CONTROL ALGORITHMS FOR *SPACE TUG* RENDEZVOUS

Project Design Proposal

16.621

Fall 2002

Author: Timothée de Mierry

Advisor: Olivier de Weck **Partner:** Gergana Bounova

Tuesday, December 10th

EXECUTIVE SUMMARY

The Space Tug project finds its origin in an ongoing joint research project between MIT and the Defense Advanced Research Projects Agency (DARPA), whose purpose is to develop an orbital servicer satellite, the Tug. The purpose of the satellite would be to carry out servicing missions for other satellites in orbit around Earth. The DARPA project's objectives are: the development of a capability to service satellites; an economic solution to the high-energy problem in space; a universal grappling capability; and a strategy to find the target satellite efficiently. This 16.62x project is a subset of the general problem and relates to the efficient search and rendezvous strategy.

The proposed experiment will test a set of search strategies that represent all possibilities of rendezvous algorithms that could be used on the Space Tug. The problem will be modeled as a two-dimensional situation, in which a robot simulator will be programmed to find its target. The time elapsed and the energy consumption will be measured to provide the necessary data to develop a cost function representing the trade-offs of each search strategy. The most efficient strategy will be chosen to minimize the cost function. Conclusions drawn from the analysis of the data will be presented to the broader DARPA project.

The estimated budget for the proposed experiment is approximately $500 and the results will be presented in a final report by May 13th, 2003. Oral presentations and progress reports will be produced during the fifteen-week period.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1. INTRODUCTION

1.1 Background and Motivation

The *Space Tug* project is an ongoing MIT/DARPA research project that aims at developing a satellite – the Tug – to carry out rendezvous and docking with a target satellite. The mission of the Tug will be to capture its target, change its position and orbital elements by a predefined amount, and release it without damage to the target satellite or itself. The capabilities of the Tug must include the following: rescue satellites from unusable orbits, orbital debris removal, tactical operations, and other emergent uses. The control system of the Tug used to find and dock with the target is a major aspect of this project.

The search for the target satellite is a complex procedure. Although approximate coordinates for its location would be provided, the Tug would still have to search a finite space to find it, since current tracking does not give precision below a magnitude of the order of one hundred meters. As a result, the Tug has to have intelligent identification and sensing strategies implemented in its control system to approach the target.

A major technology risk in the *Space Tug* project is the target identification and docking. Showing that such a process is feasible would reduce this risk and would provide a possible solution to the problem. In addition, given that the control system of the Tug has thus far been modeled as a black box, the results of this research project could give clues as to what the architecture of the control system of the vehicle should be. Furthermore, successful search and rendezvous strategies could be used in other aeronautical applications, such as autonomous and formation flight.

Search strategies and algorithms for robots have been studied extensively in the past years. As a result, the design part of this project will be influenced by previous work done on search algorithms, notably by the MIT Department of Electrical Engineering and Computer Science. The basic strategies will be enhanced to fit the purpose of the Space Tug.

Time and energy consumption, as well as the successful implementations of three searching strategies, will be the focus of the research project. It will provide a first look at the most effective rendezvous strategy that could potentially be used in the *Space Tug* project. Spanning the space of possible search algorithms, three strategies will be tested: (1) random sensor-less search, (2) semi-autonomous with a human in the loop search and (3) fully autonomous search with sensors. Ultimately, the goal of this project is to show that the semi-autonomous with a human in the loop search will satisfy this important requirement.

1.2 Hypothesis and Success Criterion

The use of a semi-autonomous search system with a human in the loop is the algorithm that will be the most effective for rendezvous and docking strategies in terms of time and energy consumption. Success for this project is a clear definition of whether or not the semi-autonomous search system is the most effective algorithm for rendezvous and docking strategies in terms of time and energy consumption. The three different search strategies span the full space of search algorithms in an effort to provide a valid assessment of the hypothesis.

1.3 Objectives

The project is divided into two distinct parts that will achieve two different but closely related goals. The preliminary objective is to develop, implement, and test three different strategies for two-dimensional, non-cooperative target search and precise docking. This goal is the first step in achieving the primary objective.

The primary objective of the experiment is to develop a cost function in order to compare these three strategies based on the trade-off costs between time and energy. The algorithms will be evaluated using predefined criteria, developed into the cost function, to examine the performance of each.

2. LITERATURE REVIEW

The article entitled "On-board software for the Mars Pathfinder Microrover", written by Morrison and Nguyen[1] describes the software used to control the motion of the rover on Mars. The constraints, in terms of communication and energy, on the control system of the Mars rover are similar to what the Tug will face in space. It is thus important that these constraints be taken into account when modeling the searching procedure. Due to electrical and processing power limitations, the control system of the rover is unable to communicate and move at the same time. In addition, due to communication restrictions, mainly the time it takes to transmit information from Mars to Earth and back, the control system of the rover uses waypoint navigation and autonomous collision avoidance algorithms. In the absence of any obstacles, the rover proceeds directly forward to the waypoint, including stops for proximity scanning – for hazard detection. During proximity scanning processes, the rover uses its on-board optical sensors to generate an approximate map of the terrain map in front of the vehicle. Based on height differences in the map, the navigation system analyzes the possible locations of obstacles. Finally, an alternate working mode of the control system is the "rock finding" option, which uses the terrain map to detect a rock. The navigation system corrects the rover heading, centering it between the rock edges.

The Mars Pathfinder rover uses a collision-avoidance control system. This technique is the opposite of what has to be developed for the Space Tug. While the rover uses the terrain map to trace a route around objects, the Tug will have to trace a route to its target. Although this paper is not very useful in describing the actual search algorithms, it provides a background for the type of software architecture that is usually used in space vehicles. The same power and communication constraints apply to the Tug, and therefore its on-board software needs to make use of the same techniques for telemetry, which are necessary for the semi-autonomous search. Furthermore, the proximity scanning process implemented in the Mars Pathfinder is similar to what needs to be developed for the fully autonomous version of the Tug. Although this paper is general and does not provide more detailed information about the core of the software, it describes a basis for the architecture for the control system that will be used on the Tug.

