Simon Fraser University
Burnaby, BC
V5A 1S6
trac-tech@sfu.ca

April 30, 1999

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, BC
V5A 1S6

Re: ENSC 370 Project Gerbil Process Report

Dr. Andrew Rawicz,

The attached document, *Project Gerbil Process Report*, provides a summary and review of the Project Gerbil. The Gerbil's core purpose is to autonomously navigate and map a room, utilizing a positioning and sensory system. This project would allow for various applications. One such application would be the cleaning of hospital floors, which is the purpose we will explore.

Contained in this document is the current status of the project, our future plans for TRAC Technology, and a look back on what we have learned from the Gerbil.

TRAC Technologies comprises of four intelligent, hard-working, and innovative third-year-engineering students -- Celina Tinio, Robbie Grue, Andrew McPherson, and Travis Hammond. If you have any questions or concerns regarding our functional specification, please contact Celina Tinio at 872-3750, or by e-mail at mtinio@sfu.ca.

Sincerely,

Celina Tinio
TRAC Technology

Enclosure: *Project Gerbil Process Report*

Project Gerbil Process Report

Submitted by: **TRAC Technology**
Celina Tinio, Travis Hammond, Robbie Grue, Andrew Mcpherson
trac-tech@sfu.ca

Contact: **Celina Tinio**
School of Engineering Science
Simon Fraser University
mtinio@sfu.ca

Submitted to: **Andrew Rawicz**
School of Engineering Science
Simon Fraser University

Steve Whitmore
School of Engineering Science
Simon Fraser University

Date: April 30, 1999

Table of Contents

Introduction

This process report summarizes the work completed and still pending for the Gerbil Autonomous Navigation Robot. Together we have toiled continuously, night and day, to achieve the goals outlined in previous documents. Although our implemented prototype has not been fully integrated, we realize that the project began with highly ambitious intentions. We feel satisfied with the challenges we faced and overcame and look forward to completing our work in the future.

Project Design Requirements

In addition to a common interest in robotic systems, the idea behind the Gerbil Project originated from the numerous uses of a robot that is capable of self-sufficient navigation and cartography. Examples of such applications include automated floor washing and lawn mowing, various cinematography applications, and safely detecting land mines. Figure 1 below shows the data flow outline for the Gerbil project.

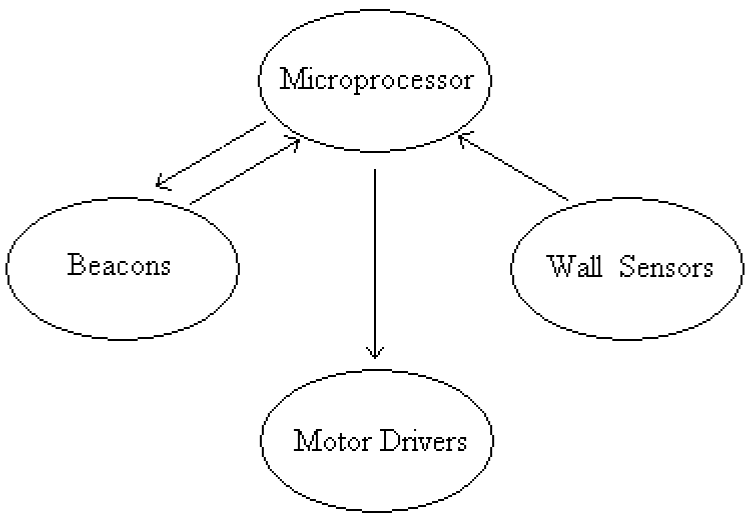

Figure 1: Data Flow Organization

As in all complex projects, many design issues had to be solved prior to undertaking any development. Some key issues include:

- How will inner objects/walls be sensed and what will be done to resolve object obstructions?
- What method of mapping will be used? (i.e. using an external positioning system (such as a beacon) vs. using a wall/corner reference)

- Do we want to include a user interface for any map viewing or other mapping functions?
- How will the object be capable of following a wall?

