Autonomous Robots 4, 351–367 (1997)

Improved Occupancy Grids for Map Building

KURT KONOLIGE
Artificial Intelligence Center, SRI International, 333 Ravenswood Avenue, Menlo Park, California 94025
konolige@ai.sri.com, http://www.ai.sri.com/∼konolige

Abstract. Occupancy grids are a probabilistic method for fusing multiple sensor readings into surface maps of the environment. Although the underlying theory has been understood for many years, the intricacies of applying it to realtime sensor interpretation have been neglected. This paper analyzes how refined sensor models (including specularity models) and assumptions about independence are crucial issues for occupancy grid interpretation. Using this analysis, the MURIEL method for occupancy grid update is developed. Experiments show how it can dramatically improve the fidelity of occupancy grid map-making in specular and realtime environments.

Keywords: map-making, sensor fusion, occupancy grids

1. Introduction

Working with mobile robots has forced AI researchers to confront the problem of uncertainty in sensor measurement, as they try to build environment maps using unreliable sensor readings. One of the most popular and successful methods of accounting for uncertainty is the occupancy grid method (Elfes, 1990, 1992a; Moravec and Blackwell, 1992; Moravec and Elfes, 1985). Occupancy grids divide space into a regular grid of cells (either 2D or 3D), and estimate the probability of any cell being "occupied" by a surface, based on readings from a sensor. The mathematics of occupancy grids are well-understood: in technical terms, they are "recursive estimations of a tesselated spatial random field" (Elfes, 1992a). Having said this, however, there is still a lot of work to be done in making the method work for particular applications. A fully general solution demands more information about sensor behavior and environment than is readily available; and the simpler approximations that have been suggested do not adequately address the real environments in which they are employed.

This paper examines two areas where naive assumptions of the theory are problematic: specular reflection and redundant readings. Specular reflection is a property of active time-of-flight sensors such as sonars and radars, in which the energy from the device is reflected at an angle by a surface, and reflects off multiple surfaces before returning to the device. Specular readings, unlike readings in which the beam is reflected diffusely back to the device, do not give direct information on the distance to the nearest surface.

In typical indoor environments, the problems posed by specular reflection from active time-of-flight devices such as sonars are severe. Specular reflection can also occur frequently with radar and sonar sensors in outdoor environments, especially in the presence of manmade objects such as cars, poles, or anything with flat surfaces that are not perpendicular to the sensor beam.

A second problem for realtime domains is the presence of redundant readings. The easiest assumption is that all readings give independent information about a particular cell in the occupancy grid. However, in theory and practice this assumption is violated. A sensor reading gives information about the combined probability of occupancy of a set of cells, not just a single cell. Interpreting this combined probability as a simple, independent probability for a single cell can lead to large errors in occupancy estimation.

To correct these problems of interpretation, the MURIEL method (MUltiple Representation, Independent Evidence Log) for updating occupancy grids is developed. This method is a careful refinement of the

occupancy grid mathematics along two lines. First, it splits the sensor model into two parts, a diffuse and specular model. By logging the sensor readings impinging on a cell, the model mixture can be adjusted *dynamically* for each new reading, resulting in a better estimation of occupancy. Second, to deal with the problem of independent evidence, it keeps track of the position and orientation (or *pose*) of all sensor readings at a cell. Although sensor readings from the same pose add very little independent information about occupancy at a single cell, readings from different poses do. The evidence log at a cell can be used to filter out readings that have redundant poses, thus eliminating the problem of double-counting the evidence.

The paper is divided as follows. The next section discusses previous work in map-making, and shows how the MURIEL algorithm extends occupancy grid methods. Then, we introduce the mathematics of occupancy grids, and carefully develop the theory by looking at the cases of single and multiple targets. The main contribution of this section is a novel approach that captures the main features of occupancy grid update in a simple way, and uses assumptions that are readily adapted to typical sensing environments. In contrast, previous methods either involved assumptions that were unjustified in real environments, or needed complex information about the prior distribution of objects in the environment.

After this, a section on specular models shows how to integrate gross errors in detection that accompany certain kinds of sensor readings. Then, we discuss the problem of independent evidence: how the recursive nature of the occupancy grid update function arises from the assumption of independent evidence, and how that assumption is violated for sensors with static poses. Finally, based on the analysis of these sections, the MURIEL method is introduced, and results of indoor environment experiments with sonar sensors are presented.

2. Previous Work

Work in fusing multiple sensor readings for map-making falls into two broad categories: target tracking models and occupancy grid models. In target tracking, one or more geometric features of the environment are modeled and "tracked," that is, their location is estimated at each new sensor reading. Target tracking models have been used, with impressive results, since the work of Crowley (1985, 1989) and Leonard and Durrant-Whyte (1992). This work uses the techniques of Kalman filtering, originally developed for satellite tracking, to update uncertainty estimates for the robot and targets as the robot moves and gathers information with sonar sensors.

Target-tracking methods are appropriate when there are a small number of targets, such as a few landmarks, and their interaction with the sensor is well-known, i.e., their surface reflectance and geometry. For example, Leonard and Durrant-Whyte use sonar corner reflectors as their main target, and pick out a few strong targets as landmarks for updating the robot position. The Kalman filter gives the optimal update for the robot and target positions, given noise in the sensor reading and robot position.

A key issue in the target-tracking paradigm is the data-association problem: how to identify the target that a given sensor reading is associated with. Picking the wrong target is a gross error (in contrast to noise-induced ranging and robot position errors) that can lead to divergence of the Kalman filter, where the robot is completely lost in the environment. Cox and Leonard (1994) point out the importance of this problem, and suggest a Bayesian tree approach to formulating and processing multiple hypotheses about data associations.

While target-tracking is a good method for navigation using landmarks, in many map-making situations it may be important to determine not just the position of a few landmarks, but the complete surface geometry of the environment. Obstacle-avoidance is one application; place-recognition is another. For these applications, target-tracking methods aren't appropriate, because they rely on a small set of landmarks whose geometry is specified beforehand, and can't fill in the complex, unknown surface geometry. The occupancy grid method originated by Moravec (1985) and extended by Elfes (1990, 1992a) provides a probabilistic framework for *target detection*, that is, determining whether a region of space is occupied or not. Unlike the case of target tracking, in occupancy grids the primary problem is one of data association: does a sensor reading give information about surfaces in a particular area? Although there is uncertainty in the exact range of an echo, the geometric uncertainty of the beam width (which part of the beam was reflected?) and multiple reflections dominate the range error.

Initial experiments with the occupancy grid method simply ignored geometric uncertainty, assuming that all sensor returns were simple reflections (Moravec and

Elfes, 1985), and ignoring the problem of beam width. Later, Elfes (1990, 1992a) reformulated the method as a probabilistic Bayesian updating problem using gaussian noise with a very large variance to account for the gross errors entailed by multiple reflections. He also addressed the problem of geometric uncertainty associated with sensor beam width by considering target detection under all possible configurations of the environment.

While Elfes' work represents the best current development of the occupancy grid model, it has a number of undesirable features. First, modeling multiple reflections as gaussian distributed is not realistic, since typically they give highly-correlated readings from nearby positions. Further, the use of a gaussian distribution implies an *averaging* model, in which every sensor reading is assumed to be corrupted by the same "gross error" noise. Recent work by Moravec (1992) on tuning the noise model for particular environments still treats all sensor readings the same. In fact, there are test such as Drumheller's *sonar penetration condition* (Drumheller, 1985) for estimating whether individual sensor readings are the result of multiple reflections or not.

A second problem with Elfes' framework is the practical matter of enumerating and updating probabilities for all possible environmental situations of target detection, since the number of such situations grows exponentially with the spatial area covered. In practice, given the overwhelming combinatorics of keeping track of data associations for each reading, simplyifying independence assumptions are made to reduce the computational complexity of Bayesian update (Borenstein and Koren, 1991; Matthies and Elfes, 1988). That is, each cell of space is treated as an independent target in the presence of the geometric uncertainties induced by the beam width. This leads to unrealistic estimates for target map updates, e.g., all the cells at the leading edge of the beam have their probabilities raised, when in fact usually only one cell is responsible for the echo.