The second reference is a paper by Gelenbe[2] entitled "Autonomous search for information in an unknown environment", which describes different search strategies from a "computer science" point of view. The author models the autonomous process in which an agent, a robot or a software algorithm searching for information in a computer database, searches the space around its current location for information it wants. The search area is divided in a set of locations (x, y), defined in a Cartesian space. Associated with each location is a probability q(x, y) representing the likelihood of finding the information wanted at this location. Assuming the environment is static, the space can thus be described as a probability space. The agent, which in the context of this project is the Tug, always moves in the direction where the probability q(x, y) is the greatest. Once the agent moves to the new location – from (x_0, y_0) to (x_{new}, y_{new}) –, the probability $q(x_{new}, y_{new})$ of finding information at the new point is updated depending on what was found. The algorithm thus continuously updates the probability space, until the agent finds the right information – the target for the Tug. The paper further develops a more advanced model which is more applicable to information search in a computer system than to robotic search.

The above search algorithm, referred to as the "Greedy Algorithm" by Gelenbe[2], is relevant to the random search strategy that needs to be implemented in the Tug simulator. The algorithm that will be used in the project will most likely incorporate some or all components and rules of the Greedy Algorithm. While the results of the experiments are not significant for the project, the modeling process used by Gelenbe in his experiment will be useful in developing the model of the search space for the Tug. The mathematical tools used in the Greedy algorithm will be the same as the Tug's random search strategy, since the underlying probabilistic decision-making processes are similar (e.g. the agent goes to the location with the greatest probability in the space). Furthermore, the same principles can be used in the fully autonomous search. The main difference will be that the agent, which has sensors with a given range R, can now check for information in a space of radius R around the location (x, y). In this process, a greater number of probabilities can be updated to recalculate the space. In addition, the agent is also able to build a map of the environment revealing the exact location of the information with greater precision. Gelenbe[2] falls short of developing a smarter algorithm and explaining how the sensors would affect the efficiency of the search.

The third most relevant article to the project is by Hillenbrand and Hirzinger[3], and is entitled "Probabilistic search for object segmentation and recognition". Object recognition is viewed as a two part process. Firstly, a sequence of hypothesis about the object – its location, geometric shape, possible movement – is generated, using exterior sensors. The second part of the process evaluates these hypotheses based on the object model. This paper describes a new technique for object recognition in a specific scene in a probabilistic framework. It also introduces a new statistical criterion – the truncated object probability – to produce optimal hypotheses about the object to be evaluated for its match to the data collected by sensors. The author further develops a mathematical model to fit the search sequence in the experiment.

The depth in which this article goes is most likely beyond the scope of the Tug project. However, some of the concepts developed are useful for the autonomous search strategy to be implemented in the Tug. Based on the data from its sensors, the Tug will have to be able to recognize the target in a largely unknown scene. The object recognition technique developed by Hillenbrand and Hirzinger[3] is too advanced to use in an environment with one target. However, if implemented in the control system of the autonomous Tug, it will be provide an expandable algorithm that can, for example, be slightly altered to recognize multiple targets in motion.

3. TECHNICAL APPROACH

3.1 Overview of Experiment

The experiment's main objective will be to simulate the Space Tug's rendezvous with its target in a simplified two-dimensional environment. The space in which the real Tug has to operate is complicated and contains several degrees of freedom that cannot be reproduced in two dimensions. As a result, modeling assumptions will have to be made. The simulation will make use of the relative positions of the Tug and the target. The satellites are in the same orbital plane relative to Earth and their orbits have the same eccentricity. Therefore, the target is fixed at a point in space, relative to the search space's reference frame.

The experiment will make use of floor space for the search area, whose dimensions will represent the appropriate ratio of search area to Tug/target sizes. This ratio will be calculated using the real sizes of these vehicles and the space around the target created by position uncertainties. The experimental set up is shown in Figure 1.

