Sonar Sensing and Obstacle Detection

Gregory Dudek[1], Michael Jenkin[3], Evangelos Milios[3], David Wilkes[2]

[1] McGill Research Center for Intelligent Machines
McGill University, 3480 University St.
Montreal, Canada H3A 2A7

[2] Department of Computer Science, University of Toronto
Toronto, Canada M5S 1A4.

[3] Department of Computer Science, York University
Downsview, Canada

Abstract

The combined problems of sensor error modelling and the exploration of unknown environments are fundamental aspects of autonomous robotics. We consider these problems in the context of mapping an unknown closed environment with a sonar sensor.

This leads to three issues at different levels of abstraction: modelling the characteristics of the sensor, dealing with the unavoidable anomalies in the data obtained, and constructing a long-term representation of the environment. An important aspect of this partitioning of the problem is that it takes us from local quantitative descriptions to large-scale symbolic ones.

This paper presents algorithms for extracting obstacle contours from sonar data, that accounts for the properties of sonar modelled in our previous work. This work is then related to work on the high-level problem of constructing a map of an unknown environment.

Our model of sonar range sensing for robot navigation accounts for multiple reflections of the sonar signal between transmission and reception. This gives more realistic results than previous models. This controlled data-acquisition methodology can be subsequently used to compare techniques for the interpretation of the acquired data.

From this model of sonar sensing itself, a new model for inferring coherent reflecting surfaces is developed, based on an assumption of coherent smooth surfaces in the world. This is accomplished by using an "energy based minimization" algorithm to reject sonar measurements that are not consistent with an *a priori* world model.

Finally, we briefly discuss a representation for large scale structure of the environment that is based only on the topological interrelation between places of interest and the paths connecting them.

1 Introduction

The evolution of autonomous robotics depends critically on the ability to navigate in unknown environments. This is true not only because of the impracticality of obtaining and encoding sufficiently accurate maps of many places of interest, but also because even in well-known environments things are liable to change over time.

In the process of building and using a map of an unknown environment, an autonomous robot must also take into account the characteristics and fallibilities of its sensors. In general, this implies having a notion of what kinds of errors the sensor is prone to as well as how to deal with them. We consider map construction using sonar data. In doing so, we deal with the issue of modelling the sensors, integrating unreliable measurements and constructing a model for a large-scale environment. Although we deal with sonar range sensing, the conclusions and methodologies we present are applicable to other ranging technologies.

This paper describes an accurate and reproducible model for sonar range sensing and presents an algorithm for fitting obstacle contours to such data. We set the work in the context of our related research on sonar modelling and map building for two reasons. First, the algorithm is successful largely because it accounts well for the peculiarities of sonar range measurement, that can only be understood through careful modelling of sonar devices. Second, the algorithm forms a bridge between the low-level sonar data and our high-level exploration algorithm.

Acoustically based sensors have been used in a number of different robotic systems[2, 11]. Acoustic sensors measure the time of flight from the sensor to material in the environment. The technology is simple, inexpensive, and easily available. Sonar range sensing refers to technologies for emitting sound signals and measuring the characteristics of the echo that returns. The commonest form of sonar range sensing – the one we will address here – is based on emitting a sonar signal and measuring the time delay until an initial echo is detected. This provides a particularly simple and economical method for distance measurement. One commonly used commercial system is the Polaroid time-of-flight package. This unit utilizes the same technology that is available in Polaroid cameras. The unit sends out an ultrasonic pulse and records the delay to the reception of an echo. Given the speed of sound in air the distance to the target can be estimated. The output of a single such sensor provides limited information about the environment; by taking the output of several sensors at different locations and orientations an fairly complete estimate of the surrounding structure can be obtained.

Of the various possible sensor arrangements, the most common is a ring shaped one. Each unit is fired in sequence, and radial distances to surfaces are recovered. The platform containing the sensors can be moved between presentations, and the known motion of the sensor platform and the sonar responses at different locations are integrated into a description of the current location. By moving the robot between scans, and knowing the distance and direction of motion between scans, it is possible to integrate the measurements into a description of objects in the environment.

Such data is often idealized as a set of infinitely thin beams that return the distance to the closest surface. In reality, the finite beam thickness, the reflectance properties of the objects being sensed, and the possibilities of mutiple reflections are important issues. For example, sonar scans which are not perpendicular to an object may be reflected away from the sonar unit and lost, or they may be bounced off other objects in the environment leading to incorrect distance measurements citewilkes1990,McKerrow1990. In addition, it is not possible to completely and accurately specify the motion of a robot. For example, each time a robot moves there is some slippage between the wheels and the surface. As these measurements are compounded over a number of steps the effect of this error is cumulative. After a number of steps the error associated with early measurements can become unacceptably large.

In this paper, we begin by briefly reviewing our work on sonar modelling. Following this, we present an algorithm that behaves well in the presence of the types of error characteristic of time-of-flight sonar devices. Finally, we describe the relationship of this work to our earlier work on mapping of unknown environments.

