

Though a convex polyhedron has been used in the numerical example, the proposed approach is, in fact, adaptable to concave polyhedra, since no special property of a convex polyhedron is used.

For a given feasible canonical grasp, in the form of a set of contact pairs, there may be infinite configurations of the grasping system. Consequently, an interesting future research direction is to exploit these solutions to select optimal grasps based on other additional criteria, expressed as new constraints on the GO model. Another important consideration, which is not dealt with in this paper, is the inclusion of other constraints on a stable grasp, mainly force equilibrium, frictional constraints, and torque constraints on joints. These constraints can be also incorporated into the same GO model, although the complexity of the model will increase.

Besides the direct application in feasibility analysis of a desired grasp, the potential applications of the proposed framework can be outlined as follows.

- Automatic generation of all canonical grasps in autonomous grasp planning for a grasping system.
- Planning and control of dexterous manipulation with finger gaiting, where contact and collision avoidance have to be considered in the procedure of switching and relocating fingers.
- Workspace analysis and computation of a multifingered hands with respect to an object, which is a research topic to be explored.
- Robotic motion planning and control considering collision avoidance in a complex environment.

REFERENCES

- [1] A. Bicchi, "On the closure properties of robotic grasping," *Int. J. Robot. Res.*, vol. 14, no. 4, pp. 319–334, 1995.
- [2] ———, "Hands for dextrous manipulation and robust grasping: A difficult road toward simplicity," *IEEE Trans. Robot. Automat.*, vol. 16, pp. 652–662, Dec. 2000.
- [3] M. Buss, H. Hashimoto, and J. Moore, "Dextrous hand grasping force optimization," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 406–418, June 1996.
- [4] J. Canny, "Collision detection for moving polyhedra," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 200–209, Feb. 1986.
- [5] A. Farahat, P. Stiller, and J. Trinkle, "On the geometry of contact formation cells for systems of polygons," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 522–536, Aug. 1995.
- [6] R. Fearing, "Implementing a force strategy for object reorientation," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1986, pp. 96–104.
- [7] B. Goeree, E. Fasse, and M. Marefat, "Determining feasible contact states of pairs of spatial polyhedra," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2000, pp. 1396–1401.
- [8] Y. Guan and H. Zhang, "Kinematic graspability of a 2-D multifingered hand," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2000, pp. 3591–3596.
- [9] J. Hong and G. Lafferriere *et al.*, "Fine manipulation with multifinger hand," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1990, pp. 1568–1573.
- [10] W. Howard and V. Kumar, "On the stability of grasped objects," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 904–917, Dec. 1996.
- [11] Y.-H. Liu, "Qualitative test and force optimization of 3-D frictional form-closure grasps using linear programming," *IEEE Trans. Robot. Automat.*, vol. 15, pp. 163–173, Feb. 1999.
- [12] ———, "Computing n-finger form-closure grasps on polygonal objects," *Int. J. Robot. Res.*, vol. 19, no. 2, pp. 149–158, 2000.
- [13] M. Mason and J. Salisbury, *Robot Hands and the Mechanics of Manipulation*. Cambridge, MA: MIT Press, 1985.
- [14] B. J. McCarragher, "Task primitives for the discrete event modeling and control of 6-DOF assembly tasks," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 280–289, Apr. 1996.
- [15] D. Montana, "Contact stability for two-fingered grasps," *IEEE Trans. Robot. Automat.*, vol. 8, pp. 421–430, Aug. 1992.
- [16] C. Ong and E. Gilbert, "Growth distances: New measures for object separation and penetration," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 888–903, Dec. 1996.
- [17] J. O'Rourke, *Computational Geometry in C*. Cambridge, U.K.: Cambridge Univ. Press, 1993.
- [18] J. Pinter, *Global Optimization in Action*. Norwell, MA: Kluwer, 1996.
- [19] J. Ponce *et al.*, "On computing four-finger equilibrium and force-closure grasps of polyhedral objects," *IEEE Trans. Robot. Automat.*, vol. 16, pp. 11–35, Jan. 1997.
- [20] K. Shimoga, "Robot grasp synthesis algorithms: A survey," *Int. J. Robot. Res.*, vol. 15, no. 3, pp. 230–266, 1996.
- [21] J. Trinkle, "On the stability and instantaneous velocity of grasped frictionless objects," *IEEE Trans. Robot. Automat.*, vol. 8, pp. 560–571, Oct. 1992.
- [22] J. Trinkle, A. Farahat, and P. Stiller, "First-order stability cells of active multi-rigid body systems," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 545–557, Aug. 1995.
- [23] J. Xiao and X. Ji, "Automatic generation of high-level contact state space," *Int. J. Robot. Res.*, vol. 20, no. 7, pp. 584–606, 2001.
- [24] H. Zhang, K. Tanie, and H. Maekawa, "Dextrous manipulation planning by grasp transformation," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1996, pp. 3055–3060.

The Arc-Transversal Median Algorithm: A Geometric Approach to Increasing Ultrasonic Sensor Azimuth Accuracy

Howie Choset, Keiji Nagatani, and Nicole A. Lazar

Abstract—This paper describes a new method for improving the azimuth accuracy of range information using conventional (Polaroid) low-resolution ultrasonic sensors mounted in a circular array on a mobile robot. Although ultrasonic sensors are fairly accurate in measuring distance in depth, they commonly have significant uncertainty in azimuth. We model this uncertainty with a uniform distribution along an arc. This means that the echo has an equal likelihood of originating from any point along the arc. We then introduce a new method to fuse sonar data to better approximate the actual obstacle location. This new method is termed the arc transversal median method because the robot determines the location of an object 1) by intersecting one arc with other arcs, 2) then by considering only "transversal" intersections, those which exceed a threshold in angle, and 3) by taking the median of the intersections. The median is a robust estimator that is insensitive to noise; a few stray readings will not affect its value. We show, via some simple geometric relationships, that this method can improve the azimuth accuracy of the sonar sensor by a specified amount under well-defined conditions. Experimental results on an ultrasonic sensor array situated on a mobile robot verify this approach.

Index Terms—Exploration, mapping, mobile robots, sonar, ultrasonic sensors.

I. INTRODUCTION

Mobile robot exploration of unknown environments motivated the work described in this paper. In our work, the robot enters an unknown environment and, relying solely on ultrasonic sensor information, it

Manuscript received December 19, 2001; revised November 16, 2002. This paper was recommended for publication by Associate Editor J. Leonard and Editor S. Hutchinson upon evaluation of the reviewers' comments. This work was supported in part by the Office of Naval Research (ONR) under Grant 97PR06977 and in part by the National Science Foundation (NSF) under Grant IRI-9702768. The work of K. Nagatani was supported by the NSF CISE Postdoctoral Program under Grant 98-08271. This paper was presented at the IEEE International Conference on Robotics and Automation, Detroit, MI, 1999.

The authors are with Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: choset@cs.cmu.edu).

Digital Object Identifier 10.1109/TRA.2003.810580

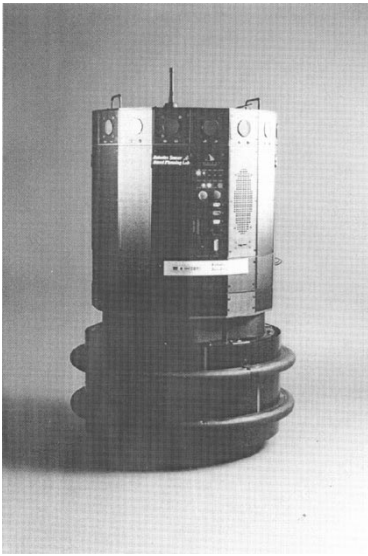


Fig. 1. Nomadic technologies mobile base.