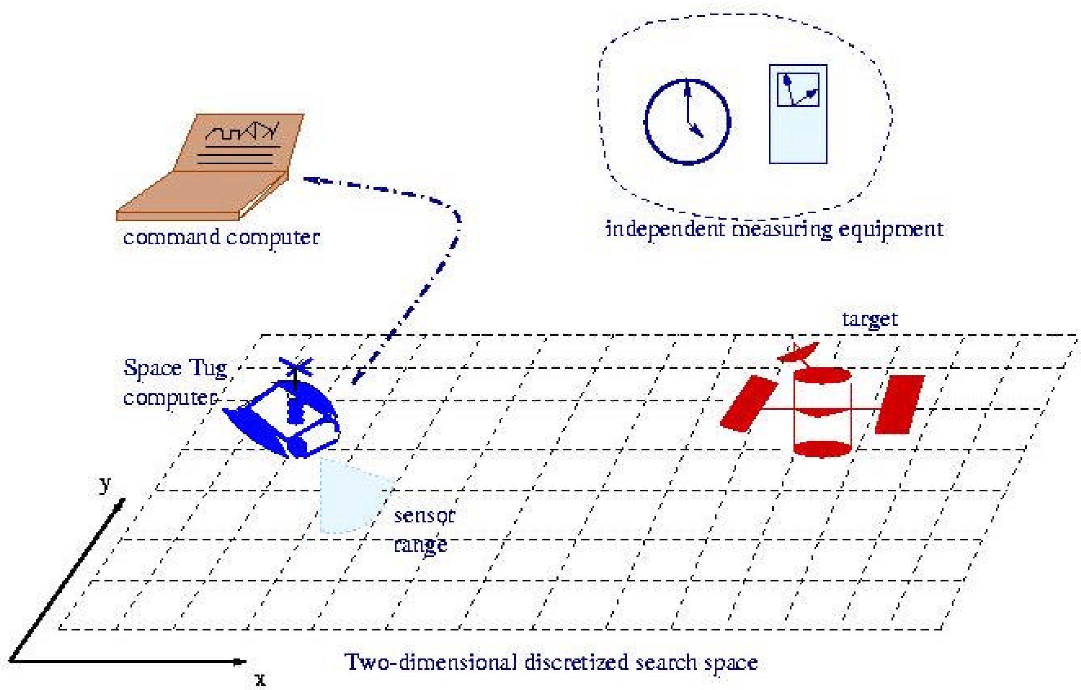


Figure 1 – Test-bed environment for Tug/Target rendezvous simulation

As shown above, the Space Tug computer will have to search through the space for the target, using its sensors. It is understood that the sensors' range will be much smaller than the size of the search space. Furthermore, in the case of the human-in-the-loop search strategy, a computer will be used to transmit commands to the robot using an infrared device, in accordance with the sensor data that the Tug computer will send to the human. For the other two search strategies, the computer will be used only to download the control system that will move the robot and the decision making software that will tell it where to go. Both the Tug and target will be made with LEGO? Mindstorms, using an on-board computer. While the target will be non-cooperative and inert, the Tug will carry, as mentioned previously, a collection of on-board sensors, including an ultrasonic range sensor, an infrared proximity sensor and a touch sensor. The first two will collect data about the position of the target, while the last one will stop the Tug upon running into the target.

The independent measuring equipment shown in Figure 1 will be used to record the time it takes for the Tug to find its target and the energy consumed during the process. Using this data, the cost function between time and energy will be developed, and the effectiveness of each strategy will be compared in order to assess the hypothesis of the experiment.

3.2 Overview of Hardware

3.2.1 Space Tug Robot

The robot simulating the orbital servicer will be made of Lego Mindstorms parts. The on-board computer is a RCX 2.0, shown in Figure 2. The computer has a total of six input ports. Three of them are used for sensors, while the others are motor inputs used to control the Tug's

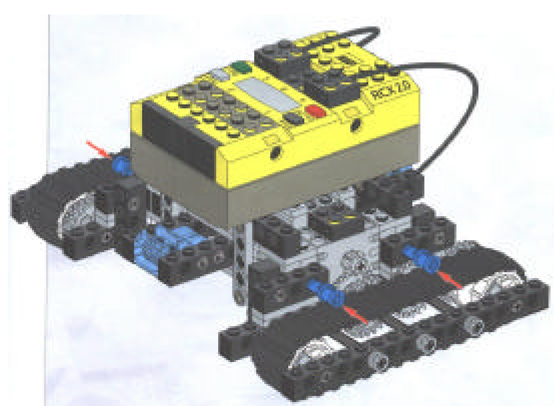


Figure 1 - Base design for Space Tug robot

movements. The computer also has an integrated infrared transmitter and receiver, which provides the necessary interface between the command computer and the Tug robot. The infrared signals are exchanged with an infrared tower, which communicates to the command computer through a universal serial bus (USB) interface. As a source of energy, the Space Tug robot makes use of six 1.5 volts rechargeable batteries (standard AA).

The RCX computer is programmed in a special language called Not Quite C (NQC). As its name suggests, it is similar to the C programming language but contains custom functions to define and use the input ports on the computer. The NQC language was developed by Baum[5] using Lego's MindScript™ and LASM™ codes via the RCX software development kit. Programming tools and compilers for NQC are readily available and fully compatible with the Space Tug simulator.

3.2.2 On-board Sensors

The orbital servicer simulator will carry an array of sensors in order to carry out its search of the target. Added to the standard touch sensors, the Tug will possess an ultrasonic range sensor and an infrared proximity sensor. These are shown below in Figure 3.

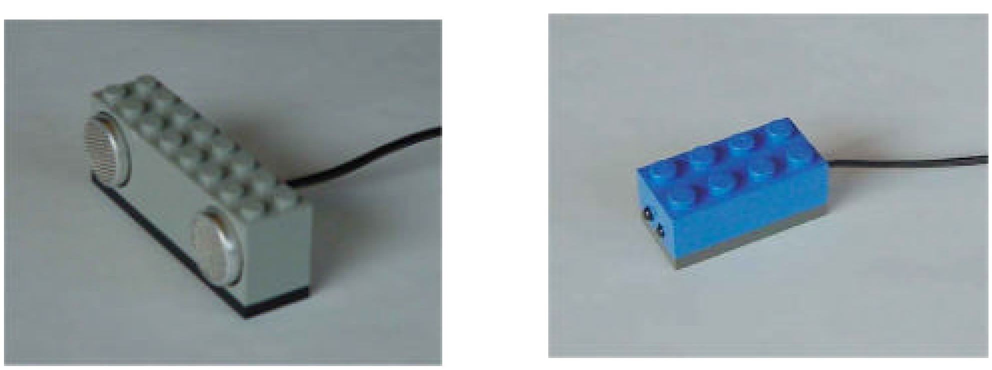


Figure 2 – Ultrasonic range sensor (left) and infrared proximity sensor (right)

The ultrasonic range sensor uses high quality ultrasonic transducers and a built-in controller to calculate the distance to the closest object or obstruction.[6] It returns a value between 0 and 100 that represents the range to the object in half inch units. Table 1 shows the operating range for the ultrasonic sensor.