2 Analysis of data acquisition

There are several complexities involved in the analysis of simple "range" measurements. First, the return signal detected by the system may have followed a path from the transmitter involving multiple reflections, rather than a single reflection. Second, the return signal may have originated off the aiming axis of the transmitter (since power is not concentrated along a single line). Third, diffraction may play a role in the propagation of the sound energy.

In order to study ways of dealing with such sensor errors, it has proven very useful to construct an accurate model of these unpleasant realities. We begin with a description of single-surface interactions [9] and elaborate it to deal with these other factors. The availability of a model which accurately captures these features allows sensor integration methods to be validated quickly and with data that can be *precisely replicated* from one execution to another. The result is a model that simulates and models the behavior of multiple individual sonar "rays". Sample signal paths are followed from the transmitter, through bounces and diffraction, back to the receiver.

The model is premised on several assumptions about the behavior of sonar signals in normal circumstances. First we assume predominantly specular reflection of the sonar signal from surfaces, and that floors and ceilings can safely be ignored due to the large angle of incidence. For the typical sonar frequencies, reflection from most wall surfaces are primarily specular. To account for out-of-plane structures such as floors, we may run our two-dimensional simulation on different cross-sections of the environment to recover reliable range readings. For example, the simulation could be run on one horizontal and one vertical cross-section intersecting the transducer, or (with slight modification) on several parallel cross-sections of the environment. We assume that the sonar system responds only to the first occurrence of the reception of a signal over the threshold strength, in a given trial. Also, we assume that the rangefinder incorporates an amplifier whose gain increases linearly with time to compensate for the dispersion of the signal into space (this is also consistent with commonly-used technology). Other parameters of the model include transducer impulse response and the ratio of the strengths of a specularly reflected signal and a signal diffracted from a straight edge [15].

Under the assumption that the objects reflecting the sonar pulse of wavelength λ are at distances z much larger than $\frac{a^2}{\lambda}$ from a circular transducer with radius a, the impulse response of a transducer at angle α to the sonar wavefront is given by [9]

$$h_R(t, z, a, \alpha) = \frac{2c * \cos\alpha}{\pi a * \sin\alpha} \sqrt{1 - \frac{c^2 (t - 2z/c)^2}{a^2 \sin^2\alpha}} \qquad (1)$$

for c the speed of sound in the environment, $t \epsilon \left[\frac{2z - a * \sin\alpha}{c}, \frac{2z + a * \sin\alpha}{c}\right]$, and $0 < |\alpha| \le \frac{\pi}{2}$. Note that h_R is just the delta function $\delta(t - 2z/c)$ when $\alpha = 0$ (the wavefront leaves/hits all parts of the transducer simultaneously).

Knowing the impulse response and the waveform of the sonar "chirp" allows us to determine the relative strength of each sonar ray as it travels through space, as a function of the transmission angle and reception angle. Thus, we can determine the time of flight of the first sonar ray that is sufficiently strong to trip the detection circuitry.

Below is an example run of our simulation on a simple simulated environment (figure 1). The model has been validated against simple real sonar data [15].

3 Data integration

An algorithm follows which takes sonar measurements from a ring-like collection of sonar sensors and integrates the measurements over a number of positions in order to build up a room description. The two principles motivating this approach are the need to suppress data that is due to suspected sensor artifacts and is inconsistent with an *a priori* world model, and the accumulation of information over time. The algorithm has been designed with the following features in mind.

- Sonar response errors are not well modelled by a simple white Gaussian process. In particular
  - The effect of the finite pulse width and side lobes of the sonar pulse will result in systematic errors in reported sonar distances. For example, this will result in false responses near points of high curvature (e.g. corners).
  - There is a critical angle for sonar signals which will result in partial to full signal reflection away from the sonar unit when the pulse is not sent perpendicularly to a surface. In many cases the response will not be lost, but rather will be reflected off some other structure in the environment and then back to the sonar unit, resulting in a false response. For example, right-angled corners typically have a sonar appearance identical to that of a flat wall at the distance of the corner, perpendicular to the transducer aiming direction.
- There must be some mechanism for "forgetting" older sonar measurements as the error in the position of the responses relative to the current position of the robot will grow.
- There must be some mechanism to guide the robot in its exploration task. Ideally the mechanism should act to reduce the uncertainty of the current interpretation of structure in the environment, rather than simply following a preset path.

The types of errors associated with sonar measurements are *not* well represented by normally distributed error functions. Rather the errors are very systematic and highly structured. Techniques such as fitting straight lines to the sonar data using least squares or some similar measure are likely to also fit straight lines to the errors. Rather than consider such an approach, this paper proposes to use active contours to describe structure in the environment. The version of the algorithm presented below is designed to operate in an environment that can be characterized by closed contours. This is typical of indoor enviroments or man-made outdoors ones. It uses repeated sensor scans to derive the position of structures, and it uses the current interpretation to drive the exploration process.

4 The Algorithm

The robot interacts with its environment in two ways. The robot can perform a sonar scan with its sonar ring and read the returned values from the scan, and it can move a given distance in a specified direction subject to the presence of objects in its environment. As each valid sonar point is returned, it is integrated into the global sensory map which is described below.