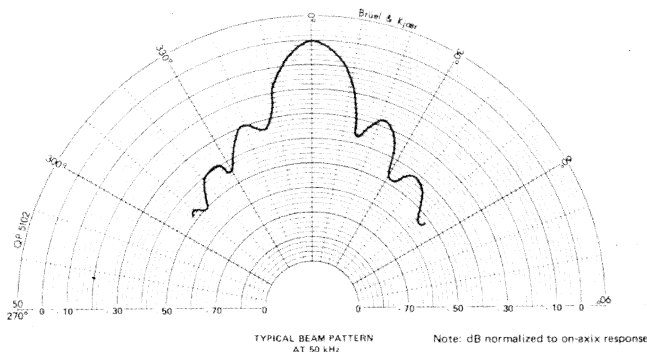


Fig. 2. Beam pattern for the Polaroid transducer installed on many mobile robots.

builds up a complete map that can be used for future excursions into the environment. We use a standard mobile robot with sixteen sonar sensors (Fig. 1). Conventional sonar sensors measure distance using time of flight. When the speed of sound in air is constant, the time that the sound requires to leave the transducer, strike an object, and return is proportional to the distance to the object. In actuality, the time is proportional to the distance to the point of reflection on the object [7]. This object, however, can be located anywhere along the perimeter of the sonar sensor's beam pattern (Fig. 2). Therefore, the distance information that sonars provide is fairly accurate in depth, but not in azimuth.

In the course of implementing our mapping method, we noticed that the low azimuth resolution of sonar sensors caused problems, especially at narrow openings. The wide beam patterns and low resolution information created the false impression that passageways were too narrow for the robot to pass through, and in extreme circumstances, that some passageways did not exist.

Using a better sensor, such as a laser range system, would help solve this problem. Laser range systems provide high resolution information, both in distance and in azimuth. Unfortunately, laser range systems that provide full 360° coverage incur a high cost, sometimes more than the mobile base itself. Moreover, the laser sensor modality does not work in all environments, such as those with glass or black-colored obstacles.

In this paper, we increase the effective azimuth resolution of the sonar sensors by appropriately moving the robot and fusing previous sonar data over time. This approach works entirely in software in real

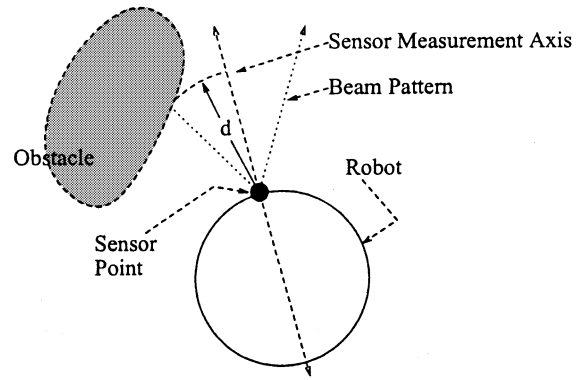


Fig. 3. Centerline model.

time, so no additional hardware modifications have to be made or purchased. This approach is called the ATM method, where the A, T, and M stand for arc, transversal, and median, respectively. Simply put, we model each sonar sensor as an arc whose radius corresponds to the range reading of the sensor; initially, the echo can originate from anywhere along the arc. We then intersect the arcs and consider only transversal (those exceeding an threshold) intersections, and then presume the median of the transversal intersections approximates the echo.

This paper begins with a review of a naive sonar model that is commonly used because it is easy to manipulate. We then discuss some prior work that improves upon this naive model and serves as a basis for the method described in this paper. We then introduce the ATM method and demonstrate the azimuth resolution improvement. Finally, we include experimental results.

II. PRIOR WORK OF SONAR MODELS

Most models of ultrasonic sensors are derived from their beam patterns (Fig. 2) which map the strength of the response of an echo as a function of angle. This beam pattern typically has a central lobe which has a strong response that dominates the rest of the pattern. Since this central lobe response is so strong, we can model the beam pattern by an arc that envelops the central lobe. The size of this arc varies with different sonar sensors, but is 22.5° for the Polaroid ultrasonic sensor that is found on most robots.

Along this arc, prior work has specified three models for the origin of the echo. The simplest is the centerline model which assumes that the echo comes from the center of the arc, and nowhere else. The other two assign Gaussian and uniform distributions along the arc, respectively. All three models do assume that at the point of reflection, the arc and obstacle are tangent to each other, in other words, the sonar echo returns to the transducer in a path perpendicular to the obstacle normal at the echo. See [12] for a thorough overview of sonar sensor use.

A. Centerline Sensor Model

Recall that a sonar sensor only provides a real number. This is all the planner knows. The sonar models described in this section provide a means for interpreting this real number. Naturally, the data correspond to distance to an echo, but the open question is: from which direction did the echo originate? In beam pattern terms, where is the echo along the sonar arc? As its name suggests, the centerline model assumes that the object is located at the center of the sonar arc. As can be seen in Fig. 3, although the echo could originate from any point along the arc, the model assumes that the echo is in the center.

The centerline model produces reasonable estimates for the locations of objects if the arc of the cone is small. The arc can be small if the sonar sensors have narrow main lobes and thus, provide well-localized information. The arc can also be small if the object is close to the sensor, i.e.,

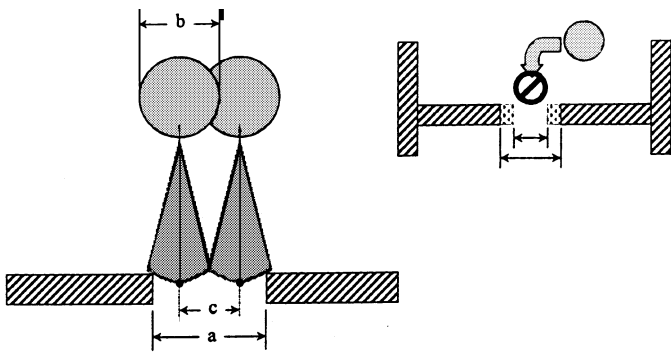


Fig. 4. Center of cone model makes the passageway seem narrower.

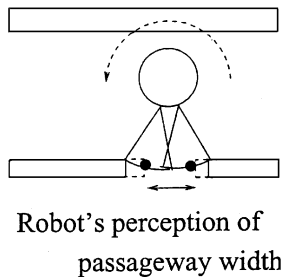


Fig. 5. Center of cone makes the passageway seem narrower.

the centerline model produces a reasonable estimate for the locations of objects that are close to the sonar sensor.

This model has the advantage of simplicity. Researchers use this simple model and a “local map” to improve the resolution of ultrasonic sensors. Moravec and Elfes introduced a type of local map called the *occupancy grid* [9], [15], which is a planar discrete representation of the robot’s environment where the value of the cell or pixel represents its occupancy status along with a certainty value: 0, unknown; $[-1, 0)$, empty; and occupied, $(0, 1]$. Borenstein’s group developed a high-speed method for updating an occupancy grid with the histogram in-motion mapping for mobile robot obstacle avoidance (HIMM) [1]. With the HIMM method, as the robot moves around the space, the center pixel of the arc is overlaid with the corresponding pixel in the occupancy grid. The information with the center pixel is then used to update the occupancy grid. Naturally, this may produce a less accurate map, but it constructs the map more quickly.