Table 1 – Operating range for Tug sensors

Sensor	Upper bound [meters]	Lower Bound [meters]	Resolution [meters]
Ultrasonic	1.4	0.15	0.01
Infrared proximity	0.20	0.01	0.006

As can be seen in Table 1, the infrared proximity sensor has a much shorter range than the ultrasonic sensor, although the two ranges overlap. It is a highly sensitive sensor that uses short pulses of bright infrared light. The sensor measures the amount of infrared light reflected from a surface and returns values from 0 to 100, where zero represents no reflection detected. By using infrared pulses instead of visible light, the effect of shadows and room lighting are eliminated, thus providing more accurate readings.[6]

The third type of sensors used on the Tug robot is the touch sensor. There are two touch sensors used on a dynamic bumper built on the vehicle. These sensors detect any pressure applied on the bumper arms and return a Boolean (true or false) value to the computer.

4. EXPERIMENTAL DESIGN

In designing the experiment, the important items to be considered to make the project meet the success criterion are: (1) the size of the search space, (2) the search strategies to be used and (3) the measurement systems. Further concerns are related to safety and the use of human subjects. However, this experiment will not make use of dangerous equipment for which special safety precautions have to be taken. Although a human subject will be used for the semi-autonomous strategy, the role of the human will be solely interpretation of the sensor data and communication with the robot. The experiment thus does not need specific safety guidelines to be performed.

4.1 Design of the search space

The size of the search space is an important aspect of the experiment setup. It needs to relate to the relative sizes of the satellites in space. Furthermore, it is necessary to model the space correctly, so that the results from this experiment can be validated for the space environment. To calculate the size of the test search space, some information such as global position system (GPS) accuracy, satellite sizes and sensor ranges has to be collected. In the US Army Corps of Engineers manual [4], GPS accuracy is reported as approximately one hundred meters. Once the target satellite has been located, the space is which the Space Tug has to search is thus a sphere of radius one hundred meters, centered at the expected location of the target. Therefore, for a target satellite of size two meters, the linear size ratio of search space to target size is estimated to be fifty to one. The space transformation process is shown in figure 4.

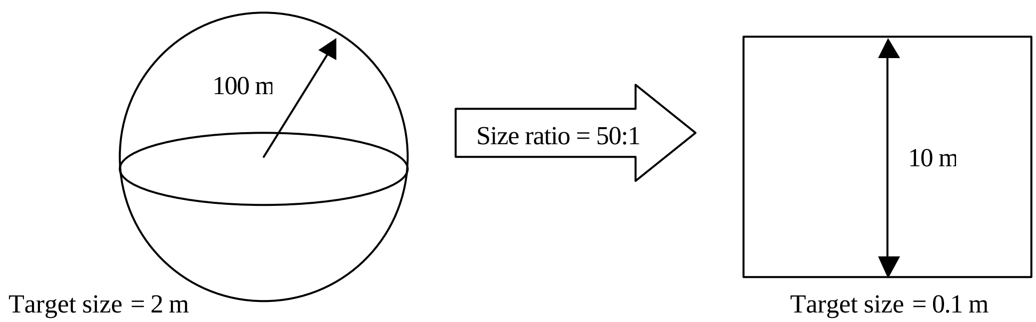


Figure 3 – Search space transformation

As can be seen above in Figure 4, the actual sphere space that the real orbital servicer will have to search is three-dimensional. However, since the target is not stationary, four variables are needed to define its position, a length, two angles and time. The space transformation involves going from four dimensions to only two. As a result, the main modeling assumption that has to be made is that the Tug is capable of putting itself in the same orbit as the target. The space is then modeled as a two-dimensional problem such that the Tug and the target are in the same orbital plane with respect to Earth. From Figure 4, it can be seen that, for a target of size 0.1 meters, the size ratio is maintained if the search space is of radius 5 meters. Since the space is designed to be a square, the sides of the search space will be 10 meters.

4.2 Search Strategies

The three search strategies are random sensor-less search, semi-autonomous with a human decision maker search and fully autonomous with sensors search. The algorithms need to be designed so that they span the space of all different strategies that could be used to find the target.

4.2.1 Random Search Strategy

The random search algorithm that will be implemented in the Space Tug robot is inspired by the "Greedy Algorithm" described by Gelenbe[2] in his paper on the autonomous search for information. It is a probabilistic search where the agent – the Tug in the experiment – is able to learn as it moves in the space. Each displacement in the space provides information to the robot. In other words, when the robot moves to a point and does not find the target, it then knows that the target is not located at that point. Its knowledge about the search space has increased.

The search space is transformed into a grid that contains a certain number of locations, as shown in Figure 5. The distance between each point has to be

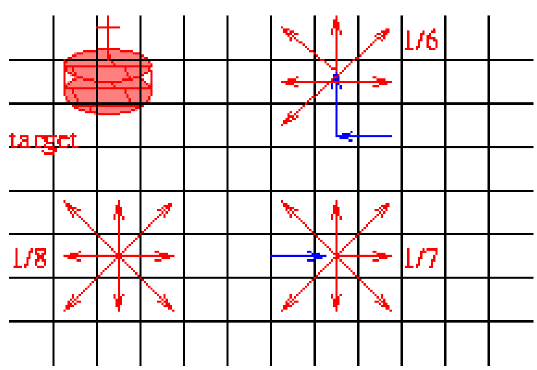


Figure 3 - Grid for random search

dependent upon the size of the Tug and the size of the target. An appropriate separation between two point given that the Tug and target sizes are approximately 20 centimeters would be of the order of two times the size of the objects, or 40 centimeters. The Tug at its starting location has eight possibilities for its next move. As can be seen in Figure 5, the probability of going to any of the eight next locations is 1/8. Once the Tug has moved, the probability associated with the location that the vehicle just left is set to zero. As a result, the Tug has now only seven possibilities for its next move. The Tug computer thus learns about the space as it moves from point to point. The search ends when the target is found, which is detected by both the touch sensors and the infrared proximity sensor – due to its extremely short range, the infrared sensor will only detect the target once the Tug is close to it.