The goal of the algorithm is to obtain a description of objects in the environment given the results of previous sonar scans. Internally, the robot will represent its environment as a collection of closed, non-intersecting two dimensional contours $C = \{c\}$, as shown in figure 2. One of these contours, c_0, is distinguished in that it contains all of the other contours within it, and for all other contours c_i and c_j it is not the case that c_i contains or intersects c_j. At each iteration of the algorithm, the robot will move, perform a sonar scan, and then integrate the new measurements into C. Initially c_0 is set to the contour formed by joining the initial set of sonar measurements together in a radial fashion.

The algorithm also maintains an external energy surface $E(\vec{x})$. This surface is initialized to zero, and is updated as new sonar responses are recorded, or when sonar responses are aged. This energy surface is used in fitting the contours to the sonar responses. The energy surface is similar to occupancy models of representation by Elfes [7], except that there the representational surface is a probabilistic representation of the presence of structure in the environment, while in this paper the energy surface is an additive field allowing for the ability to subtract previously given support.

Whenever a sonar scan is made, the robot obtains a list of distances (and directions) to structures in the environment from the sonar units. This scan gives support for the existence of solid structure in the environment at a number of different places (where the sonar

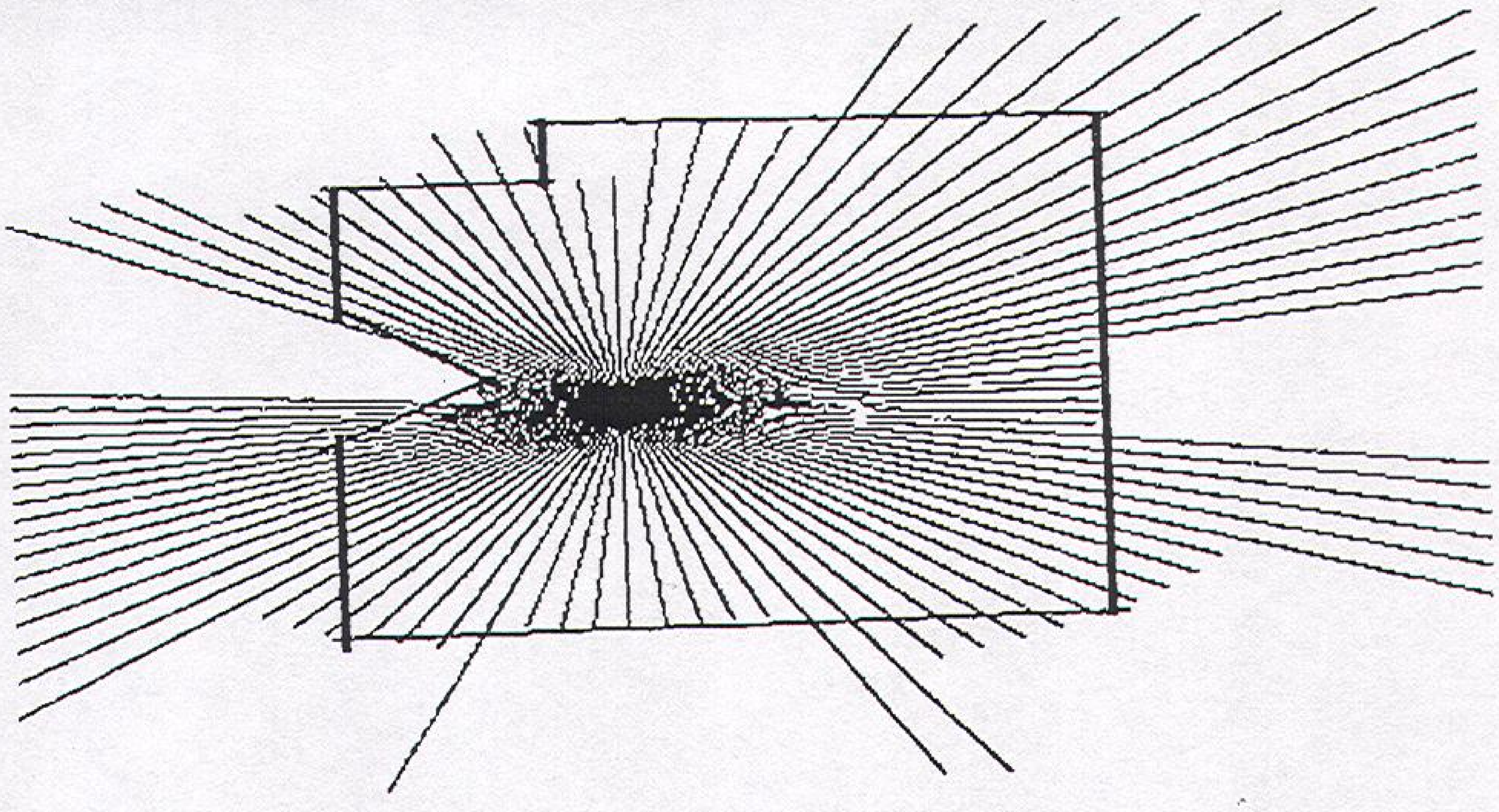


Figure 1: Simulated sonar responses

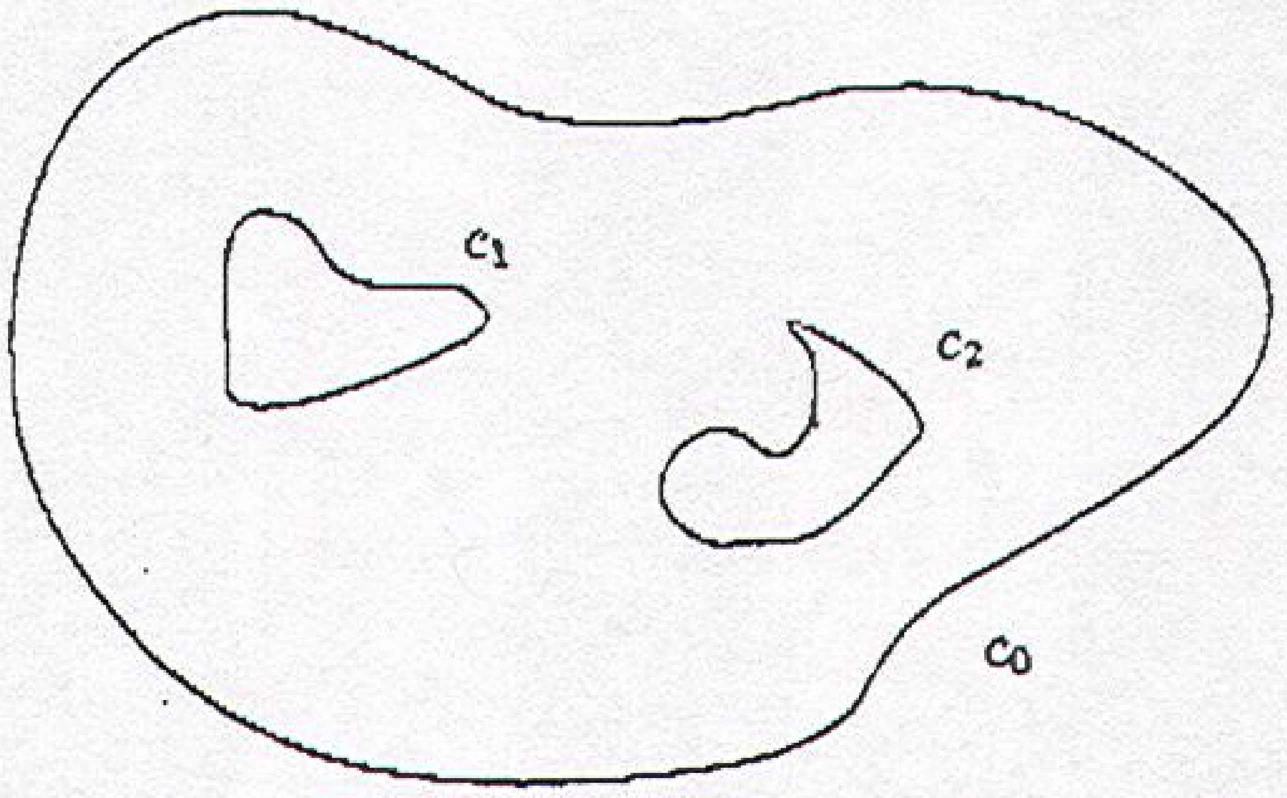


Figure 2: Contours representing the environment.