Using the standard Polaroid transducer, the centerline model breaks down quickly in approximating the location of obstacles far from the robot. In Fig. 4, the robot passes by an opening (of width a) which is wider than the diameter of the robot (of width b). The sonar cone from the downward-pointing sensor is drawn at the two robot locations, each depicted by a circle. The solid dot in the center of the arc represents the robot’s perception of where obstacles are located. Based on this perception, the robot believes the passageway (width c) is too narrow for it to pass through, i.e., $a > b > c$. See Fig. 4.

Note that, using the same reasoning as above, simply rotating the robot by half a sonar cone width and interleaving the additional center-cone measurements will *not* double the resolution of the sonar sensors and will give a similar false impression that corridors are more narrow than they are in actuality. See Fig. 5. If the sonar cones had also halved in width, then the rotate-in-place method would have worked, in theory.

Fig. 6 contains an environment with a narrow opening, similar to the examples above. In this experiment, the Nomad 200 Mobile Robot equipped with 16 sonar sensors used the naive centerline model to sample the location of objects. The dots represent the centers of the

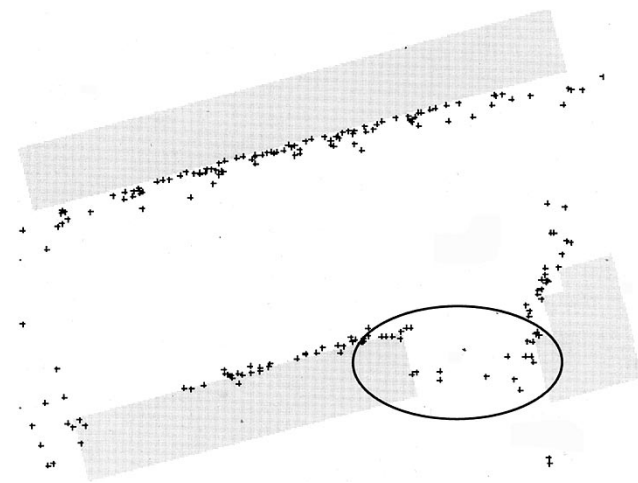


Fig. 6. Centerline method was used. The points correspond to the center of arcs from real sonar data and the light grey obstacles are drawn for the sake of display. The actual obstacles were walls outside of the lab.

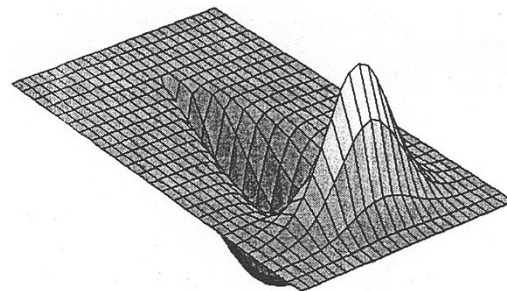


Fig. 7. Gaussian distribution along the arc of the sonar cone.

sonar arcs. This method gives the robot a false impression of the world because the robot perceives there *not* to be an opening. Others have studied the problem of negotiating narrow doorways, but Schultz first pointed out this problem to the authors [18], [19]. Also, the architecture presented in [2] handles a robot negotiating through a narrow doorway. We should emphasize here that the ATM work is not about detecting narrow doorways, but rather, providing higher azimuth resolution sensor data so that robot can perform its task, which could be detecting doorways. In our mapping work, we do not have an explicit doorway detector to achieve mapping, but rather, a sensor-based reactive controller which takes as input raw sensor data; due to the low azimuth of the raw sensor data, our reactive behaviors miss the opening, and thus, do not map the space.

B. Gaussian Distribution

We believe that Moravec and Elfes’ contribution is the occupancy grid [9], [15], as described above. However, they use a different probabilistic model to process their occupancy grid. This paper uses the sonar model to both infer the location of obstacles and free space; in other words, the model uses a planar cone with an arc base whose height is the distance at which the sonar sensor detects an obstacle. This model places a Gaussian distribution centered at the arc’s midpoint to reflect the likelihood that an obstacle is located along the arc. (See Fig. 7.) This distribution assigns a high probability that the object is located in the center of the arc. Since the likelihood of an obstacle being located in the interior of the arc is quite low, the model assigns a value close to -1 for almost all points in the interior. Finally, this model is then discretized into cells to match the grid representation of the world. Initially, all cells in the world map have a zero value, corresponding to “unknown.”

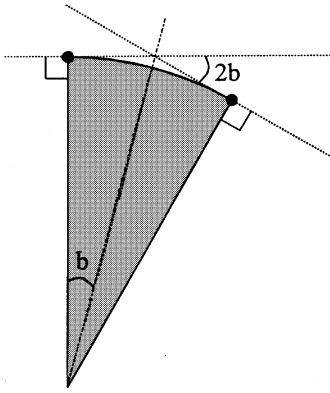


Fig. 8. Object must be tangent to the arc, but there is an infinite number of orientation possibilities. Uncertainty is the beam width.

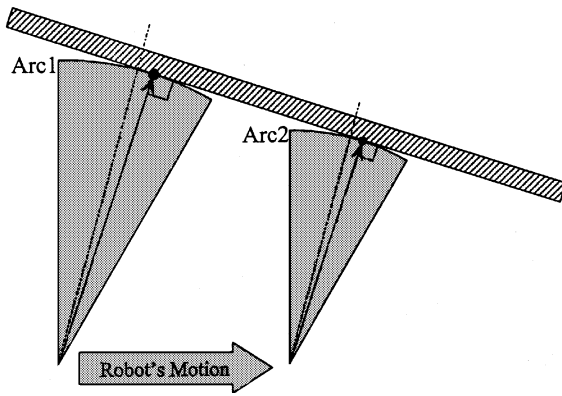


Fig. 9. Object must be tangent to Arc1 and Arc2. Now there is only one possible orientation. Intersection points are the endpoints of the desired segment.

As the robot acquires new sensor data, the cells of the world map are updated using Bayes' Rule.

C. Uniform Distribution

Already, there has been considerable success in using a Gaussian distribution to model the location of an object along a sonar arc [9]. However, documentation [7] of the widely-used Polaroid ultrasonic sensors indicates that a uniform probability distribution more accurately models the location of the echo, and hence, the reflection point of an object, along the arc of the sonar cone. We use the uniform distribution model in this paper, and we have experimentally verified it. Other researchers such as [17] also use a uniform distribution model with their grid-based approach.

McKerrow stepped away from the conventional pixel-based approaches by fitting line segments through the sonar arcs to recover edges in a polygonal environment [14]. First, McKerrow assumes that the radius of the cone is accurate but that the obstacle may lie anywhere tangent to the arc. Naturally, there are infinitely many points along the arc, and thus, infinitely many possible tangent locations for the obstacle. This range of tangent angles is the same size as the width of the sonar arc, which is of size $2b$ in Fig. 8.