Our design decisions were then based on the following criteria:

Positioning System

The robot should be capable of traveling to any desired position (provided this position does not lie inside any object or beyond an outer wall). This allows the robot to predict and project itself towards different points without using walls or objects as reference. This is especially useful in open areas, such as lawns, etc. Using a beacon system provides the necessary positioning, and is also useful in obtaining accurate position measurements. Therefore, accurate mapping will also result from this approach.

The beacons are considered to be two-way data devices to the microprocessor in that they receive an RF trigger, and transmit an ultrasonic pulse. In this manner, the delay associated with the returning ultrasonic sensor can be timed and used as a position.

Wall Sensory System

Wall sensory should ideally cover the full 360° around the robot. This ensures that an object can be hit and sensed from any angle. We would also like to minimize the number of sensors required to achieve maximum coverage. A rotating wheel was then designed and implemented for this purpose. This wheel approach only requires 4 sensors to 8 angles of detection (I.e. front, front-right, right, etc). The wheel was also used to aid in following a wall. Theoretically, by pushing slightly into a wall, the wheel would provide a bumper to keep the core of the robot off the wall, while rotating to allow the robot to continue forward alongside the wall.

However, once implemented, we found that the robot would pull itself into the wall, rather than going forward. This slightly limited our ability to map objects. Figure 2 depicts this occurrence.

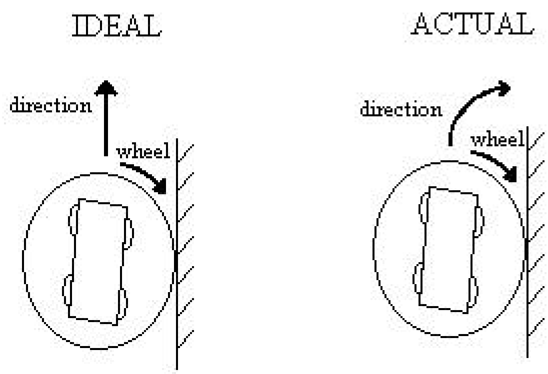

Figure 2: Wall Following Problem

Deviations from Original Proposal

As discussed in our preliminary demonstration, we realized that our design is not efficient for pure mapping functionality. Because we are using the wheel sensor, as opposed to some type of "envelope" around the robot, we are forced to touch each spot along a wall to determine the shape of an object. As well, the beacons must be pre-placed, which severely limits the robot's use in high risk areas.

Therefore, we decided to narrow the robot's core application scope to one that effectively uses the beacon and wall wheel systems. These applications include those in open areas (such as lawn mowing), but are also useful in cleaning floors since the robot would not be typically following a wall.

Therefore, we are concentrating more on being able to download pathing information to the robot, as opposed to uploading the mapped information from the robot. This allows users to program the robot with a custom pathing algorithm, which will be then executed by the robot. This allows users who know how they would like the robot to proceed. An example is that of custom programming the robot to cut a lawn. Because no two lawns are identical, the pathing algorithm can be customized for each lawn. This will be achieved through a GUI drawing program, along with another program, which downloads this information into a usable form on the robot. Figure 3 below shows this process.