The MURIEL method is a result of addressing these two problems of occupancy grid map-making. We first introduce a multiple target detection model that accounts for the typical features of occupancy grid update, the surface and freespace hypotheses, without complicated summation over all environments. The model assumes a random (unbiased) distribution of surfaces in the environment. Using this model, we derive a mixture formulation that can weight an individual sensor reading based on an estimate of single or multiple reflection. The weighting scheme makes it possible to use information about individual sensor readings to update the occupancy grid more accurately; all previous approaches used a fixed weighting for all readings.

We then focus on the problem of data association generated by the large beam widths of typical sonar and radar sensors. Instead of trying to keep track of correlations produced by each sensor reading, we ask under what conditions the multiple target detection model gives the assumed random distribution of surfaces. One answer is that sensor readings from different *poses* (positions and orientations) of the sensor give the assumed distribution. Based on this reasoning, we propose treating sensor readings as independent only if they come from different poses. In this way, the combinatorics of data association is eliminated, and the independence assumptions are made to correspond to realistic situations.

Finally, there have been some attempts to deal with dynamic objects in the occupancy grid by using temporal information to filter older readings. Borenstein and Koren (1991) introduce the Vector Field Histogramm (VFH) method. They use a spatial histogram of sonar "points," along the axis of the sonar beam, to identify areas that are likely to contain obstacles. The histogram is updated rapidly as new returns come in, and older ones are abandoned. The VFH method has the advantage that it can deal with dynamic and noisy environments; but, because it is only loosely related to probabilistic methods, it has not been used to build stable maps of an area.

3. Probabilistic Sensor Models

Figure 1 shows an abstract sensor interpretation method for gathering information about the environment. A sensor S measures some condition by transforming

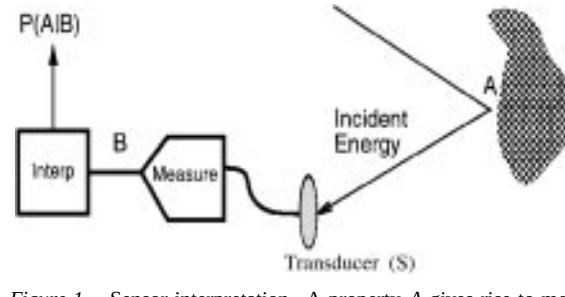


Figure 1. Sensor interpretation. A property A gives rise to measurement B, which is interpreted by considering $P(A \mid B)$.

energy (radiation, mechanical energy, etc.) into an electrical signal whose characteristics are measured (B) and interpreted to give information about some property of interest (A). In the case of time-of-flight sonars, which are the principal example in this paper, a sonar transducer emits a short pulse of sound ("ping") and then listens for the echo. The measured quantity is the time between the ping emission and echo reception, conveniently referred to as a *range reading*, that is, one-half the distance the ping traverses in that time. We write a range reading of distance D as $r = D$. The interpretation routines uses the range reading to estimate the distance of the nearest surface within the cone of the propagating ping.

A *sensor model* describes how a sensor interacts with the environment. An ideal sensor would give perfect information about the properties it reports on, but in practice there is always some uncertainty associated either with the sensor reading, or how the sensor reading relates to the quantity of interest. The uncertainty can be expressed using probabilistic methods, specifically Bayes' rule. We have:

$$P(A \mid B) = P(B \mid A)\frac{P(A)}{P(B)}, \tag{1}$$

where $P(A)$ and $P(B)$ are the prior probabilities of A and B.

The *sensor model* is the quantity $P(B \mid A)$, the probability of getting the measurement B given that the environment has property A. Normally this is determined by taking readings of the sensor in known environment states. In the case where B is a continuous quantity, the result is a conditional probability density function. The randomness introduced by the measurement process is typically considered to be gaussian, that is, for a fixed A the probability of getting a reading B will have a gaussian distribution around a mean value.

It is often inconvenient to estimate $P(B)$, and by simple probability rules Eq. (1) can be rewritten as:

$$\begin{aligned} P(A \mid B) &= \frac{P(B \mid A)P(A)}{P(B, A) + P(B, \bar{A})} \\ &= \frac{P(B \mid A)P(A)}{P(B \mid A)P(A) + P(B \mid \bar{A})P(\bar{A})}. \end{aligned} \tag{2}$$

Here, we assume we are given the prior probabilities $P(A)$ and $P(\bar{A})$ as initial information; the quantities $P(B \mid \bar{A})$ and $P(B \mid A)$ are determined experimentally or theoretically, based on the sensor characteristics.

This form of the sensor model brings out two interesting aspects of the model. First, if the sensor reading B is independent of the surface A, then the posterior probability equals the prior probability. To see this, just substitute $P(B \mid \bar{A}) = P(B \mid A)$ into the above equation, and notice that the right-hand side simplifies to $P(A)$. Second, if the sensor does not give false positives ($P(B \mid \bar{A}) = 0$), then the posterior probability will always be 1 when a reading is detected, since the only source of such a signal must be the occupied cell.

Change in odds is a often much more intuitive quantity to deal with than the absolute probability, because it factors out the priors in a nice way. To convert to an odds formulation, we use the definition:

$$\begin{aligned} O(A) &\doteq \frac{P(A)}{P(\bar{A})} \\ O(A \mid B) &\doteq \frac{P(A \mid B)}{P(\bar{A} \mid B)}. \end{aligned} \tag{3}$$

The odds of a proposition being true range from 0 (absolutely impossible) to $+\infty$ (absolutely true).

The *odds-likelihood posterior* can be computed as:

$$O(A \mid B) = \frac{P(B \mid A)}{P(B \mid \bar{A})}O(A) \doteq \lambda(B \mid A)O(A). \tag{4}$$

If the ratio $\lambda(B \mid A)$ is 1, then there is no change to the odds of A being true.

$\lambda(B \mid A)$ is called the *likelihood ratio of B given A*, and it has the same range as an odds, from 0 to $+\infty$. Using a logarithmic form gives a more natural additive scale for representing odds:

$$\log O(A \mid B) = \log \lambda(B \mid A) + \log O(A). \tag{5}$$

Here the contribution of the prior is added to a contribution from the observation.

3.1. *Occupancy Grid: Single Target Model*

For occupancy grids, space is divided into a regular grid, and the property A that we are trying to determine, for each cell, is whether there is a surface in the cell, that is, whether the cell is *occupied*. If i is the index of a cell, we'll write C_i to represent the proposition that cell i is occupied, and $\overline{C_i}$ that it is unoccupied. C_i is a binary random variable. The quantity we are interested

in is the probability of C_i, given a range reading $r = D$:

$$P(C_i \mid r = D). \tag{6}$$

$P(C_i \mid r = D)$ is a simple probability for any particular value D.

In this paper we'll treat r as a continuous variable, so that the quantities $p(r = D \mid C_i)$ and $p(r = D \mid \overline{C_i})$ are (conditional) probability density functions with the continuous range variable r.[1] Now, we can write Eq. (2) as

$$P(C_i \mid r = D) = \frac{p(r = D \mid C_i)P(C_i)}{p(r = D \mid C_i)P(C_i) + P(r = D \mid \overline{C_i})P(\overline{C_i})}. \tag{7}$$

or, in terms of the likelihood ratio,

$$\lambda(r = D \mid C_i) = \frac{P(r = D \mid C_i)}{P(r = D \mid \overline{C_i})}. \tag{8}$$

If the sensor interpretation problem is to detect a *single* target, then the above probability formulation can be applied fairly directly. We'll work out the details for the simple 1-D case first, where the occupancy grid is a linear strip along the beam axis of the sensor.

The conditional probability densities $p(r = D \mid C_i)$ and $p(r = D \mid \overline{C_i})$ can be estimated theoretically. Let's look at a cell i, whose distance from the sensor on beam axis is r_i. What is the probability of a range reading given that cell i is occupied by the target? To a first approximation, the signal detection probability density for the target is a gaussian with a peak at the distance r_i. There are two other effects that modify this initial proposal (Elfes, 1992a).