is inferred to have been reflected), as well as the presence of free space (the straight line from the robot to the reflection point). The robot integrates these two potential sources of information into the description of the environment in two ways. First, the support for true structure in the environment is integrated into an energy model which is used to "vote" for structure in the environment. Votes are "counted" in the energy surface $E(\vec{x})$. Each positive response votes by adding to the local energy surface near $\vec{x_0}$. The support is modeled as

$$I(\vec{x}) = -I_0(||\vec{x} - \vec{x_0}||^2 + 1)^{-1} \tag{2}$$

The ray from the robot to this vote region is also recorded in the energy surface as a region of positive response, indicating support for free space along this direction (see figure 3). As each sonar response is recorded, its support is added to the energy surface. After a sufficiently large number of measurements have been made, the energy surface begins to appear quite similar to the description of the room. Additional localization information (such as the free space a robot has passed through during the exploration of its environment), or the presence of obstacles, (indicated by the collision of the robot with structure in the environment), can easily be integrated into this energy surface.

As additional sonar measurements are made, they are added to the energy surface. Older sonar measurements, which are deemed to be no longer positionally accurate, can be subtracted from the energy surface. Note that this type of operation is not possible if the occupancy space is interpreted in a probabilistic manner. The basic task that the robot considers is solving this energy surface in order to obtain room and object descriptions.

The second way in which the information is integrated into the scene description is through the hypothesis of a new, unoccupied region formed by connecting the returned sonar measurements obtained by the current scan. This region is logically added to the existing free space region(s) as identified by the active contours (described below). The resulting region is typically includes erroneous outliers and the active contour nature of the representation is then used to combine this new region with existing measurements.

