus to the correct receiver. In this way, it is simple to add more sensors or motors to the system by merely adding more Brainstems to the on-board controller network.

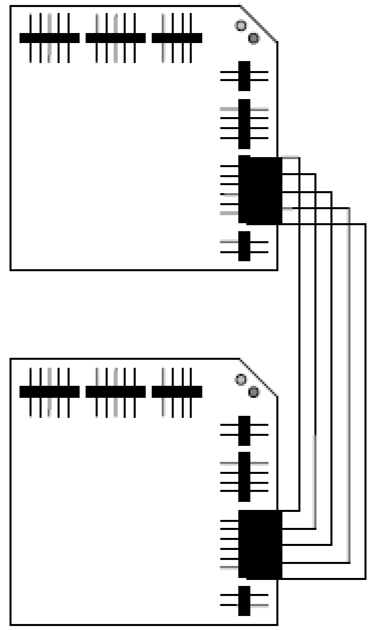

Figure 3.5 Networking two Brainstems Via the IIC Bus

The new chassis still has a CMPS03 compass sensor to obtain orientation information, but now also possesses two SRF08 ultrasonic sensors mounted on each side of the front of the vehicle for better targeting and tracking. The motors of the tank are controlled by two Texas Instruments SN754410 Quadruple H-Bridge Drivers which are interfaced with the Brainstem's digital outputs. An H-Bridge is a circuit which allows for the reversal of current through a DC motor, controlled by logic signals, to allow for the motor to be run in two different directions. To simplify operation, it was decided that the turret of the tank should be locked in place so that all targeting and tracking would be done by rotating the entire body of the robot. We plan to remove this constraint in the next version. Each robot was capable of communicating with others via a Bluetooth radio connection, which is a feature of the Compaq iPAQs.

Figure 3.6 The Modified Tank Chassis

Overall, the final product (see Figure 3.6) is an extremely robust chassis that is fairly easy to construct. In addition, the capabilities of each robot easily exceed the qualifications required for the task at hand.

A.3.3 BLUETOOTH COMMUNICATION INTERFACE

In a team-based task such as this, one of the most important aspects is the communication system, which facilitates coordinated team activity. This is because failure of the communication system can lead to disastrous results. In fact, a failure in communication can be more damaging than the loss of any single member of the team because it has an effect on the actions of all the robots involved in the task. In contrast to this, if the communication system is robust and adaptable, then the loss of one of the team members will only have a minor effect on the outcome of the task. In a situation such as this, the remaining robots could adapt their behavior based on the realization that one robot has been lost, and continue on to complete the task with only a minor reduction of effectiveness. This is particularly advantageous in our military-like scenario, where it is almost guaranteed that several of the robots will be lost before the task is complete. As autonomous vehicles, they have the ability to decide whether to continue or withdraw, depending on the attack strategy.

The communication system that was developed in the first phase is extremely susceptible to the loss of either of the robots involved in the Honeybee task. For example, if the guide robot is lost while searching for the light bulb, then the whole process is unsuccessful because the blind robot will continue to wait for the coordinates without realizing that the task is no longer achievable. In the same way, if the blind robot fails, the seeker will send the first coordinate and wait forever for an acknowledgement that the coordinate had been received. Thus a major goal of the second phase was to expand the current communication system so that it is no longer dependent on any single robot, in order to minimize the impact of team member losses on the team' s overall performance.

To come up with a model for maintaining the status of all the robots involved in the task, we turned to the idea of Routing Information Protocol in the field of computer networking. The Routing Information Protocol (RIP), helps the members of a network maintain a list of all the active routers in the system. In RIP, each member sends its entire table of active routers to its nearest neighbor every 30 seconds. Upon receiving the table, the member updates its own table to reflect the new information [2].

The communication model for the current phase works similarly to RIP. In the system, each robot sends out a "heartbeat" signal at regular intervals to all the other robots within its range. Each robot maintains a Status table that contains a Boolean status variable for each robot, as well as the time allotted since the last heartbeat received from each robot. If the robot does not receive a heartbeat signal from a particular robot for an extended period of time, then the robot switches the status variable to "0" or inactive ("1" means active), signifying that the other robot is either not functioning or has passed out of range.

Table for Robot 1			Table for Robot 2		
Robot #	Status	Time	Time	Status	Robot #
Robot 1	1	NA	1	1	Robot 1
Robot 2	1	1	NA	1	Robot 2
Robot 3	1	2	12	0	Robot 3
Robot 4	0	13	5	1	Robot 4

New Table		
Robot #	Status	Time
Robot 1	1	1
Robot 2	1	1
Robot 3	1	2
Robot 4	1	5

Figure 3.7 Combining Status Tables

In addition to the heartbeat signal, each robot also sends a copy of its status table to all the other robots that it still considers active. Upon receiving a status table, each robot updates its own table to reflect the new information. This is done by always reverting to the most recent heartbeat time. Thus, in Figure 3.7, robots 1 and 2 are combining their status tables. In this example, an elapsed time of ten minutes results in a classification of inactive, as shown by Robot 4’s time of 13, which means no heartbeat has been received for 13 minutes. Thus Robot 1 has labeled Robot 4 as inactive. But Robot 2 lists Robot 4 as still active, since the last heartbeat it received was only 5 minutes ago. So, the new table for both robots (1 and 2) reflects that Robot 4 is still active and the new time is reverted to 5 minutes. A similar process is done for Robot 3 also.