To resolve this uncertainty, the robot moves and collects more sonar arcs to fit common tangents through them (Fig. 9). To determine if two arcs contain a common tangent, McKerrow derives a condition on the relative sizes of the sonar arcs and the robot's motion d . Let r_{prev} and r_{curr} be the size of two sonar arcs that originate from the same sonar sensor but at different mobile robot locations after the robot translated along a straight line segment of length d . Moreover, let a be the angle

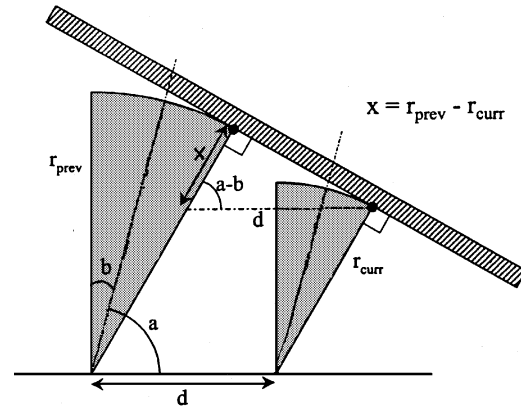


Fig. 10. One limit of a common tangent condition. Endpoints that define the closed set of acceptable radius differences are $d^* \cos(a+b)$ and $d^* \cos(a-b)$.

of the arc's center line relative to the heading of the robot, and b be the half-width of the sonar arc. See Fig. 10. If the difference $|r_{\text{prev}} - r_{\text{curr}}|$ lies within the range of values between $d \cos(a-b)$ and $d \cos(a+b)$, then there is a common line intersecting both of the arcs that is tangent to them. Note that if the obstacle intersects both arc endpoints, then this condition reduces to $|r_{\text{prev}} - r_{\text{curr}}| = d \cos(a \pm b)$. If this condition is satisfied, then determining the location of the echoes is trivial. From here, the echo points are connected by a straight line segment which is put into the map of the environment.

Leonard also uses a uniform distribution to identify corners and edges in a polygonal environment using ultrasonic sensors with the *region of constant depth* method (RCD) [13]. An RCD consists of adjacent sonar returns which have the same value (or are within an error tolerance of each other), and hence, the term, "region of constant depth." These values should all correspond to the same feature, whether it be a corner or an edge of a polygon. Again, it is worth noting that these sonar returns all reflect orthogonally from the surface of the obstacle to which distance is being measured (for corners, assume the normals fall into a convex hull of normals as described in the nonsmooth analysis literature [4], [8]). For an edge, the surface normal corresponding to the closest reading to the robot best approximates the surface normal to the edge.

Fig. 11 contains a plot of 144 sonar readings taken from a Nomad scout which houses 16 ultrasonic sensors. The Nomad rotated in place by 2.5° increments and the 16 sonar values are plotted. As can be seen in Fig. 12, the RCD when overlaid on the world looks like an arc, but corresponds to one echo point and an accurate surface normal from that point. These points are then tracked as the robot moves through the environment; if the point does not move, then the feature corresponds to a convex corner; otherwise, it could be a straight line segment which is fitted in a method analogous to McKerrow's.

The work of Wijk and Christensen [20] bears similarities to the ATM method. Wijk and Christensen lay out a sonar-based method for robot tracking. They fuse multiple sonar readings via a triangulation method to better localize the source of the echo along a sonar arc. In a sense, although the ATM method does not explicitly triangulate, the ATM method could be viewed as a probabilistic version of the triangulation-based method in [20]. Wijk and Christensen [20] use a Kalman filtering approach to then use their sonar technique to track a mobile robot.

Both the ATM approach and triangulation-based method do an excellent job of locating corners in an environment, but they are different approaches. One critical difference is that the ATM method specifically derives the improvement in azimuth resolution, which can then be used by other approaches. The ATM method is especially useful when locating narrow corridors that are far away from the robot. See Fig. 13.

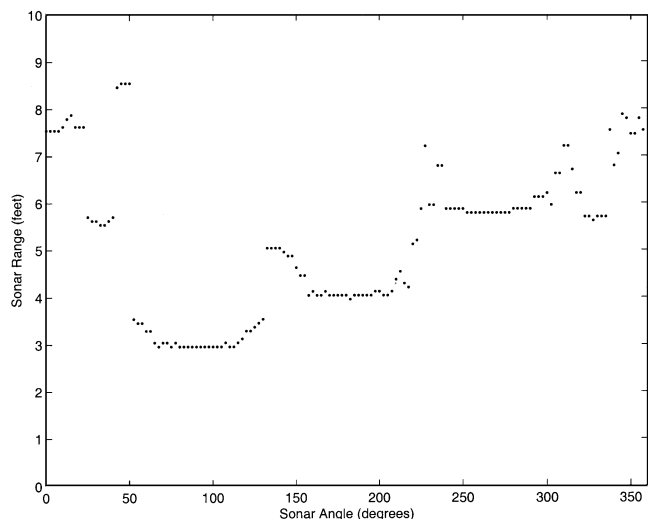


Fig. 11. Each sonar range is plotted as a dot at the angle that it was read. Note the various spans of angle for which the range remains constant. These RCDs correspond to arcs in a type of world coordinates.

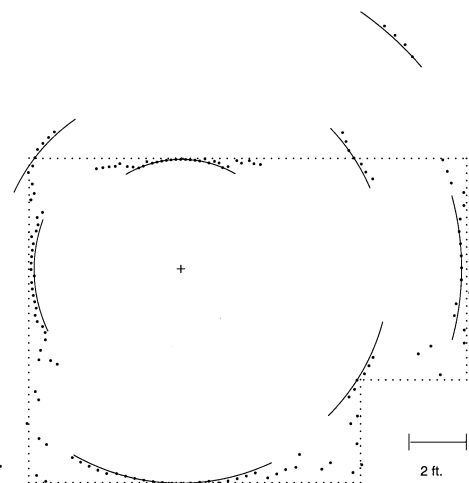


Fig. 12. The 144 sonar readings are plotted in a world frame with the cross representing the location of the robot and the dotted lines representing the hand-measured cardboard layout of the room. Arcs are plotted through regions of constant depth as determined from Fig. 11. The RCD on the top edge toward the right corresponds to a misalignment of the cardboard sheets, and the top RCD which does not lie on an edge is due to multiple reflections.

D. Processing the Echo

We should also point out that the ATM method does not explicitly detect features, such as convex corners (sometimes called edges), concave corners (sometimes just called corners), and walls (sometimes called faces). Leonard uses the RCD method to develop a voting strategy to distinguish between convex corners and walls. Kleeman and Kuc [10] present work where they use two transducers to distinguish among concave corners, convex corners, and walls. They use a notion of a virtual sensor to show that two transducers (actually, two receivers and two transmitters) are necessary and sufficient to distinguish these features. It is also worth pointing out followup work of Chong and Kleeman [3] on feature-based mapping.

Kuc also developed a biomimetic sonar system that relies on the quality of echo signal, in addition to the time of flight [11]. Instead of using time-of-flight information, his work processes the echo signal to perform object recognition. Initially, three sonar sensors are used: a center one to transmit a signal, and a left and a right sensor to receive

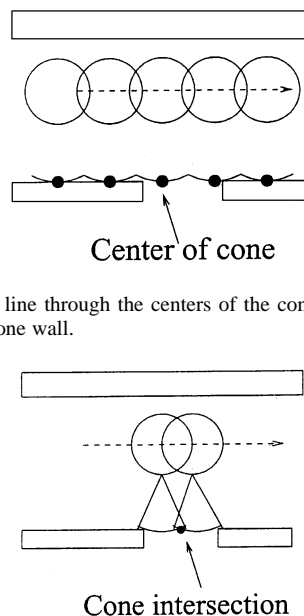


Fig. 13. Passing a line through the centers of the cones gives the robot the impression there is one wall.