1. The range error becomes proportionally larger at increasing range.
2. The probability of detection becomes smaller at larger ranges.

Given these considerations, a mathematical model for target reflection in the 1-D case is:

$$p_1(r = D \mid C_i) = \frac{\alpha(r_i)}{\sqrt{2\pi}\delta(r_i)} e^{-(D-r_i)^2/2\delta(r_i)^2} \tag{9}$$

where the target is at distance r_i from the transducer. In this model, $\alpha(r_i)$ is the attenuation of detection with distance, $\delta(r_i)$ is the range variance (increasing with distance).

For typical sonar sensors, such as the electrostatic Polaroid models (Polaroid Ultrasonics Group, 1992), the range error is fairly small, on the order of 1%. We'll use the following conservative function:

$$\delta(r) = .01 + .015r, \tag{10}$$

which is a fixed error of 1 centimeter plus 1.5% of the range.

The detection attenuation depends in a complicated way on the cross-section of the target; we'll use a simple model that attenuates linearly with distance:

$$\alpha(r) = 0.6(1 - \min(1, .25r)). \tag{11}$$

From this equation, 4 meters is the limit of target detection, which is reasonable in typical office environments. The factor 0.6 means that targets aren't necessarily detected even at close range.

What is the probability density $p_1(r = D \mid \overline{C_i})$? Under the assumption that there is a single target, if the target isn't at cell i it can be at any other cell. The simplest assumption would be that the probability density is constant, i.e., the target has an equal chance of being detected anywhere along the beam.

$$p_1(r = D \mid \overline{C_i}) = F. \tag{12}$$

Putting all this together, Fig. 2 shows the log likelihood ratio for a sensor reading of 2 meters ($r = 2$ m).

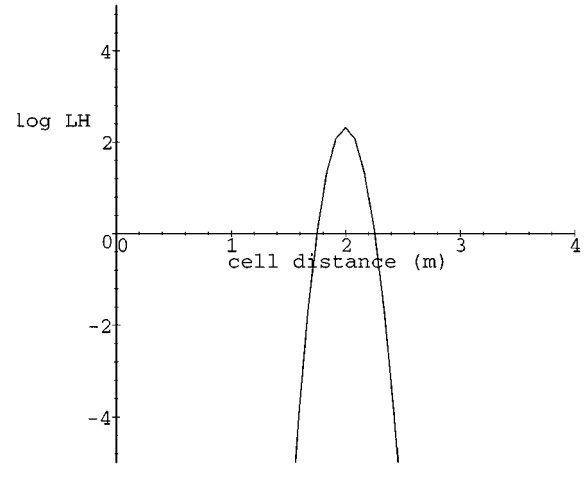


Figure 2. On-axis log likelihood ratio for a range reading of 2 meters. *X* axis is the distance r_i of the cell, *Y* axis is the log likelihood ratio used to update that cell. The range error $\delta(r)$ was made larger by a factor of 10 for this and subsequent figures to show more detail.

Cells that are in front of or behind the reading have substantially lower posterior odds ($\log \lambda < 0$); these areas are referred to as *freespace hypothesis areas*. Cells at the range reading have higher odds ($\log \lambda > 0$), as is expected; these are called *surface hypothesis areas*.

There are several interesting aspects of this figure. First, because there is a single target, the likelihood ratio falls off very quickly away from the range reading. If any range reading is returned, the target must be near that reading.

Also note the symmetry in the portions before and after the range reading. Since we're assuming there's a *single* target, a range reading at 2 m means that the target is most likely around that distance; cells on either side of it have their odds lowered. In fact, cells *behind* the sensor, which were not even scanned by the sonar beam, also have lowered odds, because the single target isn't there.

The whole point of using occupancy grids is to build up a picture of the surfaces in an environment; obviously, the single target assumption is not a good one to make in this case. The reason for presenting this unrealistic case first is to tease out the basic development of the probability model in the simplest possible way, before moving to the more complicated multiple-target case. As we show below, the biggest difference in dealing with multiple targets will be manifested in the freespace hypothesis. Cells not in the sonar beam will be unaffected by updates; and an object will tend to "shadow" other objects behind it.

3.2. *Occupancy Grid: Multiple Target Model*

In multiple target models, surfaces other than the target at cell i can reflect the sonar beam. In the simplest such model, assume that surfaces are distributed randomly so that the probability density of reflection is a small constant F. The only effect of the multiple target assumption is to add the constant F to the probability density $p(r = D \mid C_i)$:

$$p_{1m}(r = D \mid C_i) = \alpha(r_i) e^{-(D-r_i)^2/2\delta(r_i)^2} + F. \quad (13)$$

From this, the likelihood ratio must be:

$$\lambda_{1m}(r = D \mid C_i) = \frac{\alpha(r_i) e^{-(D-r_i)^2/2\delta(r_i)^2} + F}{F}. \quad (14)$$

where the target is at distance r_i from the transducer. Again, $\alpha(r_i)$ is the attenuation of detection with distance, $\delta(r_i)$ is the range variance (increasing with distance). Note that this is almost identical to the single target model (Eqs. (9) and (12)), with the exception of the extra term F in the numerator. The effect of the extra term is to make the likelihood ratio 1 (no change) for cells everywhere but in the vicinity of the range reading $r = D$. So the freespace hypothesis ($\log \lambda < 0$) has vanished in the multiple target model; why?

The answer is that, with multiple targets, detecting one target at a distance $r = D$ no longer means that there won't be other surfaces at different distances. We need to make the further distinction that the detected echo is the *first* one received[2]. To express the relevant propositions, we'll use the following notation.

$$r \not< D : \text{No return less than } D$$
$$r \mathbin{@} D : r = D \text{ and } r \not< D$$

The posterior odds of a cell being occupied, given $r \mathbin{@} D$ as the *first* range reading, is given by the likelihood ratio

$$\lambda_{1m}(r \mathbin{@} D \mid C_i) = \frac{p_{1m}(r \mathbin{@} D \mid C_i)}{p_{1m}(r \mathbin{@} D \mid \overline{C_i})}. \quad (15)$$

The sensor probability density $p_{1m}(r \mathbin{@} D \mid C_i)$ is just the probability density $p_{1m}(r = D \mid C_i)$ times the probability that no range reading was received at a distance less than D:

$$p_{1m}(r \mathbin{@} D \mid C_i) = p_{1m}(r = D \mid C_i) P_{1m}(r \not< D \mid C_i). \quad (16)$$

The likelihood ratio can be rewritten as:

$$\lambda_{1m}(r \mathbin{@} D \mid C_i) = \frac{p_{1m}(r = D \mid C_i) P_{1m}(r \not< D \mid C_i)}{p_{1m}(r = D \mid \overline{C_i}) P_{1m}(r \not< D \mid \overline{C_i})}. \quad (17)$$

To calculate the last terms, we integrate the probability density function up to the range D, and subtract it from 1:

$$P_{1m}(r \not< D \mid Q) = 1 - \int_0^D p_{1m}(r = x \mid Q) dx, \quad \text{with } Q = C_i \text{ or } \overline{C_i} \quad (18)$$

Let's give some examples of what these equations look like, to get some intuition about the probability value. First, Fig. 3 shows a probability density function for $p_{1m}(r \mathbin{@} D \mid C_i)$, for targets at 1, 2, and 3 meters. Note the primary characteristics: an initial period of

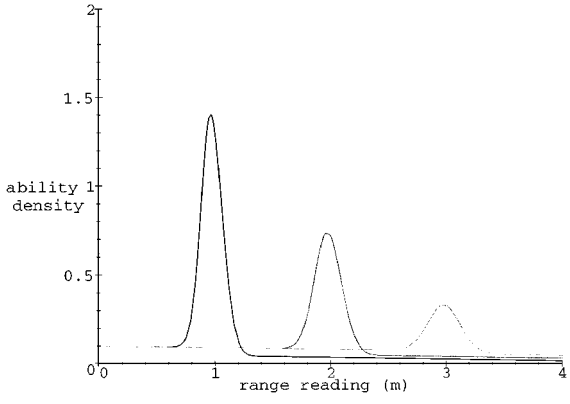


Figure 3. On-axis probability densities for targets at 1, 2, and 3 meters. *X* axis is the range reading, *Y* axis the density function. The background detection rate is similar for all three up to the target, and then falls off to a lower value, based on the distance of the target. The range variance $\delta(r)$ has been exaggerated for display.

low probability density produced by the random background targets, followed by the gaussian peak of the target at C_i, followed by a lower post-target background density. The post-target density is lower because of the shadow effect of the target. Note the difference in shadow effect between the three targets: the more likely that the target is detected (closer to the sensor), the lower the post-target detection rate.