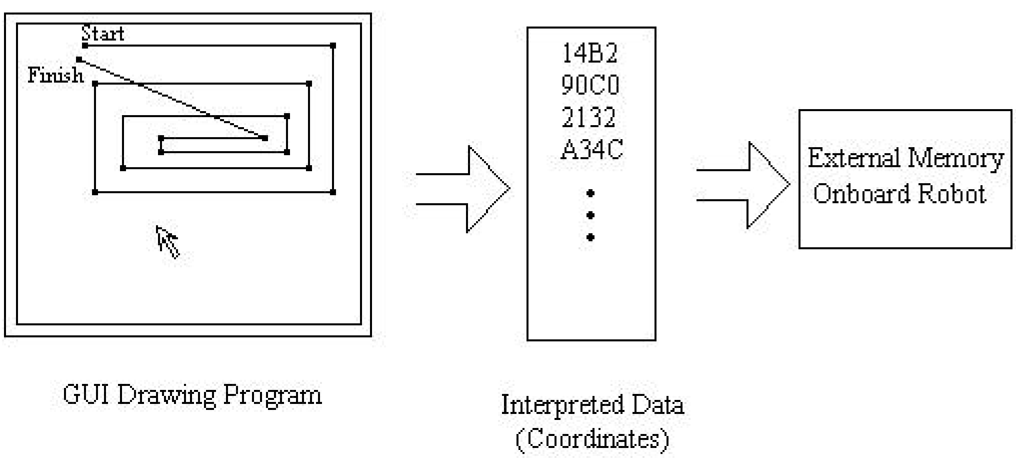

Figure 1: Map Drawing and Data Transfer

Future Plans

TRAC Technology is very eager to fully integrate the Gerbil system together, and apply this into an assembly line of industrial cleaners. Our target market is that of hospitals. Our vision to have a central robotic cleaner room, where recharging and restocking of cleaning materials takes place. From this central room, narrow inner corridors can give the cleaners access to rooms without interrupting normal hospital flow.

Using this system, hospital areas can be maintained continuously and efficiently. As well, any spills can be easily alerted and dealt with. Figure 4 below shows a top view of such a system.

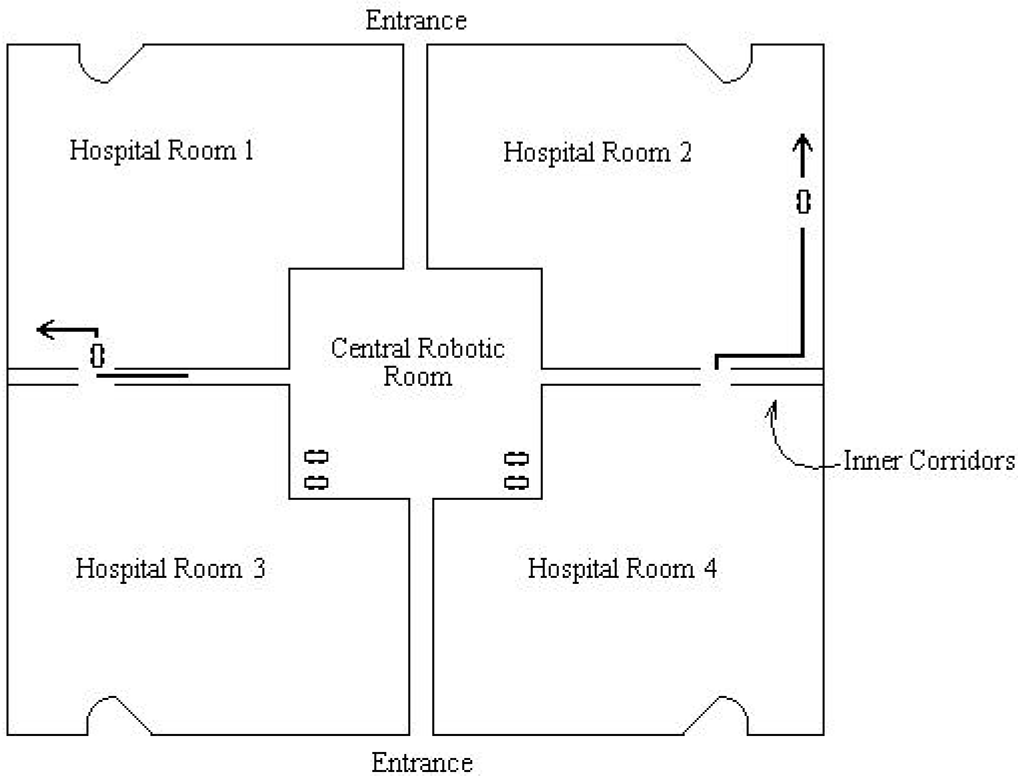

Figure 2: Hospital Cleaner System

Overall System

- *Physical Packaging* — The system is currently not enclosed enough to be used in routine hospital maintenance. The system will have to be packaged such that it can be easily washed off after cleaning. As well, the beacons will have to be integrated into a wall outlet.