4.2.2 Semi-autonomous Search Strategy

The basic concept for the semi-autonomous search will be that the decision-maker is a human controller. Using the on-board sensors, the human operator will move the Tug to find the target. Figure 5 shows a simplified flowchart for the procedure to be followed during the semi-autonomous search with human-in-the-loop. At the starting point, the Tug performs a 360-degrees sweep of the surroundings using the longer-range ultrasonic sensor. If the sensor does not report any presence of an object, then the human operator has to make a decision about where to move next. The operator sends a command to the Tug on-board computer, which then moves the vehicle to the next desired location. The sensor sweep is repeated, and the sensor data reported back to the human controller. Another decision is made based on the new data, and so on until the target is found. In order to transmit information, the Tug will have to align itself with command computer's infrared tower. As a result, there will be a lag between the command transmission and the Tug's move. Although the transmission will be time-consuming, it will be a good simulation of what happens with space transmission. For instance, as Morrison and Nguyen[1] describe, the Mars Pathfinder also uses waypoint navigation and delayed transmission to communicate with the Earth operator.

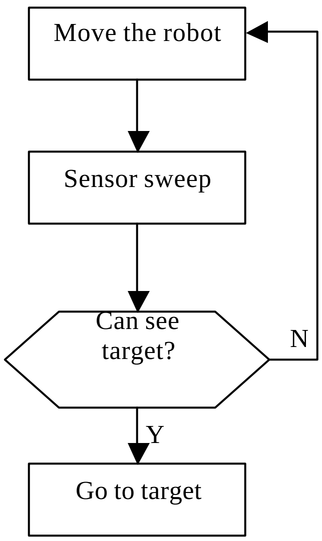


Figure 4 - Flow chart for semi-autonomous search

4.2.3 Autonomous Search Strategy

The fully autonomous search will make use of the long range ultrasonic sensor to find the target in the test space. The autonomous strategy will be based on a probabilistic model, in which the algorithm will develop a probability density function to describe the search area. Since the Tug initially will have no information about the location of the target, the probability density will be uniform across the space. As can be seen in Figure 7, the symmetry and the uniformity of the distribution places the center of mass – labeled "Cg0" – in the middle of the two-dimensional search area.

At the start of the search sequence, the Tug will travel toward the center of mass of the probability density distribution to its first waypoint. This location has to be a point in the search space close enough to the center of mass so that the latter is in range of the Tug's ultrasonic sensor. Once at its new location, the vehicle will perform a 360-degree sweep of the surroundings in an effort to locate the target. During this process, the Tug will learn about the search space. If the target is not found, the density of the swept area is set to zero. The probability density is then redistributed uniformly across the remaining space and the new center of mass is located – labeled Cg1 in Figure 7. The process just described is then repeated. Once the target is found in range of the ultrasonic sensor, the Tug vehicle will move straight toward the target for rendezvous.

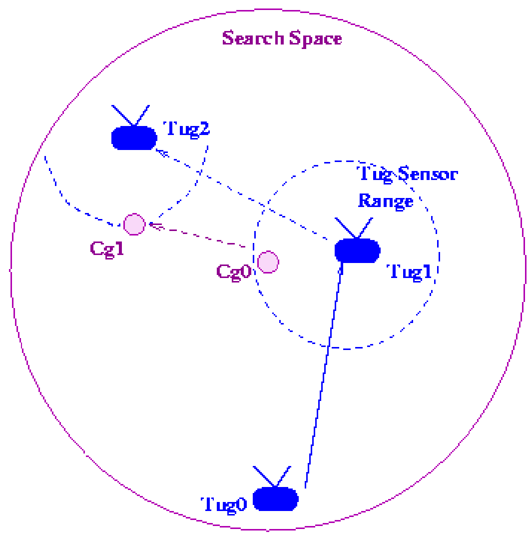


Figure 4 - Autonomous search strategy

4.3 Measurement Systems

The goal of the experiment is to develop a cost function that relates time and energy consumption during the search strategies. As such, the relevant quantities that need to be measured are the time elapsed during the search and the energy consumed from the Tug's batteries. The time data will be taken using the RCX computer's internal clock. The procedure for measuring it will be directly embedded in the software, in an effort to be as precise as possible. The energy consumed will be measured using a Texas Instruments gas gauge. The device has to be mounted in the power system of the computer to measure the starting and ending level of energy in the batteries. From this data, it is then possible to calculate the energy depleted during the search.

4.4 Sources of Error

Sources of error are associated either with measurements taken or with logical error in the coding of the search strategies. Software or logic errors in the implementation of the search strategy are systematic errors that would be hard to detect. However, a thorough and detailed debugging and testing stage for each software component will eliminate these errors. Furthermore, cross-checking of the code between the experimenters will reduce the chances of implementing a logical error in the search strategy. Efforts to eliminate these systematic errors will be particularly important for the implementation of the random and the autonomous search strategies.

Errors associated with energy and time measurements are easier to ascertain. The error in the reading of the energy level of the batteries is between ±50 and ± 150 micro volts (μV), according to the specifications of the device from Texas Instruments.[7] On the other hand, time measurements are very accurate, since it uses the Tug computer's internal clock. The latter device measures time in milliseconds and thus is precise to approximately five hundred microseconds.

An important source of error arises from the semi-autonomous search strategy. The human controller can be subject to decision-making bias in choosing the Tug's next waypoint during the search. It is of importance that the human operator has no knowledge of either the location of the target or the type of search being run. Such information about the situation will introduce a bias in the human's interpretation of the data and decision-making process. In order to eliminate this possible error, it could be necessary to use an outside person to control the Tug. The author and his partner have extensive knowledge of the search strategies and the situation and therefore would not be bias-free human controller. To reduce this effect, searches will be run with as many different human operators as possible. The results of the experiment need to be independent from the human operator. It is therefore important to eliminate the human factors effect from the tests that will be run for the semi-autonomous search. For each test run, a different decision maker will be used. The human operator will have minimal knowledge about the situation.