Before giving an example likelihood function, let's see how it behaves in the extremes, when $r_i \gg D$ and $r_i \ll D$.

1. $r_i \gg D$. This is the case when updating a cell much farther from the sensor than the first range reading. The integral of Eq. (18) will be the same for C_i and $\overline{C_i}$, since $p_{1m}(r = x \mid C_i)$ is approximately F for any value $x \leq D$. So, the likelihood ratio of Eq. (17) will be very close to 1, i.e., cells farther away than the range reading D won't change their odds.
2. $r_i \ll D$. This is the case when updating a cell much closer to the sensor than the first range reading. The term $P_{1m}(r \not< D \mid C_i)$ will be less than $P_{1m}(r \not< D \mid \overline{C_i})$, because it will include the gaussian hill located at r_i. The closer the cell is to the sensor, the greater the integrated value, and the lower the likelihood ratio.

The combination of these effects can be seen in the plot of Fig. 4, which shows the log likelihood ratio for cell update, given a range reading of 2 meters. Note that the surface hypothesis peak diminishes and widens with distance from the sensor. The upward jog of the

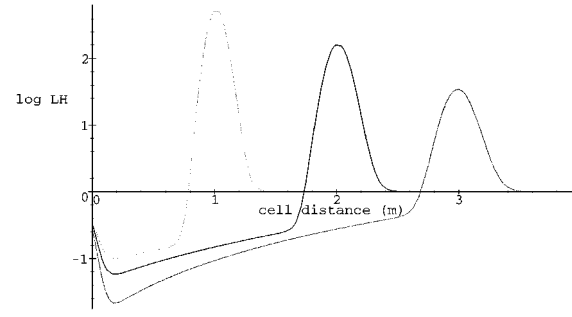


Figure 4. On-axis log likelihood ratios for range readings of 1, 2, and 3 meters. *X* axis is the cell distance, *Y* axis the log likelihood ratio used to update the cell at that distance.

graph near the origin actually mimics a problem with real sensors sonar and radar sensors: targets very close to the sensor aren't detected because they return an echo before the sensor has finished transmitting.

We can briefly compare the multiple target approach presented here with previous approaches. As we discussed earlier, these are of two forms.

1. Decomposition methods (Elfes, 1990, 1992b). This is a theoretically-motivated method that relies on a decomposition of the quantity $p(r = D \mid C_i)$ into a summation over all possible configurations $C(i)$ where cell i is occupied, using Kolmogorov's theorem:

$$P(r = D \mid C_i) = \sum_{C(i)} P(r = D \mid C(i)) P(C(i)). \tag{19}$$

   In some simple cases, the quantities $P(C(i))$ and $P(r = D \mid C(i))$ can be calculated theoretically. Otherwise, they must be estimated by performing experiments in the actual environment. Unlike the multiple target model presented here, which has a few parameters that can be adjust for different environments, there is no simple way of estimating the decomposition model given a new environment.
2. Simplifying assumptions. For example, (Matthies and Elfes, 1988) makes the assumption $p(r = D \mid \overline{C_i}) = 1 - p(r = D \mid C_i)$, which is clearly not the case for multiple targets, although it may be reasonable under the single target assumption.

3.3. *Cone Model*

We close this section by giving the 2-dimensional version of the multiple target model[3]. To calculate

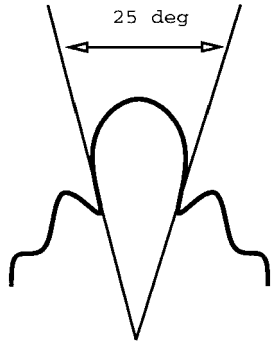


Figure 5. On-axis beam pattern for the Polaroid instrument-grade electrostatic transducer at 50 kHz. The curve represents equipotentials of the sound energy level.

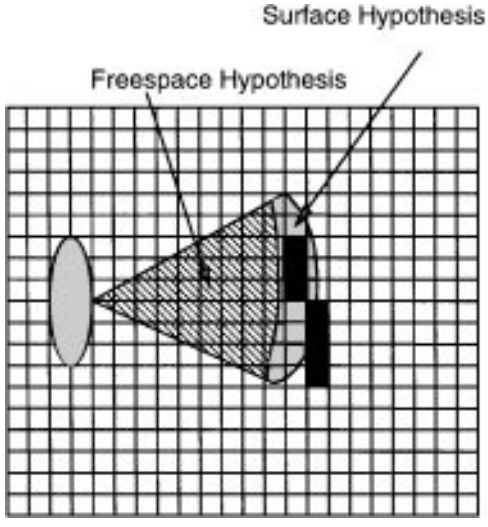


Figure 6. Sonar sensor interpretation.

the conditional probabilities, we'll use a mathematical model of the behavior of the sonar sensor, drawn from the Polaroid specifications (Polaroid Ultrasonic Group, 1992) and other literature. Sonar transducers put out an energy pattern that looks something like Fig. 5. There is a main lobe whose width depends on a number of factors, including the transducer element size and the frequency of the ultrasonic pulse. For Polaroid electrostatic transducers, the nominal width of the beam at 50 kHz is about 24 degrees. Although there are significant side lobes at up to 0.5 meters, we'll approximate the beam pattern by a cylindrical cone.

Figure 6 shows a typical 2-D cross-section of a sonar sensor with a superimposed occupancy grid. The sonar model is divided into two areas: a cone-like freespace hypothesis, where the posterior probability of occupancy will be lowered, and an arc-like surface hypothesis, where it will be raised.

Given the geometry of the beam, a first approximation for the signal detection probability density would be gaussian in θ_i (target cross section decreases), and gaussian in the range deviation around r_i (range error), with the attenuation effects discussed previously. Given these considerations, one mathematical model for detection density in the multiple target case is:

$$p_{2m}(r = D \mid C_i) = \frac{\alpha(r_i)}{2\pi\delta(r_i)\sigma} e^{-\theta_i^2/2\sigma^2} e^{-(D-r_i)^2/2\delta(r_i)^2} + F \qquad (20)$$

where the target is at range r_i and angular deviation θ_i from the transducer. In this model, σ is a measure of the beam width; a typical value would be 12 degrees (for a beam width of 24 degrees). As before, $\alpha(r_i)$ is the attenuation of detection with distance, $\delta(r_i)$ is the range variance (increasing with distance), and F is a measure of detecting random other targets. With the exception of the angular term, this target density function is the same as in the previous section. All of the mathematics developed there carries over; for example, the value of $P_{2m}(r \not\leftarrow D \mid Q)$ is:

$$P_{2m}(r \not\leftarrow D \mid Q) = 1 - \int_0^D \int_{-\pi}^{\pi} p_{2m}(r = x \mid Q)\, d\theta dx, \quad \text{with } Q = C_i \text{ or } \overline{C_i} \qquad (21)$$

We can plot the log likelihood function for a 3-meter range reading (Fig. 7). The sensor reading is fixed at

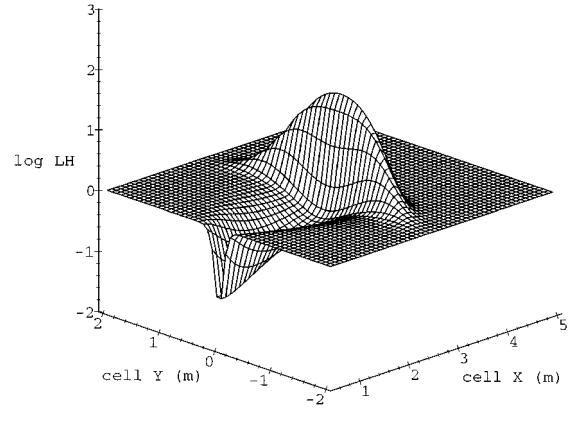


Figure 7. Multiple target model $\ln \lambda_{2m}(r = D \mid C_i)$ for a 3 meter range reading. The sensor is at 0, 0. X and Y axes are the coordinates of the cell to be updated; the Z axis shows the log likelihood value.

a distance D, and a value for $\ln \lambda$ at different cells C_i is computed. Thus, this plot shows how each cell in the grid will be updated given a diffuse sonar reading of the indicated range. The same features present in the 1-D plots of Fig. 4 are also present here.