Active contours or snakes are a computational mech-

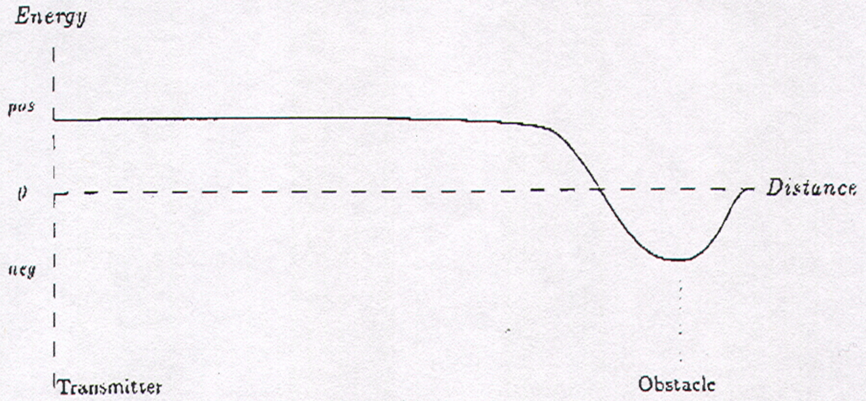


Figure 3: Energy update value as a function of distance to an obstacle.

anism for fitting a contour to an energy surface. Snakes have found applications in zero finding, object tracking, and stereopsis[8]. A snake is an energy-minimum-seeking structure composed of an ordered sequence of knots. The total energy of a snake (parameterized by $v(s) = (x(s), y(s))$ is given by[8]

$$E_{snake} = \int E_{int}(s) + E_{image}(s)\, ds \qquad (3)$$

where E_{int} represents the internal energy of the spline due to bending and E_{image} represents the image forces As the contours we are using are closed no special processing is required at the ends of the snakes. Following [8] the internal spline energy can be written

$$E_{int} = \alpha(s)|v_s(s)|^2 + \beta(s)|v_{ss}(s)|^2 \qquad (4)$$

The two terms of the internal spline energy enforce different kinds of smoothness onto the contour, distance (controlled by $\alpha(s)$) and curvature (controlled by $\beta(s)$). Either type of curvature (or both) could be used to control the form of the snake.

The E_{image} term is used to fit the active contour to the sonar points obtained by the robot. Whenever a sonar measurement is obtained, the E_{image} term is updated as described earlier. The collection of snakes S are then used as starting points for an updated object model. The snakes seek local minima in order to minimize their total energy.

There is no reason to assume that the minima selected by the snakes will be conveniently arranged so that no two contours intersect, nor is the minimum seeking process guaranteed to drive individual snakes to unique solutions. Thus the minima selected by the snakes may violate the simple non-intersection, non-containing assumptions described earlier. In order to reform the snakes back to the non-intersecting model described earlier, the regions identified by the snakes are broken down into simple non-intersecting non-containing regions, except for the outermost region s_0. This allows the snakes to identify obstacles as well as the exterior boundary of the robot's location. The process of taking the regions identified by the snakes and reducing them to simpler non-intersecting, non-containing regions is accomplished by projecting the closed regions described by the snakes onto an array, and then performing boundary following on the resulting binary image using a function that performs like Papert's turtle, which is described in [1]. Snakes which encompass extremely small locations with external E_{image} near zero are discarded.

It is also possible for this process to miss spatial regions that are small relative to the sonar pulse spacing or knot spacing. Regions which have no snake allocated to them but which have a high local E_{image} values can be easily identified and snakes inserted in order to try to identify these regions as true objects. This is not attempted in the current implementation.

The robot explores its environment by moving towards the snake point which has the largest E_{image} term. Such a point has either been interpolated (possibly over an opening), or may have been drawn away from real structure. When the robot moves towards this wall (traversing through the known empty space) and then performs a sonar scan, it updates the local energy surface. The sonar scan sent near the suspect point either hits a solid surface (which decreases the E_{image} term), or passes through it (leaving behind a large positive path) which will move the snake away from the robot. Of course, the layout of the room may be such that the robot cannot navigate directly towards this suspect point, but the process of traversing through an occupancy graph is well known and many solutions exist in the literature[13]. In the current implementation, the recovered contours are used to identify the point towards which the robot should move, and the robot moves in that direction under human control. Ongoing work seeks to automate this process. The exploration, scan, and snake updating process continues until no point exists with an E_{change} term greater than a particular tolerance. At this point the environment is said to have been explored.

As sonar information is only added to the energy surface through known sonar scans, sonar data can be easily aged by remembering the positions of scans, and then removing the additive effect of a scan from the energy surface. The snakes then are no longer under the effect of the measurement and can move accordingly. This aging process allows for older, less accurate measurements to be forgotten.

5 Sample Run

Figure 4 is a partial floorplan of the 7th floor of the Ross building at York University. Note that the floor consists of a number of hallways with central regions. Initially the robot was dropped near the elevators and allowed to roam the halls. Figure 5 shows the explored region after 10, 20, 30, and 40 moves. Note that during the exploration the robot successfully identifies the central block of the department, and that by the 40th move the floorplan obtained by the robot is essentially the floorplan of the building.

As the robot explores, support from true structure in the scene becomes an anchor for the active contours. As new regions are identified, the sonar scan pushes the snakes into unknown areas, while the existing measurements keep the snakes connected to the reality of earlier measurements.