5. EXPERIMENTAL PROCEDURE

5.1 Test Matrix

The main variables in the experiment will be the relative position of the target as seen by the Space Tug robot, the time it takes for the Tug to find the target – which will occur when the Tug runs into the inert target – and the energy consumed in the process. Moreover, another variable will be the type of the search strategy used in the trial. As a result, the experiment will have two independent variables – the search strategy used and the relative position of the target – and two dependent variables – time and energy consumed.

The two independent variables are the only parameters that can be changed during the experiment. They will affect the time and the energy consumption but will provide the necessary data to assess the hypothesis. The search strategy used can only have three different values: random sensor-less search, semi-autonomous search with human-in-the-loop sensor interpretation, and autonomous sensor-driven search. The relative position of the target as seen by the Tug will also affect the data. For instance, if the distance between the target and the Tug is very large, then the random search could be as efficient as the semi-autonomous or fully autonomous strategies. As a result, a wide range of relative positions of the target needs to be tested in order to increase the validity of the cost function. The independent variables are shown in the two-dimensional test matrix in Table 2. There will be nine tests in the whole experiment, and for each, the time and the energy consumption will be recorded.

Table 2 – Two dimensional test matrix

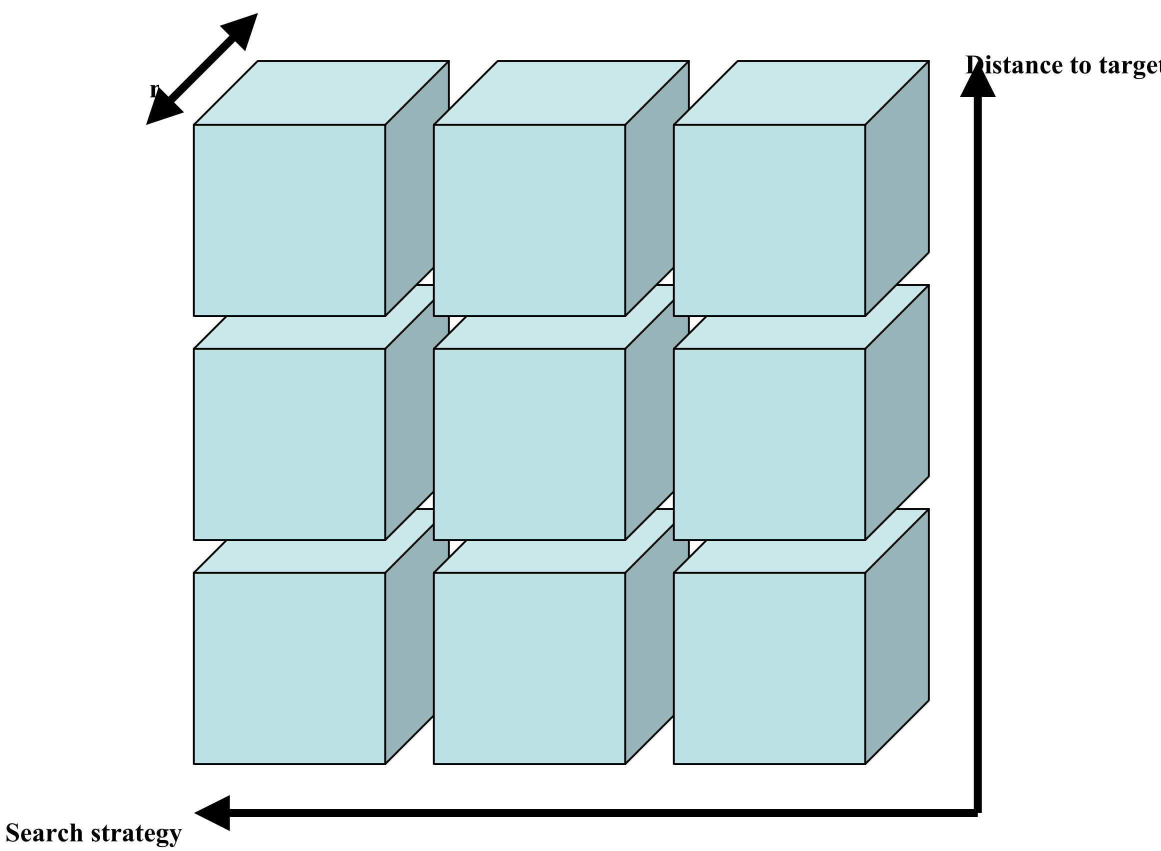
Search strategy / Relative position of target	Random sensor-less Strategy	Semi-autonomous with human-in-the-loop and sensors strategy	Autonomous sensor-driven strategy
Maximum distance - 100% of length of search space			
Medium distance - 50% of length of search space			
Short distance - 10% of length of search space			

The above test matrix does not show the number of trials that will be done for each test. To reduce the error in the results and increase the validity of the results, the number of trials needs to be determined using estimation theory. The distribution of the results is assumed to be a normal distribution, and thus the sample size, n, is defined by Hogg and Tanis[8] to be

$$n = \frac{z_{\alpha/2}^2 \sigma^2}{\varepsilon^2}, \tag{1}$$

Where σ is the sample's standard deviation, ε is the desired error and 100(1 – α)% is the confidence interval. Therefore, to determine the needed sample size, it is necessary to know the sample standard deviation. The latter can be calculated upon completing the first trials in the experiment. It is expected that the number of trials for each run will be no larger than ten for an eighty percent confidence interval. Using the number of trials, n, it is possible to construct a three-dimensional test matrix, as shown in Table 3.

Table 1 – Three dimensional test matrix



The above test matrix shows the three dimensions of the experiment: the two independent variables and the number of trials. Each box will contain a measurement of time and one of energy. Other parameters in the code, such as in the software, will not be varied between runs and therefore do not need to appear in the test matrix.