4. Specular Models

In both the single and multiple target models, we have assumed that a range reading indicates the presence of a target at that range, somewhere within the surface hypothesis associated with the reading. This assumption is based on a *diffuse* reflection of energy from the target: if the surface roughness of the target is larger than the wavelength of the impinging beam, it will act like a point reflector, scattering energy equally in all directions.

When the surface of an object is smooth with respect to the wavelength of the beam, it will produce *specular* reflections, in which most of the energy is transmitted coherently. If the surface is angled obliquely to the beam axis, the energy will not be reflected directly back to the transducer, but will undergo multiple reflections before it is received (Leonard et al., 1990). In this case, the measured time-of-flight does not represent reflection from the nearest surface, and the target detection density $p_{2m}(r = D \mid C_i)$ given in Eq. (20) is incorrect (see Fig. 8).

Although it may be possible to extract information about the distance of surfaces, especially given some knowledge of their geometry, in most applications specular reflections are considered to yield no information about surfaces. In terms of the multiple target model, for specular reflections we would have:

$$p_{2s}(r = D \mid C_i) = p_{2s}(r = D \mid \overline{C_i}) = F. \qquad (22)$$

Given there are two distinct types of beam reflections, it makes sense to try to distinguish them. Unfortunately this is not possible by direct examination of the beam echo. Instead, we have to rely on indirect evidence about surfaces in the beam cone to decide whether a given reading is specular. Before examining this idea in more detail, we first develop the mathematics of model mixtures so that the information can be applied.

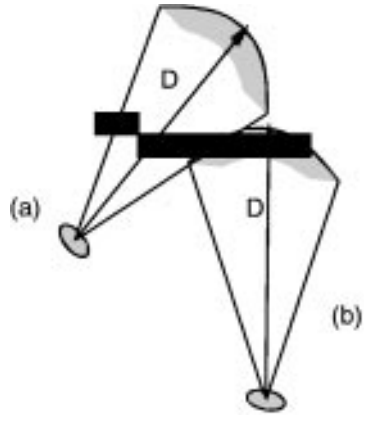


Figure 8. Specular (a) and diffuse (b) readings in a known local environment.

4.1. Model Classes and Mixtures

The natural division of range readings is between those that are from specular reflection, and those from diffuse reflection[4]. We'll call the class of specular reflections S, and diffuse reflections $\bar{S}$. Cases like Fig. 8(a) would be classified as specular, because a diffuse reflection from the surface would have generated a smaller range reading. The respective target density functions are given by Eqs. (22) and (20).

In general, we'll have only uncertain knowledge of whether a given reading is in S or not; so the diffuse and specular models must be combined with different weights, based on their probability. Since the diffuse and specular classes are mutually exclusive and exhaustive, we have $P(\bar{S}) = 1 - P(S)$. We are interested in calculating the quantity $p_c(r = D \mid C_i)$ as a function of the two model classes. By use of Bayes' rule we obtain:

$$\begin{aligned} p_c(r = D \mid C_i) &= p(r = D \mid C_i, \bar{S})P(\bar{S}) \\ &\quad + p(r = D \mid C_i, S)P(S) \qquad (23) \\ &= p_{2m}(r = D \mid C_i)(1 - P(S)) \\ &\quad + p_{2s}(r = D \mid C_i)P(S). \qquad (24) \end{aligned}$$

Equation (23) is just a probabilistic mixture of the two detection densities $p_{2m}(r = D \mid C_i)$ and $p_{2s}(r = D \mid C_i)$[5].

Figures 9 and 10 shows the results of an equal mixing and a 0.1 diffuse mixture. Since the target detection function p_{2s} adds no information, mixing it in quickly attenuates the effect of the diffuse model.

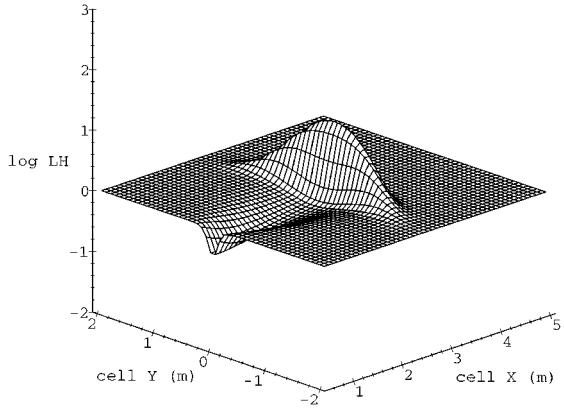


Figure 9. Model mixtures. This a log likelihood plot for an equiprobable mixture of specular and diffuse readings, with a range reading at 3 meters.

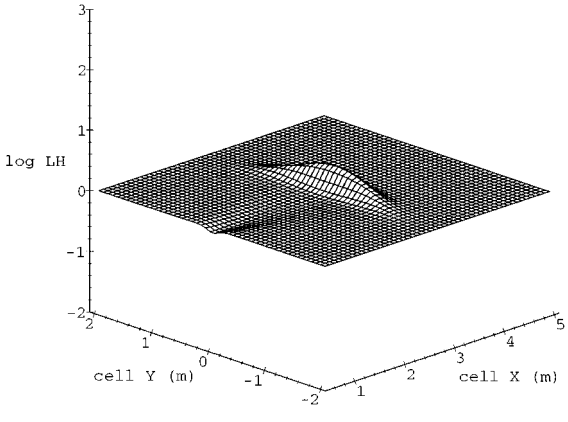


Figure 10. Model mixtures. This a log likelihood plot for a 10 percent diffuse/90 percent specular mixture, with a range reading at 3 meters.

4.2. Dynamic Model Mixtures

In the first experiments done with the occupancy grid model, it was assumed that the readings were all diffuse, mostly because the office environment consisted of many small and large articles that provided plenty of diffuse or corner reflectors for the sonars (Moravec and Elfes, 1985). Later, when the method was used in more specular environments such as office corridors, the value of $P(S)$ was determined theoretically (Elfes, 1990), or experimentally (Moravec and Blackwell, 1992) by tuning the sensor model for the overall environment. In the latter case, the diffuse and specular mix can vary for each value of the range reading, e.g., specular readings are much more likely when the range reading is high.

In all of these models, all readings are treated using the same target density function, with no attempt to take into account the local environment of the particular reading; hence we'll call them *averaging* models. Often there is information that helps to determine whether or not a particular sensor reading is specular. The most important is whether a reading is specular or not; Drumheller (1985) states a *sonar penetration condition*: the freespace hypothesis of a sonar reading should not impinge on a high-confidence surface. Knowledge of the environment, either *a priori* or acquired from previous sensor readings, can be used to estimate the probability of a given reading being specular, and thus change the model mixture *dynamically* to yield a more accurate interpretation of the measurement.