6 Map Construction

The availability of stable descriptions for each place allows a global map interconnecting these places to be constructed. This involves being able to recognize places from the curve that describes their boundaries and can be accomplished using one of several curve recognition methods[12, 6]. For reasons of accuracy as well as computational efficiency, however, it is important to construct large-scale maps at a coarser level. Robotic exploration and map-based navigation based on dead reckoning is difficult to accomplish over large spatial scales without reference to external features of the world, due to the accumulation of error in robot position and orientation[10, 3].

In other work, we have considered the problem of exploring and mapping a world which can be described in terms of an embedding of an undirected graph $G = (V, E)$ where recognizable places are modelled be the set of vertices V and potential routes between them by the set of edges E[4, 5]. We have established that without any knowledge of distances or global coordinates a single movable place token (also described as a "marker" or "beacon") is sufficient to fully map the environment. The robotic exploration problem described here cannot be solved simply using depth or breadth-first search. The identity of individual vertices of the graph cannot be established without first solving the mapping problem. Once the graph has been explored, it can then be searched efficiently by standard techniques (and the algorithm, in fact, does this within the portion of the graph that has already been mapped out). By considering the sonar signatures of specific locations as place tokens, it would appear that it is straightforward to construct maps of arbitrary environments.

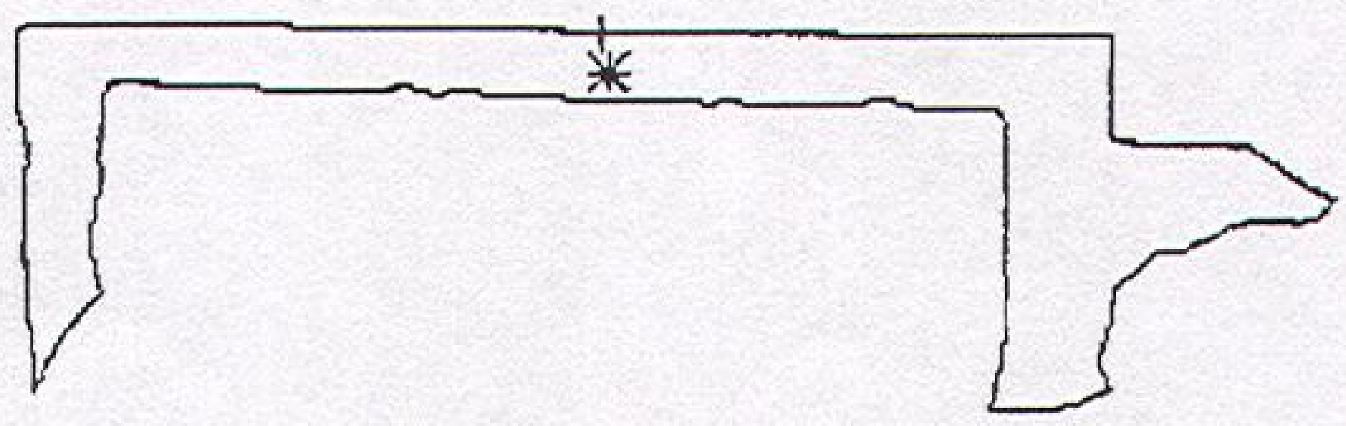
By showing that the map acquisition problem can be solved with acceptable complexity bounds under these circumstances, we demonstrate that such tasks are solvable within at least these bounds by more sophisticated systems. Furthermore, although more sophisticated perceptual mechanisms including, for example, global positional information, may be available, they are rarely completely dependable (not only must one account for sensor errors, but the sensed data is also domain dependent). Hence, even robots with powerful sensing systems may occasionally find themselves reduced to the level of the model described here.

7 Discussion

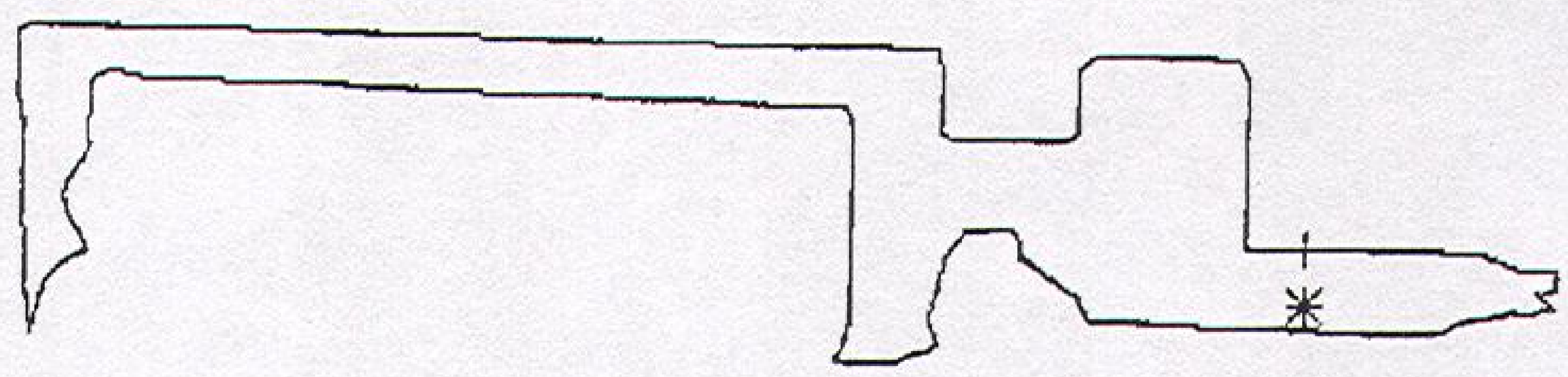
The algorithm presented in this paper is capable of recovering static closed room descriptions (including obstacles) using sonar data. The algorithm utilizes active contours or snakes in order to recover the room descriptions. The use of snakes allows the algorithm to drive the exploration process, and to be less sensitive to systematic errors in the sonar measurement process. The use of an additive energy surface allows the robot to "forget" older sonar measurements which will be suspect relative to the robot's co-ordinate system.