1.1 Data Analysis

Data analysis will consist of processing the raw data and plotting the points in a time-energy space. The data analysis process is shown in Figure 8. From the test matrix, the data will be collecting in a Microsoft Excel spreadsheet in order to be able to produce the time-energy plots efficiently.

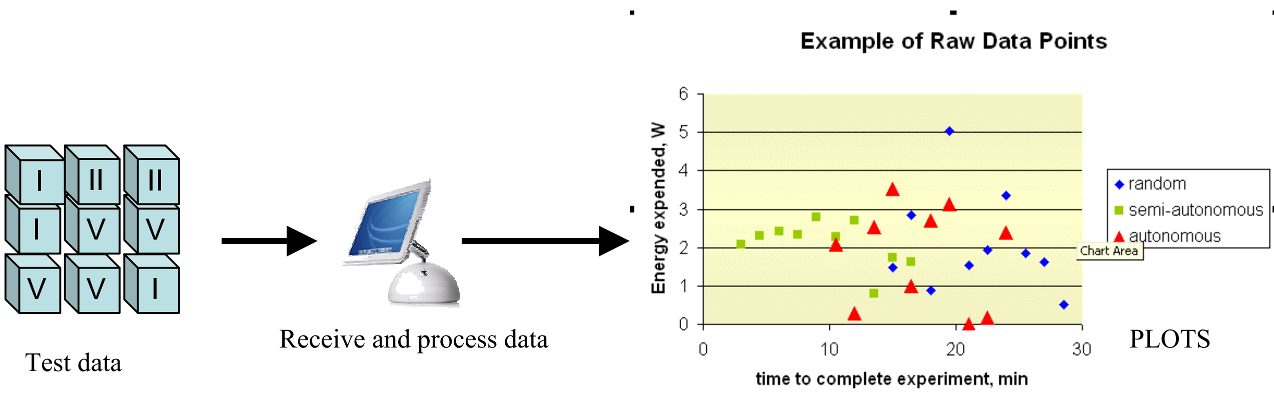


Figure 1 – Data processing chain

From the collected data, a cost function in terms of time and energy will be developed to describe the search strategies. This function will illustrate the trade-offs that exist between time and energy and could have the following form:

$$C(t,E) = (\lambda \times t)^n + [(1-\lambda)\times E]^m, \qquad (2)$$

Where t and E are the time elapsed and the energy consumed during the search, respectively. In equation (2), the constant λ as well as the exponents n and m are experimentally determined based on the trade-off between time and energy consumption.

The search strategy that minimizes the cost function – the value of C(t, E) – will be defined to be the most efficient algorithm for rendezvous and docking strategies.

6. PLANNING

6.1 16.622 Schedule

The 16.622 project is limited to one academic term and therefore needs appropriate planning. Particular care should be taken to assure that data can be acquired in the time allotted for the course during the semester. The scope of the project has to be limited in order to collect the necessary data to achieve the objectives and to fulfill the success criterion. Table 4 shows the 15-week schedule for the spring semester.

Table 4 – 16.622 schedule

Task	Start	End	Feb-03				Mar-03					Apr-03				May-03	
			3-Feb	10-Feb	17-Feb	24-Feb	3-Mar	10-Mar	17-Mar	24-Mar	31-Mar	7-Apr	14-Apr	21-Apr	28-Apr	5-May	12-May
Building Tug and target	2/3/03	2/14/03															
Coding	2/3/03	3/7/03															
Random algorithm	2/3/03	2/11/03															
emi-autonomous algorithm	2/11/03	2/19/03															
Autonomous algorithm	2/20/03	2/28/03															
Debugging and testing	2/10/03	3/7/03															
Progress review 1	2/11/03	2/11/03															
Experiment	2/24/03	4/18/03															
Random search	2/24/03	3/13/03															
Semi-autonomous search	3/14/03	4/3/03															
Autonomous search	4/4/03	4/18/03															
Oral progress report	3/4/03	3/4/03															
Progress review 2	4/1/03	4/1/03															
Analysis & presentation	4/18/03	5/13/03															
Data analysis	4/16/03	4/30/03															
Written report	4/30/03	5/13/03															
Final oral report	5/1/03	5/1/03															
Last day to take data	4/18/03	4/18/03															

The first two weeks of 16.622 will be devoted to two different tasks, building the Tug and the target and coding. However, the latter task will continue for an additional three weeks beyond the first two. The coding period includes debugging and testing and should require less than the five week periods allocated for it, assuming the software implementation runs smoothly.

Following the initial coding period, a total of eight weeks will be dedicated to running the experiment. It is expected that the random search tests can begin before the end of the coding period. There will be an overlap in the schedule. Although the current data acquisition plan will be completed at the end of the allowed data collection period, the eight-week estimate is conservative. All of the test runs should be completed in about six hours, which corresponds to five weeks in the 16.622 twelve-hour week. Therefore, this plan leaves a three-week opening with which the schedule can be adjusted to start the data analysis period earlier. Finally, the five weeks of the semester will be allocated to data analysis and final deliverables. During this period, data analysis will take place in the first two weeks before starting the final written report.

Two progress review team meetings fall during the 16.622 schedule. The first one will happen at the beginning of the coding period, and the second progress review falls toward the end of the data acquisition period. Moreover, the oral progress report is schedule to be at the beginning of the data collection period.

6.2 Required facilities and materials

Resources required to complete the experiment include the hardware – the RCX computer, the Lego parts and the sensors – which has been purchased, as well as the building material for the test area and the target. The search area will be delimited by a standard garden hose, which will be readily available from the Home Depot. Furthermore, the target will be made out of a metal can, which can be found quite easily. The gas gauge from Texas Instruments needs to be purchased before the end of the 16.621 term, in order to be able to start the data acquisition period on time. All other materials, such as the batteries, are readily available from the MIT Aero/Astro laboratory or from local stores.