Figure 8 illustrates the difference between static and dynamic mixtures. There are two sonar readings of the same range D; one is specular, intersecting a surface at an oblique angle. The freespace and surface hypotheses are obviously wrong for the specular reflection. Since the mixture is fixed for a given value D, the model does not take into account the very different local environments of the two readings. Using the sonar penetration rule, the specular reading should be recognized and the mixture adjusted accordingly to weaken the hypotheses. In the MURIEL method, discussed below, we use local information to determine the probability $P(S)$ of specularity for each individual range reading, leading to a more refined estimate of the likelihood function.

5. Independent Evidence

In typical realtime applications, sensor readings are continuously acquired and processed to update the grid. In general, the posterior odds for occupancy of a cell are given by:

$$O(C_i \mid r_1 = D_1, r_2 = D_2, \ldots) = \lambda(r_1 = D_1, r_2 = D_2, \ldots \mid C_i) \times O(C_i). \quad (25)$$

Estimating the joint sensor model $\lambda(r_1 = D_1, \ldots \mid C_i)$ is difficult, so the assumption is made that the sensor readings are *conditionally independent of the cell's state*. Then the above equation has the form:

$$O(C_i \mid r_1 = D_1, r_2 = D_2, \ldots) = \lambda(r_1 = D_1 \mid C_i)\lambda(r_2 = D_2 \mid C_i) \cdots O(C_i). \quad (26)$$

This is a recursive equation: the prior odds are multiplied by the likelihood ratios, and stored as the updated prior. New readings just repeat the process.

Is conditional independence reasonable for the occupancy grid model? Let's look first at the simplest situation, the single target model. For conditional independence to hold, we have to have:

$$p_1(r = D, r' = D' \mid C_i) = p_1(r = D \mid C_i)\, p_1(r' = D' \mid C_i) \quad (27)$$

$$p_1(r = D, r' = D' \mid \overline{C_i}) = p_1(r = D \mid \overline{C_i})\, p_1(r' = D' \mid \overline{C_i}). \quad (28)$$

For the first term, the assumption of conditional independence is reasonable, because the gaussian range error is generated by small fluctuations in the amount of energy received or the receiver sensitivity, which are randomly distributed from one measurement to the next.

The second term, on the other hand, represents the target detection distribution when the target *isn't* at cell i. We've assumed that this is a uniformly random distribution. However, if the sensor and target are stationary, then two successive readings for the target will be highly correlated around the distance of the target, that is,

$$p_1(r = D, r' = D' \mid \overline{C_i}) \simeq p_1(r = D \mid \overline{C_i}). \quad (29)$$

So, taking two successive readings from the same position should give almost the same result as just one. If you think about it a little, this makes sense. Suppose, to the contrary, that every reading were conditionally independent. Then, at the end of a series of readings, the probability for a cluster of cells around the target distance r_i would be arbitrarily high. But we know that the target can be at only one cell, so conditional independence must be wrong.

There are several ways one might correct the conditional independence assumption. In the single target case, a *renormalization* of the resultant cell probabilities, so that they sum to one, would give a reasonable answer. This would make use of the fact that there is only a single target. In the multiple target case, this solution wouldn't work, because we know there can be more than one occupied cell. Instead, we might try to understand the correlation between occupancy of sets of cells, by keeping a probability for each configuration $C(i)$ of cells (see Eq. (19)). The target detection density for cell configurations is conditionally independent, so that the recursive updating formula would work. And individual cell posterior probabilities could be calculated by summing up the posterior probabilities $P(C(i) \mid r_1 = D_1, r_2 = D_2, \ldots)$. But except for very small evidence grids, this method is impractical because of the exponential number of cell configurations we would have to update.

Let's look at the problem from another angle, and ask under what circumstances we can treat $p_{2m}(r = D, r' = D' \mid \overline{C_i})$ as conditionally independent. The assumption is that the multiple targets are distributed to give a uniform set of readings, i.e., if we take enough sensor readings, we approach a uniform distribution. Since the environment is stationary, one way to try to make this the case is to move the sensor randomly, i.e., to take successive reading from different sensor *poses* (position and orientation). In fact, in most previous papers on evidence grids, it is an implicit assumption that each sensor reading is taken from a different pose. The usual strategy for collecting readings is to move a mobile robot to several different positions, and take a circular scan from each position. This method, in effect, is an approximation to random sampling based on sensor poses.

In more realistic situations, the robot will be continuously moving from one place to another, perhaps retracing its tracks. Even at a modest 20 readings/sec, the robot will accumulate thousands of readings, many of them redundant, over the period of a few minutes. Some mechanism is needed to filter just the independent readings. The simplest method is to keep a list of the pose of the sonar when it is fired, and check any new reading against the list. Although there do not appear to be any obstacles to implementing this scheme, the MURIEL algorithm uses a dual representation, in which each cell represents the pose of readings affecting the cell. There are several reasons for this choice.

1. The representation is local to each cell, so decisions about occupancy can be made on the basis of local information.
2. Pose information at a cell is useful for determining if a specular reading impinges on the cell.
3. In applications where readings can be coupled in complex ways, pose information can be used to estimate the coupling.

In the next section we develop algorithms for integrating multiple sensor readings at each cell.

6. The MURIEL Method

MURIEL stands for MUltiple Representation, Independent Evidence Log. It is an algorithm for determining cell occupancy using diffuse/specular mixed models and a log of all readings impinging on a cell.

The strategy of the algorithm is categorize the sonar readings at a cell into a set of discrete classes, based on (1) the distance of the sensor from the cell, (2) the angle of the sensor to the cell, and (3) whether the cell is in the freespace or surface hypothesis region of the sensor. If a cell contains a number of surface readings from different poses, it is likely that the freespace readings are from specular reflections, and the model mixture for these readings is adjusted accordingly. The occupancy of the cell is then computed using Eq. (25) for all readings on the cell.

There are some general assumptions used in the derivation of the method. First, the sensor should return information from a variety of poses, to get enough information for good reconstruction. Second, we assume that the environment is static; we haven't dealt with the problem of dynamic environments (as is typical for occupancy grid methods). In the concluding section we'll point out some modifications to the method that might deal with movement.

The rest of this section develops the algorithm in some detail, starting with the derivation of specularity probabilities $P(S)$, then defining the basic algorithm, and finally adding features to make the algorithm incremental, so that it can be computed in realtime as the robot moves.

6.1. *Computing the Probability of Specularity*

For any given sensor event, the probability that it will be a specular reading (in the sense defined in Section 4.1) is a function of the geometry of surfaces around the sensor. *A posteriori*, we can conclude that a reading is specular if we know that there are surfaces of a particular kind within the freespace hypothesis of the sensor reading. These surfaces must be smooth and slanted enough from perpendicular to the beam axis. Note that small objects in the freespace of the beam don't necessarily mean a specular reading: they may just have too small a cross-section to be detected.

In practice, we often start out with little information about the environment in the occupancy grid, and build up a picture of the surfaces as more sensor readings are acquired. Ideally, once enough surface information is accumulated, we should go back to earlier readings, determine their specularity, and update the results. But this would require a lot of recomputation, and keeping track of what readings gave rise to cell occupancy values would be a hard bookkeeping task. So, instead, MURIEL uses a simplified method of computing $P(S)$ that is local to a given cell. The idea is the following: if a cell is occupied, then from some poses a sensor will give a surface hypothesis reading for the cell. If a cell has enough such readings, we assume that it is (probably) occupied, and compute the specularity probability $P(S)$ for all *freespace* readings at the cell, based on a measure of how strong the surface hypotheses are. We have found that the following estimation gives good results.

First, sum up all of the $\log \lambda$ surface readings at the cell; call this value $\log \lambda_S$. Then, establish a cutoff value C_S for surface determination. If the value of $\log \lambda_S$ is greater than C_S, then $P(S)$ is 1, that is, any freespace readings will be recognized as specular. If $\log \lambda_S$ is 0, then $P(S)$ is 1. For values of $\log \lambda_S$ between these endpoints, $P(S)$ is computed by linear interpolation. The cutoff C_S is a parameter of the algorithm. Higher values mean that the specular component is not recognized as quickly; reasonable values are 2–3.