Fig. 14. Intersection of cones implies that there is an object in the middle of the opening.

the signal. Based on the time of flight of the return echoes, the lateral sensors rotate inward toward an obstacle so that the echo reflecting from the object is normal to the lateral sensors. This focuses the lateral sensors on the object. Once focused, this system processes the signal and then discretizes the result into a 16-tuple vector. This vector represents the geometry of the object, and thus, can identify it. Kuc's results were successful at distinguishing among machine washers, ball bearings, O-rings, and paper clips. This work does not address the issue of mapping, but the authors believe his work can augment localization work currently underway [6].

III. ATM METHOD

Simply assuming that the echo emanates from one fixed location on a sonar arc is not sufficient. Instead, the entire arc must be considered. Being consistent with the documentation [7] of the widely-used Polaroid ultrasonic sensors, we use the uniform distribution model in the ATM approach. Using the uniform distribution model, we explain one more naive method to infer obstacle location, and build up our approach from there. Once we explain our procedure, we will demonstrate some experimental results, and then in the following section, explicitly derive the ATM method. Like others, in this paper we assume that each sonar arc corresponds to only one point of reflection, and hence, one obstacle. This method can be easily upgraded to handle the rare situation where the sonar arc corresponds to two different points of reflection.

A. Arc Intersections

Initially, we considered arc intersections of two arcs, labeled arc1 and arc2, each originating from different robot locations. The point of reflection can lie anywhere on arc1; likewise, this point can lie anywhere on arc2. If these two arcs intersect, then intuitively the point of reflection is more likely to lie at the intersection. However, considering single intersections may give the robot a false impression about the locations of objects. In Fig. 14, the robot receives individual echoes from the two lower objects. The two corresponding sonar cones intersect in the middle of the opening, and thus, assuming every possible intersection corresponds to an obstacle location is misleading.

The ATM method considers several intersections on a particular arc, one arc at a time. If many sonar sensor arcs all intersect at one point

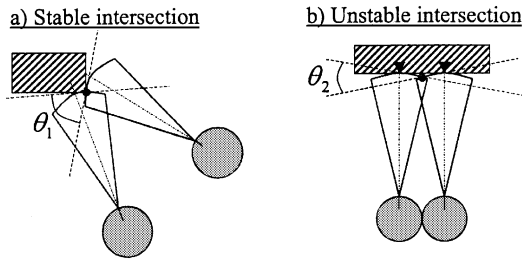


Fig. 15. Arcs with a steep intersection are transversal; these are stable because a small perturbation to one arc does not dramatically change the location of the intersection. Arcs with a gentle intersection are nontransversal; these are not stable because a small perturbation to one arc dramatically changes the location of the intersection.

on an arc, the probability of there being a point of reflection from an object is quite high, as described in Section IV-A. Due to sensor resolution in distance and slight error in dead reckoning, many arcs may not intersect exactly at one point, even if the source of the echo is constant. Instead, their intersections will form a cluster on one arc corresponding to the same point on an object. Any element of the cluster serves as an approximation to the exact location of the object along the arc.

B. Median of Intersections

One arc now has a cluster of transversal intersections. Any element of this cluster would serve as a good approximation to the location of the echo along the arc. The median of all intersections on an arc serves as the canonical element of a cluster of intersections. Using the median bypasses the need to employ an explicit clustering routine. Furthermore, the median is robust with respect to noise; it automatically eliminates bad sonar echoes and spurious intersections that correspond to other objects. These would manifest themselves in our data as outlying readings which we want to ignore.

There are some special cases that require more care when using the median. If an arc has no intersections, the robot uses the center of the arc as the location of the echo. If the arc has one intersection, then the robot uses the intersection as the location of the echo. When the arc has three or more intersections, the median operation can be applied. In the scenario where there is an even number of intersections, the robot uses the mean of the two middle values. That is, the median of $\{1, 4, 6, 10\}$ is 5. If the robot has two intersections, we currently ignore the readings.

C. Transversal Intersections

Finally, we do not consider all intersections, just those that “stably” intersect. Two arcs stably, or *transversally* intersect, if their intersection does not significantly change after one of the cones is slightly perturbed. The two arcs on the left side of Fig. 15 transversally intersect because if one cone were to be slightly perturbed, the intersection would not significantly change. On the other hand, the two arcs on the right do not transversally intersect because if one cone were to be slightly perturbed, then the location of the intersection would change. We only consider transversal intersections when computing the median. (This is a similar criterion to stereo vision or structure from motion.) Since we consider the median of only the transversal intersections, we termed our approach the ATM method.

D. Experimental Verification With a Narrow Opening

We now apply the ATM method using the same sonar data to the environment depicted in Fig. 6. Recall that the centerline approach “inflated” objects, giving the robot the false impression that there was no opening. First consider Fig. 16, which displays all of the sonar arcs and all of their respective transversal intersections, denoted by dots.

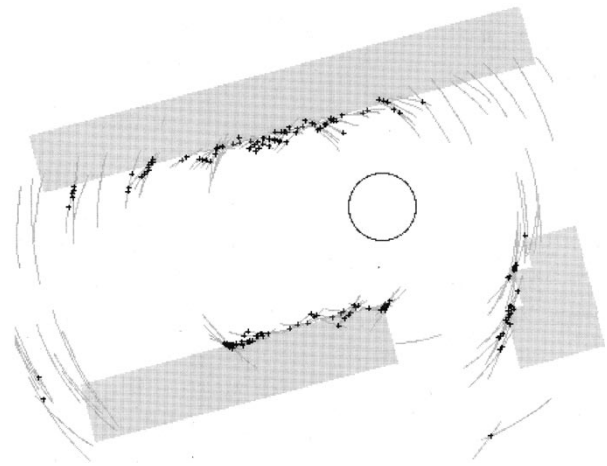


Fig. 16. All arcs corresponding to sonar readings when the robot drives down the corridor passing by the opening. Arcs correspond to real sonar sensor data and the light grey obstacles were drawn in for the sake of display. Actual obstacles were walls in the lab.

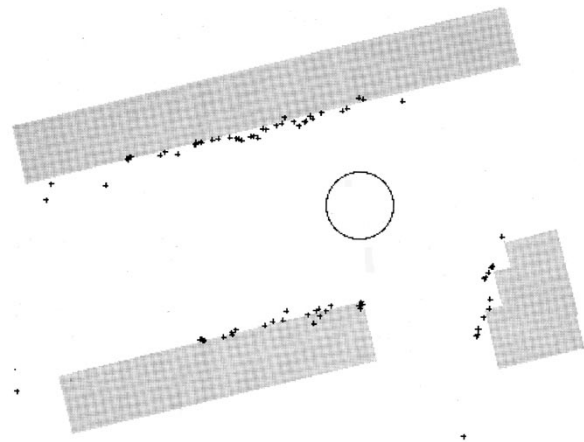


Fig. 17. Median method was used. Gray area denotes the location of objects and the plus marks represent the medians of transversal intersections.

For this experiment, we used intersections whose tangents were 30° or more. The choice of 30° is derived in Section IV-B. The dots in Fig. 17 represent the medians of each arc’s intersection points. These points represent a more accurate view of the environment, most notably at the entranceway, which was not apparent in Fig. 6. In other words, the medians of the arc intersections better approximate the real environment. In the next section, we derive the ATM method and then its sequel describes more experimental results.