The experiment is designed so that it can be setup anywhere and will not required a permanent space dedicated for the sole purpose of this project. Instead, it is planned that, for each testing session, the search space will be setup in a large enough area. Such facilities include the Johnson Athletic center or the Hangar space in the Gelb Laboratory.

6.3 Budget

The budget for this project is restricted in terms of non-hardware resources, such as consulting time with the 16.622 technical staff and the use of specialized facilities, such as the wind tunnel. The costs come entirely from the purchase of the hardware from Lego and HiTechnic, which are the suppliers for the RCX computer and the additional sensors, respectively. The detailed budget is shown in Table 5.

Table 5 – Budget

Item	Acquire from	Cost
Lego Mindstorms Computer and parts	Lego	$ 200
Ultrasonic Range sensor	HiTechnic	$ 80
Infrared proximity sensor	HiTechnic	$ 40
Touch sensor multiplexor	HiTechnic	$ 19
Battery gas gauge	Texas Instruments	$ 5
Search space building materials	Home Depot	$ 50
Rechargeable batteries	Radioshack	$ 100
	Total	**$ 494**

As can be seen above, the total expected cost of the project comes out to be $ 494. However, this is an estimate, and it is expected that the search building materials and the batteries will cost less than accounted for in Table 5. As mentioned previously, most of the hardware has already been purchased, except for the gas gauge and the building materials.

7. CONCLUSIONS

Although hazard avoidance has been extensively studied in the field of electrical engineering and computer science, the use of search and rendezvous strategies applied to aeronautics and astronautics becomes important at a time when pilots and human operators are slowly being replaced by control systems that have the ability to learn and make decisions based on their sensing abilities. The project described above will attempt to show that human decision making is still necessary when cost reduction is needed, as is the case in most space programs. Furthermore, the experiment will narrow down the different possibilities for the design of the MIT/DARPA Space Tug.

ACKNOWLEDGEMENTS

First and foremost, the author would like thank his partner, Gergana Bounova, for her help and her cooperation in the design of the project. Her precious knowledge and insights has been of great importance to the successful completion of the design phase of the Space Tug project.

Second of all, the author thanks Professor Olivier de Weck, project advisor, for his support and insights on the project. The multiple suggestions and advices that he has given to this team have helped guiding it toward an interesting experiment with great potential.

Finally, this project proposal would not have seen the light without the help and support of the 16.62x staff and faculty. The many suggestions made by the technical staff during the oral presentation has provided the necessary information to make this project proposal complete. Moreover, the feedback from the faculty, Professor Edward Greitzer, Professor Earll Murman and Jennifer Pixley, has been of great importance in the shaping of this experiment. The author would also like to acknowledge Caroline Twomey for her comments and support throughout the design phase of this project.

REFERENCES

[1] Morrison, J., Nguyen, T. " On-board Software for the Mars Pathfinder Microrover", Proceedings of the Second IAA International Conference on Low-Cost Planetary Missions, John Hopkins University Applied Physics Laboratory, Laurel, Maryland, April 1996
http://mars.jpl.nasa.gov/MPF/roverctrlnav/publications.html, NASA public site, Date accessed: 10/5/2002.

[2] Gelenbe, E. "Autonomous Search for Information in an Unknown Environment", M. Klusch, O.M. Shehory, and G. Weiss (Eds): CIA'99, LNAI 1652, pp. 47-60, 1999
http://link.springer.de/link/service/series/0558/bibs/1652/16520047.htm,
Lecture Notes in Computer Science Journal site, Date accessed: 10/4/2002.

[3] Hillenbrand, U., Hirzinger, G., "Probabilistic Search for Object Segmentation and Recognition", Proceedings European Conference on Computer Vision 2002, *Lecture Notes in Computer Science* vol. 2352, pp. 791-806, 2002.
http://cogprints.ecs.soton.ac.uk/archive/00002393/,
Cogprints site, Date accessed: 10/6/2002.

[4] "Chapter 5 – GPS Absolute Positioning Determination Concepts, Errors, and Accuracies" Engineering and Design – NAVSTAR Global Positioning System Surveying, US Army Corps of Engineers Engineering Manuals, August 1, 1996, EM 1110-1-1003. http://www.usace.army.mil/usace-docs/eng-manuals/em1110-1-1003/, US Army Corps of Engineers site, Date accessed: 09/30/2002.

[5] Lego Mindstorms Community Page.
http://mindstorms.lego.com/eng/community/resources/default.asp, Date accessed: 10/20/2002.

[6] HiTechnic Products Page. http://www.hitechnic.com/products.htm, Date accessed: 10/14/2002.

[7] Texas Instruments Products Page.
http://focus.ti.com/docs/prod/productfolder.jhtml?genericPartNumber=BQ2010, Date accessed: 11/14/2002.

[8] Hogg, R.V., and Tanis, E.A., "Estimation – Sample size," *Probability and Statistical Interference*, 6th ed., Prentice-Hall Inc., New Jersey, 2001, pp. 386-394.

[9] Bernard, D., Golombek, M. "Crater and Rock Hazard Modeling for Mars Landing", AIAA Space 2001 Conference and Exposition, Albuquerque, NM Aug. 28-30, 2001 AIAA-2001-4697

APPENDIX A

DETAILED PARTS LIST

Part	Manufacturer	Reference Number
Mindstorms Robotics Invention System 2.0	Lego	3804
Ultrasonic Range Sensor	HiTechnic	US1051
Infrared Proximity Sensor	HiTechnic	IR1021
Touch Sensor Multiplexor	HiTechnic	MX1075
Gas Gauge	Texas Instruments	BQ2010

DETAILED MATERIALS LIST

Garden hose
Rechargeable batteries