This method has the advantage of being quickly computable, and yielding a reasonable value for $P(S)$ in many situations. But, it has the following drawbacks. First, it treats all freespace readings as having the same $P(S)$ value, based on surface readings at one cell only. Obviously, there may be some freespace readings that are much more likely to be specular, because their freespace readings impinge on other occupied cells.

Second, different cells may conclude that a single reading is specular or diffuse. For example, if a cell has no surface readings, any freespace readings are assumed to be diffuse; where one of these readings impinges on a cell with substantial surface hypotheses, that cell will determine it to be specular. As a result, some freespace areas may be given overly generous updates.

6.2. *Basic Algorithm*

The MURIEL algorithm proceeds by updating cells affected by each new reading. Given an occupancy grid with pose buckets, and a new sensor reading divided into freespace and surface hypotheses, MURIEL proceeds with the following five steps for each cell in the active area of a new sensor reading.

1. Data collection. At each cell, the new reading is checked to see if it duplicates any previous readings. For this purpose, a set of discrete *pose buckets* are kept at each cell, one set for freespace readings, one set for surface readings. Each pose bucket represents a range of angles and distances to the sensor. If the reading is duplicated, it is discarded; else the appropriate pose bucket is marked as filled. This step eliminates non-independent sensor readings.
2. $\log \lambda_S$ calculation. The log likelihood ratio for all *surface hypothesis* readings is computed using Eq. (20) for the multiple target model.
3. $P(S)$ is computed for the cell, according to the prescription given above.
4. Occupancy computation. Given the value for $P(S)$, the value of $\log \lambda_F$, the freespace log likelihood ratio, is computed by using Eq. (24) for each freespace reading.
5. The final odds of cell occupancy is computed as $\log \lambda_S + \log \lambda_F + \log O(C_i)$.

The algorithm can be implemented using a data structure at each cell representing the pose buckets as bit vectors. The current implementation uses 64 angular divisions (5.625 degrees) and three displacements: less than 1 m, 1 m to 2 m, and greater than 2 m. Thus each cell needs a total of 48 bytes to represent pose information. The update likelihoods for freespace and surface hypotheses can be precomputed for each pose, and saved in a table. From this information, the cell log odds can be reconstructed using the MURIEL algorithm, as described above.

6.3. *Incremental Update*

For each new reading impinging on a cell, the basic algorithm recomputes the log likelihood ratio for each pose, then sums them up. For greater efficiency, the algorithm can be modified slightly so that λ_S and λ_F are computed incrementally. The values of λ_S and λ_F are stored at a cell; when a new surface reading comes in, λ_S is updated by multiplying it with the likelihood ratio for the new reading. When a new freespace reading arrives, λ_F is updated in the same manner, by multiplying it with the likelihood ratio for the new reading, as if it were non-specular. Finally, to compute the total log likelihood, we use a linear interpolation:

$$\log \lambda_T = \log \lambda_S + \log(\lambda_F(1 - P(S)) + P(S)). \quad (30)$$

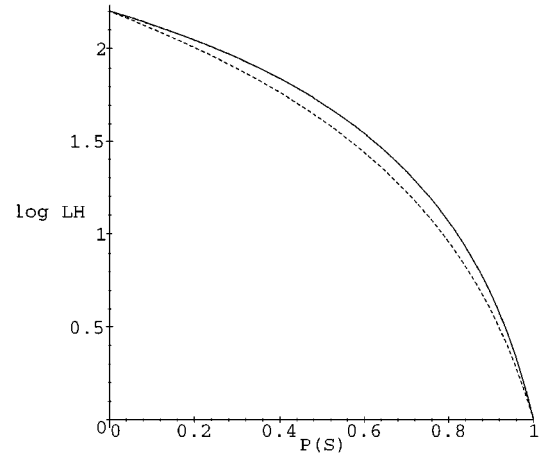


Figure 11. Approximation to the model mixture. The solid line shows the true $\log \lambda_F$ value for various values of $x = P(S)$, while the dashed line is the approximation of Eq. (30).

Figure 11 shows the difference between the true mixture and the approximation of Eq. (30). Using the approximation lets us avoid having to recalculate λ_F using all freespace readings at a cell. The steps for a new reading are now:

1. Check that it is not redundant (table lookup)
2. Update likelihood ratio (table lookup and multiplication)
3. Calculate $P(S)$ (linear interpolation)
4. Calculate log likelihood (Eq. (30))

The algorithm has been implemented in C using a discrete-beam approach to approximate the cone-shaped sonar reading: a small set of linear beams covering the cone are traversed from the origin of the sonar outwards. The cell size is 10 cm on a side, which is a reasonable tradeoff between resolution and algorithm efficiency. A typical sonar reading will update some tens of cells, and the algorithm takes less than a millisecond to do this on a Sparcstation 10–51. Given a sensor rate of 20 Hz, MURIEL takes a small fraction of realtime computational resources.

7. Results

The algorithm was tested on SRI's small mobile robot, Erratic (Konolige, 1995), in the corridors of SRI. Erratic has 7 sonars distributed in a 180 degree arc along its front. As the robot moves along the hallway, there is significant specular reflection from the diagonal and

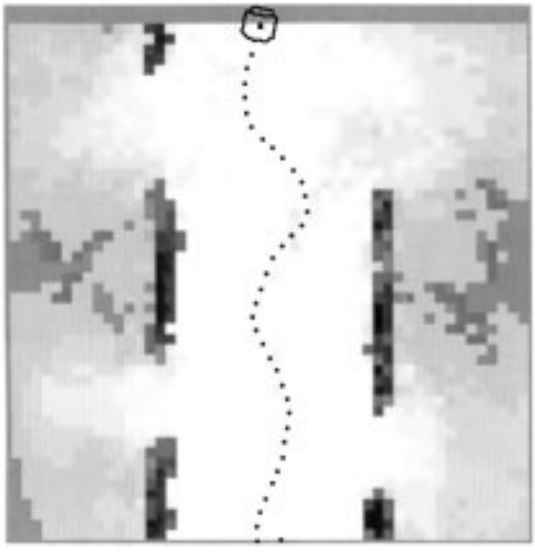


Figure 12. Results of the MURIEL algorithm on a typical hall scene. Grayscale indicates occupancy, with white ($\ln \lambda \leq -2$) being unoccupied and black ($\ln \lambda \geq 2$) being occupied. For calibration, the darker gray areas on the center far left and far right are at the unknown point (posterior odds = 1). The open wall segments are doorways, and there is an open junction in the upper right. Essentially all wall segments are correctly identified.

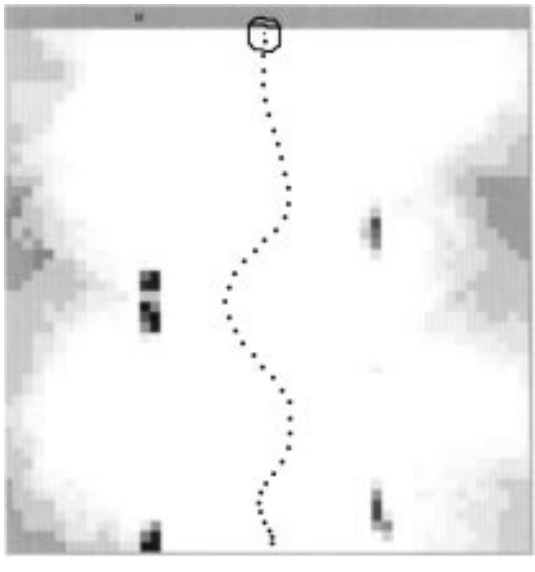


Figure 13. Results of the standard algorithm on a typical hall scene. Note how few wall segments are found, due to interference by specular reflection and the overwhelming influence of redundant readings.

side sonars. The results are shown in Figs. 12–16, comparing MURIEL with the fixed-mixture ("standard") model.