IV. DERIVATION OF THE ATM

Why does this method work so well? We first give a probabilistic argument based on the properties of the uniform distribution, to confirm our intuitive sense that areas of many intersections should correspond to the location of an object (Section IV-A). The choice of 30° for the transversal angle is then derived in Section IV-B. Finally, we discuss the median (Section IV-C) and some other issues pertaining to the ATM method at the end of this section.

A. Probabilistic Reasoning: The Role of the Uniform Distribution

Given a uniform distribution on an interval of length Θ , the probability of a point landing at random within a subinterval $(d, d + \Delta)$ of length Δ is Δ/Θ . Clearly, $(\Delta/\Theta) < 1$. This probability only depends on the length of the subinterval, not its location. If we now consider the

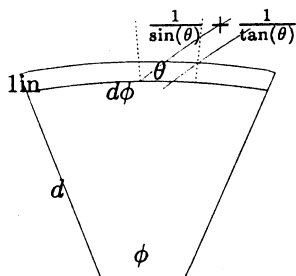


Fig. 18. The intersection of two cones is approximated by a trapezoid.

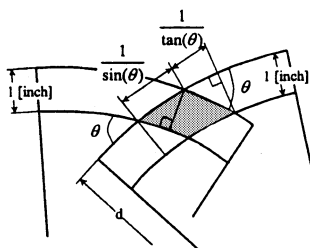


Fig. 19. The intersection of two cones is approximated by a trapezoid (closeup).

probability of n points falling into this same small interval, independently of each other, this probability can be written as

$$\prod_{i=1}^n P(X_i \in (d, d + \Delta)) = \prod_{i=1}^n \frac{\Delta}{\Theta} \quad (1)$$

which is equal to $(\Delta/\Theta)^n$. As n increases (or as Δ decreases), this probability approaches zero. In other words, as more readings fall into the same small interval, our confidence increases that there is an object that is being detected by the sonar.

In terms of statistical hypothesis testing, we have tested the null hypothesis that there is no object present against the alternative that there is an object. If the probability of the event is low, i.e., there are many intersections, then we reject the null hypothesis in favor of the alternative.

B. Transversal Intersections: Why We Used 30° in Our Experiments

In this section, we discuss our choice of using 30° as the threshold for transversal intersections. For the following calculation, we assume that each sonar arc only measures distance to one obstacle. This assumption is reasonable and we have not encountered a configuration of objects in our experiments where this assumption did not hold.

As a result of discretization in computer hardware, the sonar sensors on the Nomad robot produce range readings with a 1-in resolution, and thus, we use inches, instead of centimeters. In Fig. 18, a sonar sensor detects an object d inches away from the robot. Since we assume a uniform distribution for possible obstacle locations along the arc, this object is equally likely to lie anywhere along an arc of 22.5° (i.e., $\phi = 22.5^\circ$). Furthermore, the range resolution for the Nomadic Technologies mobile base is 1 in,¹ and thus, the object can lie in a 1-in band arc of 22.5° . The length of this arc is $d(\pi/180)22.5$ in. See Fig. 19 for a closeup view of Fig. 18.

Since we are only considering objects whose distance is significantly greater than 1 in away from the robot, the sonar band can be viewed as a 1-in by $d(\pi/180)22.5$ -in rectangular strip. The intersection of two sonar cones can then be approximated by a rhombus.

¹Note that the range resolution of the sonar sensors is much better than 1 in and that the ATM method applies to any desired range resolution. The fact that we can do so well with even poor 1-in resolution speaks well of the ATM approach.

The length of the major axis of the rhombus is an upper bound to the distance between two echoes that lie in the intersection of the two bands. To determine the azimuth range of object location for a particular sensor, we project the rhombus onto the sonar cone's arc d inches away from the sensor. The length of the projected rhombus along the arc is $(1/\sin(\theta)) + (1/\tan(\theta))$.

Therefore, when considering intersections of angle θ , the value $(1/\sin(\theta)) + (1/\tan(\theta))$ is the length of an interval along the arc where an object could be located. Note that this length does not vary with distance d . For example, if $\theta = 30^\circ$, then the sonar cone has an accuracy of 3.73 in in azimuth, regardless of obstacle distance. In other words, an object can be located in an interval of length 3.73 in centered at the intersection.

At a distance d inches for a given θ , the new resolution can be computed by

$$d\phi = n \left(\frac{1}{\sin(\theta)} + \frac{1}{\tan(\theta)} \right). \quad (2)$$

For $d = 100$ in, $\phi = 22.5^\circ$, and $\theta = 30^\circ$, $n = 10.5$. This means that at a distance of 100 in from the robot, we have, at least, a 10.5-fold improvement in resolution.

If d is small, then this range is probably longer than the arc itself, and in such situations we use the standard center of the cone approach, since that suitably approximates the location of the object.

The above analysis implies that as d increases, the resolution along the arc of the sonar sensor seemingly improves. However, as d increases, the accuracy of the sonar range does not necessarily remain fixed at ± 0.5 in. Since this accuracy changes over large distances, the 1-in accuracy in range can also become a parameter of the model.

C. Median

We project a number of trapezoids onto the arc under consideration. The trapezoids project onto intervals, not point intersections. As stated above, the maximum length of these intervals corresponds to the minimum resolution provided by the ATM method. So, in essence, we have to take the median of intervals, which is not clearly defined. If all intervals were of the same length, then we could easily take the median of the midpoints of each interval to approximate the location of the echo. However, the desired resolution (i.e., the threshold for transversal intersections) limits the variation in the size of the intervals, so taking the median is still a meaningful calculation.

D. Discussion

Detecting Flat Walls: This method does not explicitly segment the environment into corners and walls; it simply looks for echoes, i.e., data that form the boundary of the robot's free space. It can then use this data to achieve whatever task it wants, such as detecting flat walls and corners. Just like the triangulation-based fusion method described in [20], the ATM method can be used quite well to detect corners because corners have many opportunities to receive echoes from a variety of directions.

The ATM method also works well for nonspecular walls. Points on nonspecular walls can receive echoes from several angles, allowing for multiple transversal intersections to form as the robot drives along the wall. Little bumps and corners, easily detected by sonar sensors at low angles of approach, lie along the surface of nonspecular walls, as in Fig. 20. In our experiments, both inside the lab and in the halls outside our lab, we found the walls to be sufficiently nonspecular to allow for multiple arcs to intersect at several points along the wall (see Fig. 16).

The ATM method may give the robot the impression that nonspecular walls are closer than they are to the robot, but only by a small amount, on the order of the depth of the roughness of the wall surface.

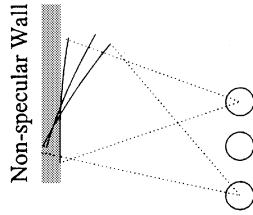


Fig. 20. As the robot passes by a point on the wall, it can collect enough arcs that intersect near the same point on the wall.

Certainly, this amount is less than 1 in, the depth resolution of the sonar sensors used in our experiments.

A specular flat wall, on the other hand, can only receive an echo when the sonar beam is normal to the wall. In such cases, the ATM method is as good as the centerline approach when passing along a wall. In other words, the robot will use the center of the cone to approximate the location of a point on the wall because the arc will not receive enough samples for the median to be meaningful.