- *Beacon Power Supply* — The beacons have not been adapted to run from an AC wall outlet, and thus this still requires integration. This only requires the use of an adapter and can be done at a later time.

- *Hospital RF Interference* — Hospitals do not allow RF signals (such as from cellular phones) to be transmitted while inside. Our original beacons used a frequency of 300MHz. Our second RF modules used a 900MHz carrier frequency. In order for our equipment to be used in a hospital we would have to be sure that the particular band in which we are transmitting is allowed in Hospitals.

- *RF System Delay* — The First RF system used introduced a considerable delay into the distance finding system. When the RF transmitter was given a square wave input, the output of the receiver jumped from low to high, but did not begin to output the desired square wave for another 100ms. This delay was too large considering the actual time to be measured was less than 10ms. To fix the problem we tried to add a preamble of a non-critical frequency so that the RF system would constantly be powered on. However, there were still random fluctuations in the measurement of time. We decided to use a more robust RF system with a higher transmission rate. This system worked fairly consistently although because it is in a band used by cellular telephones, interference can undermine system performance.

- *Wall Wheel Adjustments* — In order to allow a more reliable wall following algorithm, the wall wheel must be modified such that our original intent (see Figure 2) is maintained.

GUI Map Drawing Program

- *Provide Detailed Distances* — Currently, there are no distance references in the program, making it very difficult to accurately predict how far the robot will physically go.

- *Allow Modifications to Current Paths/Maps* — The program currently only allows pathings to be created. Current pathing maps must be viewable and modifiable.

Budgetary and Time Constraints

The estimated and actual cost of the Gerbil project is shown in Table 1 below.

Table 1 : Approximate and Actual Project Budget

Component	Estimated Cost	Actual Cost
Microprocessors/Logic/Memory	$150	$20
Motors/Actuators	$150	$250
Sensory Devices	$150	$10
Transmitters/Receivers	$100	$60
Communications Interfaces	$100	$30
Power Supply	$75	$0
Discrete Components	$50	$50
Printed Circuit Boards	$100	N/A
Casing/Wheels	$150	$20
Wiring/Cables	$75	$50
Total	**$1100**	**$490**

The motors were more expensive than originally estimated because of the need for more power to drive our robot. However, we managed to spend less than our estimated costs by borrowing or obtaining second-hand parts. As well, we decided to put our circuits onto vector boards as opposed to having PCBs made in order to reduce costs.

In terms of time management, we made a big effort to follow our proposed schedule very closely. However, due to several unexpected bugs and hardware difficulties we tended to fall behind schedule. We were fortunate to be given a few extra days to solve difficulties with our RF modules, for example.

The TRAC Team Finish

ENSC 370 has been a course which has taught us a great deal. One of the most important aspects, perhaps *the* most important aspect, has been the experience of working with a team of peers on a project which we can truly call our own.

As a team, we learned about the various design stages of a robot. While each member concentrated on a certain area of the project (microprocessor, positioning, wall following, or motor control), we also participated in the design and development of the other areas of the project. For each area, the self-initiative needed to research the various aspects and solve design problems has shed invaluable light upon the true meaning of “engineering”. We have discovered the need to be able to rely on each as a team, and to take faith in each other’s abilities in order to reach a common goal, which is a lesson that goes beyond engineering.

This course has also provided a real sense of accomplishment, since from the very beginning, this project was something that we had chosen as a group, and it progressed according to our work and effort. This experience has given us a responsible attitude towards and a sense of pride in our work, which will help us to achieve further great heights success.

TRAC Technology would like to thank Nakul Verma, for allowing us to tear his old project to shreds, and Tim Norman, for providing a very much needed last minute solution. We would also like to thank the TA’s, Jason Rothe, Victor Tang, and Greg Hall for their time and help, Steve Whitmore for spiritual guidance, and last but certainly not least, Andrew Rawicz, for giving us the opportunity to take part in this course, and providing continuous encouragement and help throughout the semester.