One of the limitations of the iPAQ 3970s that we used was that the Bluetooth version does not allow for the creation of a robust local area network (e.g. many clients conversing simultaneously). Instead, facilities are provided to establish a direct virtual serial connection between two devices. Had we used a newer model iPAQ running the Windows CE .NET framework, this problem would not have been present. As a result of this restriction, we were forced to implement a staggered message passing scheme. Thus, when one robot wants to communicate with another, a serial connection is opened; the communication proceeds; and then the connection is closed again to allow for other communication to take place. If a connection is initiated with a robot that is already engaged with another, the initiating robot receives a “busy signal” and knows to attempt reconnecting later.

It is important to note that our prototype communication system is not intended to replace current military communication protocols. Bluetooth is not an adequate solution for such applications mainly because of its limited range. Instead the focus of this project was the development of communication strategies for coordinated robotics rather than a communication system itself. Thus, the strategies we are developing could easily be implemented with a more applicable communication protocol.

A.3.4 BEHAVIOR

The behaviors in the current phase are similar to those in the previous phase in that they are divided into two parts. The first part consists of locating the target through a distributed search. This is characterized by each of the robots searching for the target independently, using a greedy method similar to the one used in the Honeybee task. The search differs from the previous phase, however, in that when the target is found, the robot does not approach it. Instead, it stays at a safe distance and waits for the rest of the team to arrive (again, this depends on the attack strategy in use). With a large number of robots engaged in the search, it can proceed very quickly.

Once the target has been discovered, the robot that has discovered it (primary attacker) sends a quick notification to all the other robots that are still active, whereupon they begin to prepare for the localization procedure by turning to a cardinal coordinate. Note that an inactive robot (with respect to the primary attacker) will also be notified indirectly.

After determining its location, the primary attacker begins the process of deciding an attack formation. The formation chosen for this task is determined based on the attack strategy and current situation. The example formation in Figure 3.8 allows each of the robots to localize easily, as well as maintain a clear line of site to the target to prevent “friendly fire” problems.

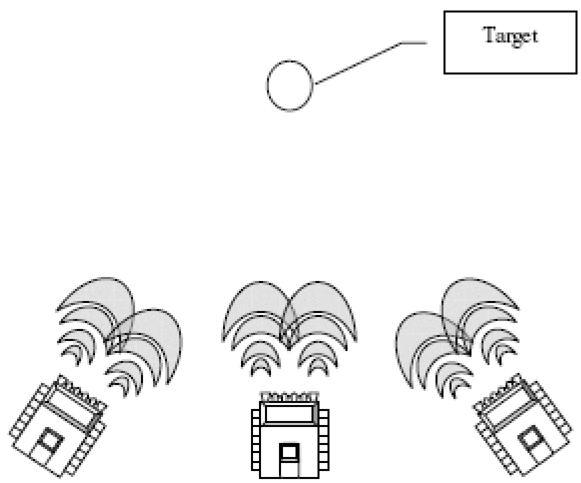

Figure 3.8 An Attack Formation

At this point, a quick determination is made to figure out how many other robots there are in the task force, this number is then passed via Bluetooth to all of the other robots in their status tables. Depending on the number of surviving robots, different attacks can be agreed upon and executed. For example, once all of the robots have maneuvered themselves into the correct position, each robot, in order will send the message that it is in position and ready to fire. Once every robot has locked in on the target, the fire command will initiate, and the target will be destroyed.

A.4 CONCLUSIONS AND FUTURE DIRECTIONS

We have presented a two-phase project with the goal of using Bluetooth communication to control distributed team-based multi-robot systems. The first phase used the Honeybee task to develop and test a robust Bluetooth communication scheme. The second phase applied what was learned to a more difficult military-like search and destroy task. Preliminary performance of the cooperating tanks indicates that Bluetooth communication is a viable option for applied multi-robotic systems.

There are several areas of the current system that warrant improvement. First, the present system is unable to account for moving targets. Future work will focus on developing the system's ability to dynamically track moving objects. In addition, work will continue on developing improved strategic planning in the overall system. Furthermore, Bluetooth communication should be improved to allow for robots that have gone out of range to reconnect seamlessly and easily to the rest of the group. Finally, a major limitation of the current system is that all activity must occur in a constrained environment, in order for the localization procedure to work. Future work will also focus on developing the system's ability to work in more complex environments.

A.5 ACKNOWLEDGMENTS

The authors wish to thank the reviewers for their insightful comments. In addition we would like to thank Dr. Michael Covington for his support in fielding questions about electronic devices of all types. Finally we would like to thank Trisha McClain, for proofreading earlier versions of this paper, and for her endless love and support of the first author through the many months that this project has entailed.

A.6 REFERENCES

[1] D.H. Barnhard, J.T. McClain, B.J. Wimpey, W.D. Potter. Odin and Hodur: Using Bluetooth Communication for Coordinated Robotic Search. Proceedings of the 2004 International Conference on Artificial Intelligence (IC-AI ‘04). Las Vegas, Nevada, Submitted.

[2] B. A. Forouzan. TCP/IP Protocol Suite. McGraw Hill, Boston, 2003.

[3] B.P. Gerkey, R.T. Vaughan, K. Stoy, A. Howard, G.S. Sukhatme, and M.J. Mataric. 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.

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

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

[6] W. Uther, S. Lenser, J. Bruce, M. Hock, and M. Veloso. CM-Pack’ 01: Fast Legged Robot Walking, Robust Localization, and Team Behaviors. RoboCup-2001: The Fifth RoboCup Competitions and Conferences, 693-696, 2001