Limitations on Accuracy Improvement: According to (2), if we increase θ , then the ATM method provides higher resolution information. Practically, there is a limit on θ . We cannot simply increase θ to achieve any desired resolution, since a higher θ threshold would result in too few intersections. With a smaller number of intersections, we have less information available about the environment. Since a 10-fold improvement in resolution for our experiments provided good results, we did not need to increase θ beyond 30° for improved resolution.

Length of Sonar History: When computing the intersections of the sonar cones, we only use the “recent” sonar sensor readings because of localization error. Keep in mind that the arc locations are with respect to the robot’s encoder coordinates. Sonar data that was acquired over a short period of time is self-consistent because the robot accrues minimal localization error, and thus, sonar readings “near” each other in time are comparable. However, sonar readings that are acquired at significantly different times are not comparable because one arc’s position has significant accumulated localization error with respect to the other.

Although the authors’ main interests are exploration and mapping of unknown static environments, using only the “recent” sensor readings makes sense in dynamic environments, where readings made in the past may not reflect the true current location of obstacles.

In our experiments, the desired improvement in resolution when detecting objects as far away as 100 in is tenfold. This leads to the 30° transversal cutoff. Therefore, for the robot to obtain an intersection at the same point 100 in away from the robot, it must move roughly $\text{len} = 50$ in, i.e.,

$$\frac{\text{len}}{2} = \sin\left(\frac{30}{2}\right) \times 100 \text{ in} = 25.8 \text{ in}.$$

Since we use a 1-s update rate for the sonar sensors and drive the robot at 5 in/s, the robot requires ten updates to move 50 in, thereby guaranteeing it will get at least one intersection. Hence, we used the ten most recent sonar sensor readings.

V. ATM APPLICATION EXAMPLE: MAPPING UNKNOWN ENVIRONMENTS

Virtually any mapping and/or localization procedures, such as [13] and [20], can use the ATM method. In this work, because of the authors’ prior experience, we applied the ATM method to a generalized Voronoi graph (GVG) mapping approach [5], [16]. Recall that the GVG in the plane is the set of points equidistant to two objects. In other words, it

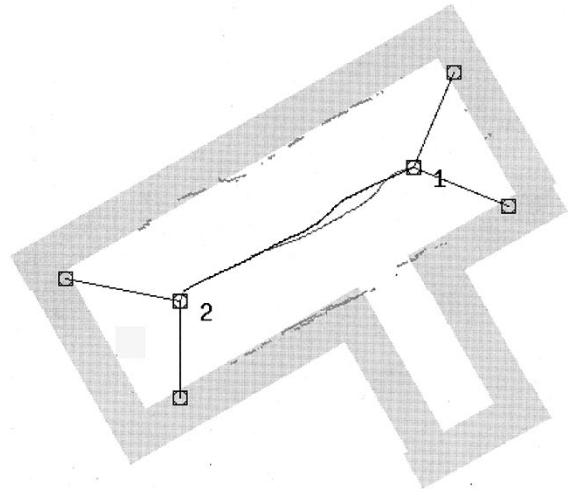


Fig. 21. Incrementally constructing the GVG using naive center line approach.

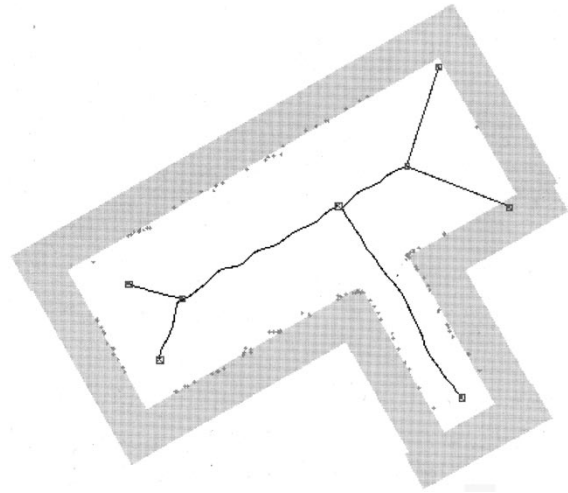


Fig. 22. Incrementally constructing the GVG using the ATM approach. Note that walls are drawn in for display purposes.

is the set of points $\{x : d_i(x) = d_j(x)\}$, where $d_i(x)$ is the distance between a point x and an object C_i , i.e.,

$$d_i(x) = \min_{c_o \in C_i} \|c_o - x\|.$$

Since the robot can use the GVG to plan paths between any two points in the environment, exploring an unknown environment is reduced to incrementally constructing the GVG using the robot’s sensors.

We conducted two experiments in the same T-shaped environment with a narrow corridor extending from the base of the T. In the first experiment, the robot used the center of the cone model when incrementally constructing the GVG and missed the opening. The gray regions in Fig. 21 correspond to the object locations, which are not *a priori* known to the robot. The dots correspond to the locations of the centers of the arcs, and thus, the robot’s understanding of obstacle locations. Notice how there are dots in the mouth of the opening, giving the robot the impression that the opening is too narrow for it pass through. Therefore, the robot did not exhaustively explore the environment and produced an incorrect GVG, represented by solid lines.

Using the ATM method (Fig. 22), the robot quite easily detected the opening and explored it. The dots represent the medians of the arcs, and thus, the robot’s understanding of the obstacle locations. The robot was successful because

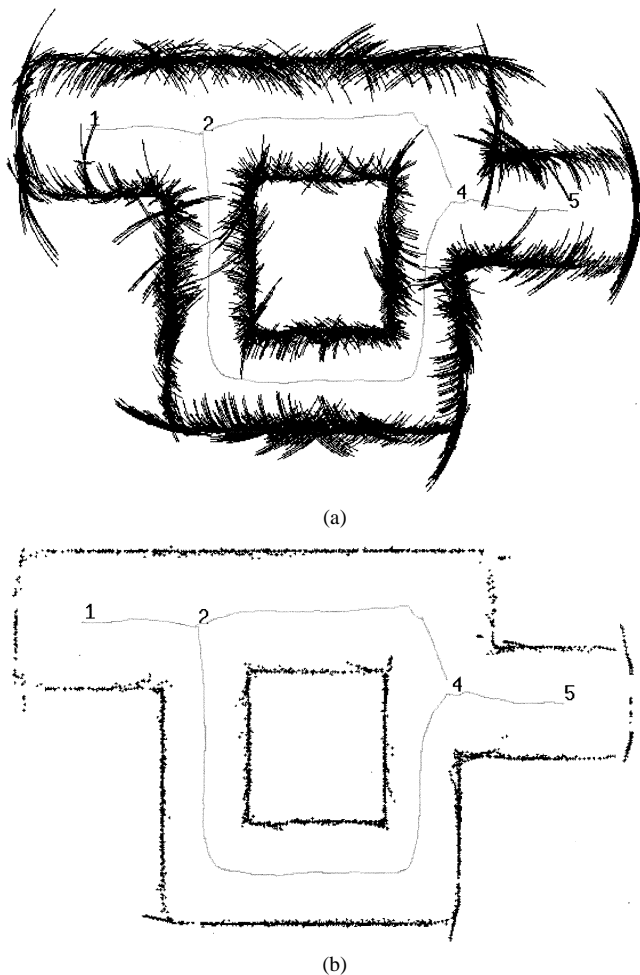


Fig. 23. (a) GVG map with the sonar arcs displayed. (b) GVG map with the ATM-processed sonar data.

the ATM method gave the robot more accurate information about its surroundings.