In order to more fully explore the limitations and advantages of this approach, the algorithm must be considered under more realistic conditions presented in this paper. The algorithm requires testing under more realistic simulated sonar data[14], and should be tested on a real sonar platform. This is the subject of ongoing research and an ongoing grant application. A number of problems and processes still require attention even in the limited domain considered here. The algorithm does not currently support full robotic control of the exploration process. The snakes are used to identify where the robot should explore next, but hu-

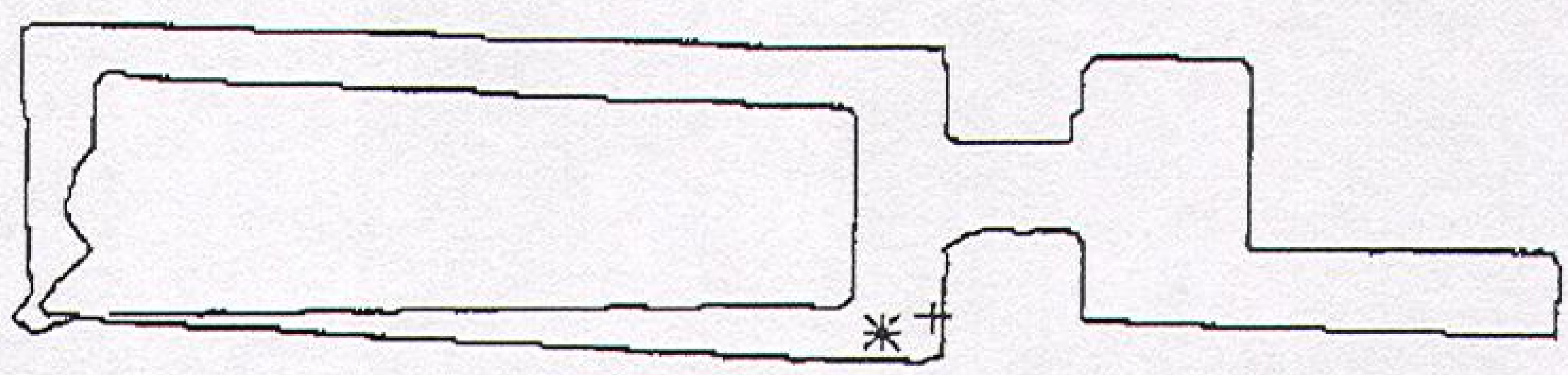
Room descriptions. Snakes are drawn as closed contours. The current location of the robot is given by a small star, and the best place to explore next is given by a small cross.