Fig. 23 contains another GVG map where the sonar arcs are displayed for visual purposes. The sonar data that are used to construct the GVG are displayed as arcs in the left-hand side of Fig. 23. Note that when constructing the GVG, the robot only uses the ten previous sonar arcs, i.e., only recent history of the sonar data. In other words, we store the ten previous sonar readings for each sensor in a ring buffer much in the same way [20] uses a ring buffer in their implementation. The idea is that the most recent readings are more accurate than older ones; also, we have observed, but cannot substantiate, that using a ring buffer local map has the effect of washing away specular reflections.

VI. CONCLUSION

Sensor-based exploration of unknown environments with mobile robots has motivated the work described in this paper. We focus on the issue of uncertainty of sonar sensors in azimuth. With simple sonar models, the limitation in accuracy prevents the robot from fully exploring an environment because the robot may not detect narrow openings. We address this problem by use of a uniform distribution model and develop a method of processing sonar data to improve sonar sensor accuracy. This method is called the ATM method.

We should point out that we do not model the environment with this method as is the case in [13] and [14] with polygonal representations or [9] and [15]. Our goal is to provide a higher azimuth resolution of data. Prior grid-based approaches certainly can be used to improve sonar

sensor resolution, but these methods do not explicitly derive a calculation that states how much improvement the sonars can actually achieve. In other words, certainty grid methods never justify their choice of pixel size. Likewise, the triangulation-based method [20] also improves azimuth resolution. The ATM method improves azimuth resolution and calculates by how much. In fact, the improvement in resolution can be an "input" to the ATM algorithm which allows a user to specifically designate a desired azimuth resolution improvement. The ATM method can then, in turn, serve as a front end to grid-based approaches, the RCD method, and other mapping methods that utilize sonar sensor data.

The improvement in resolution with the ATM method comes at a cost: the robot must move around its environment and the quantity of information decreases as the desired resolution increases. The first cost is negligible because the robot is moving around the environment anyway. The second cost requires more investigation. Since the ATM method assumes all readings are independent, it makes a conservative estimate about the presence of obstacles. Correlating sonar readings will increase the amount of information that is available to the robot, even at high desired resolutions.

Experiments with a mobile robot mapping an unknown environment that includes a narrow opening verify the strength of this method. However, the ATM method can be further refined. We need to determine the tradeoff between information gain and accuracy. In addition, the ATM method assumes that *most* readings are true. The median filter, by itself, takes care of spurious intersections on an arc, but there is a problem when several false readings intersect to form a "ghost" obstacle. Specular reflections and multipass echoes are both forms of false readings that can give rise to these ghost obstacles. In our experiments, the ghost obstacles formed far away from the robot, but on several occasions, the updating process of our local map automatically deleted them as the robot approached the vicinity of the false obstacles. Future work will consider this problem more carefully.

ACKNOWLEDGMENT

The authors would like to thank Dr. B. Kamgar-Parsi and Dr. T. McMullen at ONR, and Dr. J. Xiao, Dr. H. Moraff, Dr. E. P. Glinert, and Dr. L. Reeker at NSF for supporting this work. They would also like to thank T. "Tel-Tel" Yata for her insight into this work; this paper is dedicated to her memory. They thank Dr. S. Seitz for being a great guy and for his insight into this work as well. They would also like to thank B. Lisien for his experimental runs and help with the RCD description as well as R. Sakai and N. Takahashi with their help in deriving the figures and explanation for McKerrow's method. Finally, they thank M. Martin of Carnegie Mellon University, Pittsburgh, PA, for providing them with Fig. 7.

REFERENCES

- [1] J. Borenstein and J. Koren, "Histogram in-motion planning for mobile robot obstacle avoidance," *IEEE J. Robot. Automat.*, vol. 7, pp. 535–539, Aug. 1991.
- [2] J. Budenske and M. Gini, "Why is it so difficult for a robot to pass through a doorway using ultrasonic sensors?," in *Proc. IEEE Int. Conf. Robotics and Automation*, San Diego, CA, May 1994, pp. 3124–3129.
- [3] K. S. Chong and L. Kleeman, "Mobile robot map building from an advanced sonar array and accurate odometry," *Int. J. Robot. Res.*, vol. 18, pp. 20–36, Jan. 1999.
- [4] H. Choset, "Nonsmooth analysis, convex analysis, and their applications to motion planning," *Int. J. Comput. Geometry, Applicat., Special Issue*, vol. 9, pp. 447–469, 1998.
- [5] H. Choset and J. W. Burdick, "Sensor based planning, part I: the generalized Voronoi graph," in *Proc. IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 1995, pp. 1649–1655.
- [6] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM)," *IEEE Trans. Robot. Automat.*, vol. 17, pp. 125–137, Apr. 2001.

- [7] S. Ciarcia, "An ultrasonic ranging system," *Byte Mag.*, pp. 113–123, Oct. 1984.
- [8] F. H. Clarke, *Optimization and Nonsmooth Analysis*. Philadelphia, PA: Soc. Ind. Appl. Math., 1990.
- [9] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot. Automat.*, vol. RA-3, pp. 249–265, June 1987.
- [10] L. Kleeman and R. Kuc, "Mobile robot map building from an advanced sonar array and accurate odometry," *IEEE Trans. Robot. Automat.*, vol. 13, pp. 3–19, Feb. 1997.
- [11] R. Kuc, "Biomimetic sonar locates and recognizes objects," *IEEE J. Oceanic Eng.*, vol. 22, pp. 616–624, Oct. 1997.
- [12] J. J. Leonard and H. F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*. Norwell, MA: Kluwer, 1992.
- [13] J. J. Leonard, B. A. Moran, I. J. Cox, and M. L. Miller, "Underwater sonar data fusion using an efficient multiple hypothesis algorithm," in *Proc. IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 1995, pp. 2995–3002.
- [14] P. J. McKerrow, "Echolocation—from range to outline segments," *Robot. Auton. Syst.*, vol. 11, no. 4, pp. 205–211, 1993.
- [15] H. Moravec and A. Elfes, "High resolution maps for wide angles sonar," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1985, pp. 116–121.
- [16] K. Nagatani, H. Choset, and S. Thrun, "Toward exact localization without explicit localization with the generalized Voronoi graph," in *Proc. IEEE Int. Conf. Robotics and Automation*, Lueven, Belgium, May 1998, pp. 342–348.
- [17] D. Pagac, E. M. Nebot, and H. Durrant-Whyte, "An evidential approach to map-building for autonomous vehicles," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 623–629, Aug. 1998.
- [18] A. Schultz, B. Yamauchi, and W. Adams, "Integrating map learning, localization, and planning in a mobile robot," in *Proc. 1998 Conf. Computational Intelligence in Robotics and Automation*, Gaithersburg, MD, Sept. 1998, pp. 331–336.
- [19] A. C. Schultz and J. J. Grefenstette, "Using a genetic algorithm to learn behaviors for autonomous vehicles," in *Proc. Amer. Inst. Aeronautics and Astronautics Guidance, Navigation and Control Conf. (AIAA)*, Aug. 1992, pp. 739–749.
- [20] O. Wijk and H. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," *IEEE Trans. Robot. Automat.*, vol. 16, pp. 740–752, Dec. 2000.