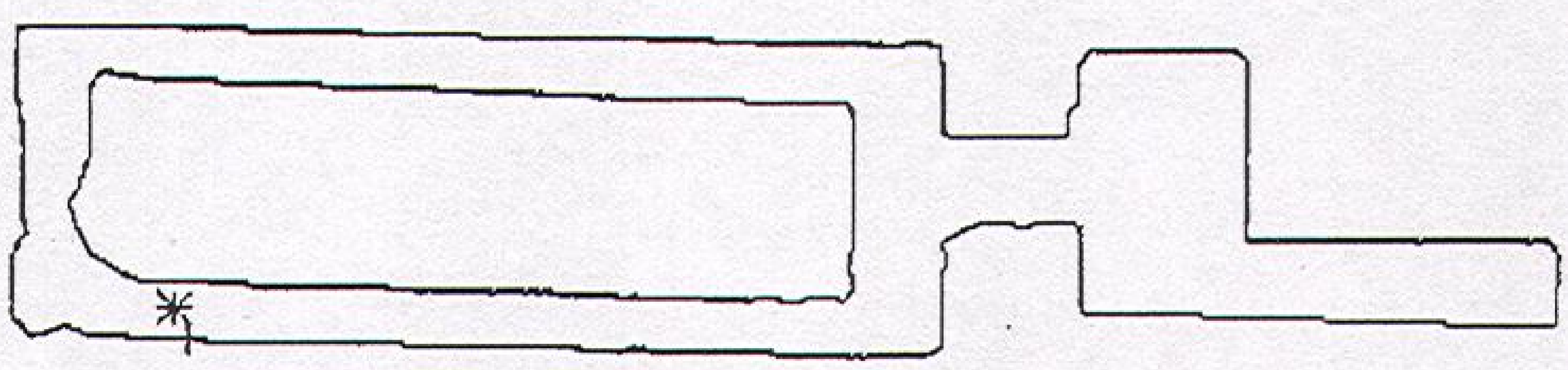
After 10 moves



After 20 moves



After 30 moves



After 40 moves

Figure 5: Room descriptions

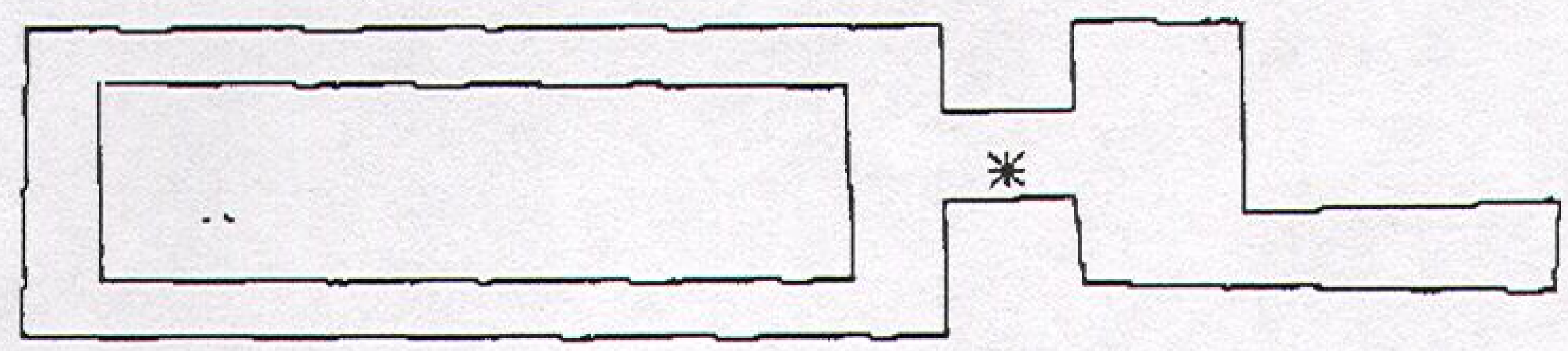


Figure 4: Input floorplan

man control is used to move the robot near this point. Small obstacles can be missed by the robot during its exploration of the environment. A process is required to "add" snakes to local energy distributions which are not associated with an existing snake.

Acknowledgements

The authors gratefully acknowledge the financial support of Natural Sciences and Engineering Research Council, the Province of Ontario Information Technology Research Center, the Government of Ontario interprovincial collaboration program and the Federal Networks of Centers of Excellence.

References

References

[1] D. Ballard and C. Brown. *Computer Vision*. Prentice Hall, Englewood Cliffs, N.J., 1982.

[2] J. Crowley. Navigation for an intelligent mobile robot. *IEEE Journal of Robotics and Automation*, 1(1):31–41, March 1985.

[3] Ernest Davis. *Representing and Acquiring Geographic Knowledge*. Pitman and Morgan Kaufmann Publishers, Inc., London and Los Altos, California, 1986.

[4] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. Using a marker to map an unknown environment. *Proceedings of Vision Interface 1989*, May 1989.

[5] Gregory Dudek, Michael Jenkin, Evangelos Milios, and David Wilkes. Robotic exploration as graph construction. *Transactions on Robotics and Automation*, 1991.

[6] Gregory Dudek and John K. Tsotsos. Shape representation and recognition from curvature. *Proceedings of the 1991 Conference on Computer Vision and Pattern Recognition*, June 1991.

[7] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer*, 22(6):46–58, June 1989.

[8] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *1st Int. Conf. Computer Vision*, pages 259–268, 1987.

[9] R. Kuc and M. W. Siegel. Physically based simulation model for acoustic sensor based robot navigation. *Pattern Analysis and Machine Intelligence*, 9(6):766–768, 1987.

[10] B. Kuipers and T. Levitt. Navigation and mapping in large-scale space. *AI Magazine*, pages 25–43, Summer 1988.

[11] H. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, pages 61–74, Summer 1988.

[12] J. Schwartz and M. Sharir. Identification of partially obscured objects in two dimensions by matching of noisy 'characteristic curves'. Robotics Research TR 46, Courant Institute, New York University, 1985.

[13] J. Schwartz and Chee-Keng Yap. *Advances in Robotics, Vol. 1: Algorithmic and Geometric Aspects of Robotics*. Lawrence Erlbaum Associates, Hillsdale, N.J., 1987.

[14] David Wilkes, Gregory Dudek, Michael Jenkin, and Evangelos Milios. The robust simulation of sonar mapping from multiple viewpoints. *Proceedings of the International Society for Optical Engineering Symposium on Advances in Intelligent Robotics Systems: Conference on Mobile Robotics V*, November 1990.

[15] David Wilkes, Gregory Dudek, Michael Jenkin, and Evangelos Milios. The simulation of sonar mapping in complex environments using multiple reflecting surfaces. *Proceedings of Vision Interface 1991*, June 1991.