The robot was run at a slow speed (approximately 100 mm/sec) for 4 meters. The sonar rate was 20 Hz,

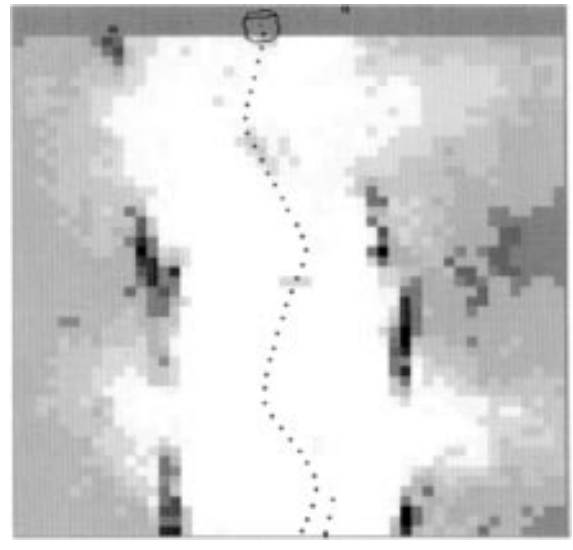


Figure 14. Results of the standard algorithm with pose buckets. All redundant readings are discarded. There is some destructive interference from specular reflections, leading to missed wall segments. Also, some free areas are taken to be occupied, at the surface hypothesis of specular readings.

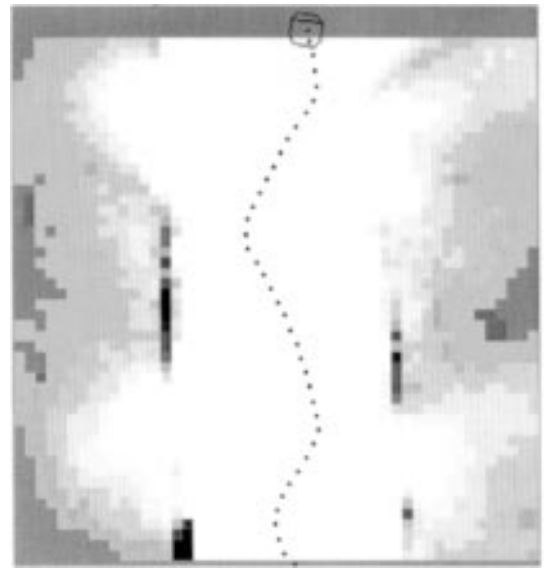


Figure 15. Results of the standard algorithm using a .3 mixture of specular reflections.

for a total of approximately 800 readings. Some attempt was made to turn the robot so that the sonars would impinge on the walls at a variety of angles. The path the robot took is indicated by the dots, which are the center of the robot at 1-second intervals.

No attempt was made to tune the diffuse model (Eq. (14)) to the hallway; the parameter values were

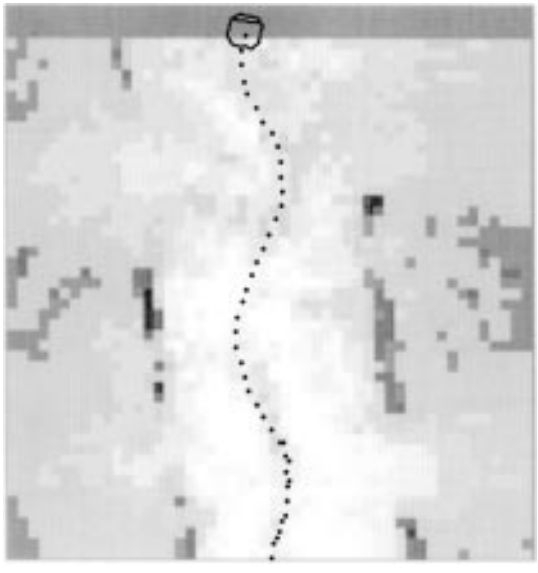


Figure 16. Results of the standard algorithm with pose buckets and .3 specular mixture.

taken from an old set of experiments with sonar sensors and a diffuse-reflecting object. The value of the parameter C_S was chosen to be 1.5, which means that a cell whose surface readings sum to more than 1.5 will reject freespace readings. This setting is a strong one, and biases the algorithm towards trusting surface hypotheses that are confirmed from a small set of poses.

As can be seen in Fig. 12, MURIEL extracts wall segments that are somewhat thick, because of the range uncertainty in the surface hypothesis, and because of the discrete cell size. These wall segments match the extents of the real hallway walls almost exactly, with the open doors and a large junction in the upper right indicated as freespace. The freespace model tends to be conservative, requiring 3 or 4 different poses at short distances before committing to full non-occupancy. Note also that there is some "bleeding" of freespace behind the walls, from specular reflections. This could be attenuated a bit by using a more global specular check, in which a reading was marked as specular for all cells if it was specular for an individual cell.

For comparison, similar readings were used to reconstruct the hall scene using the "standard" algorithm, with the diffuse reflection model (i.e., the specular contribution was assumed to be zero), in Fig. 13. Here the wall segments have been almost obliterated because of the overwhelming influence of multiple specular readings. You could say the robot is blinded by redundant specular reflections; the occupancy grid has the same saturation characteristics as a CCD sensor in too-bright light. This hallway is particularly susceptible to specular readings, since the walls are smooth wallboard.

The problem of specular reflections can be mitigated somewhat by throwing away redundant readings, as in Fig. 14. Here the standard model was used to compute occupancy, but only independent readings, as determined from the pose information, were kept. There is not nearly as much freespace saturation from specular reflections, and the hall boundaries are seen more clearly. Still, since specular reflections are weighted equally with diffuse reflections, the wall boundaries still have washed out relative to the MURIEL results in Fig. 12.

It might be remarked that because of the presence of a large number of specular readings, the natural model to use for the fixed-mixture algorithm would have a strong specular component. Figures 15 and 16 show the standard algorithm using a 30% specular mixture. The first of these uses all redundant readings. As in the diffuse case, there is a "washed-out" appearance to the grid, although it is less severe because the freespace hypothesis has been attenuated. Still, a lot of the wall segments are missing, because the surface hypotheses are also weaker.

One could argue that better results would be obtained by increasing the surface hypothesis strength relative to the freespace hypothesis. This strategy produces a different problem: the surface hypotheses of specular reflections start to become apparent, in arcs behind the walls. The point is, there is *no* model mixture that will do well in the presence of specular reflections, since these are counted the same as diffuse reflections, but their hypotheses are not correct. The most that can be done is to try to minimize their effect by discounting long-range readings, which tend to be more specular. This is apparent in the hallway scene reproduced in Fig. 2 of (Moravec and Blackwell, 1992), where the best possible fixed sonar model produces both freespace bleeding and false surface patches beyond the walls.

For completeness, Fig. 16 shows the 30% mixture with redundant readings removed. A lot of the wall area is weakly indicated, similar to Fig. 14. But the weakness of the model mixture is apparent, with very few clear freespace or surface patches.

Readers who wish to see the results of the algorithm running in a simulation system can download

the SAPHIRA software from the link `http://www.ai.sri.com/~konolige/saphira`. Follow the instructions for installing the SAPHIRA system, then start the simulator and the sample saphira client. The occupancy grid routines are invoked using the `Display/Occ Grid` menu item.

8. Conclusion

This paper presents a method, called MURIEL, for dealing with the problem of specular reflection and redundant readings in sensor fusion. The main insight is that it is possible to discount redundant and specular readings in a local fashion by keeping track of the readings that impinge on a given cell. Although the results presented here are qualitative, they provide anecdotal evidence that MURIEL can significantly improve the fidelity of the occupancy grid computation.

Although the examples in this paper are based on sonar sensors, the MURIEL method can improve the performance of occupancy grids with any sensor that has a specular component, such as radar. It can also help in multimodal fusion, in which information from different types of sensors is integrated. The sensors complement each others' strengths, giving results not possible with a single sensor (Matthies and Elfes, 1988). In multimodal fusion, MURIEL could eliminate double-counting from single-sensor readings taken from the same pose. It could also help filter specular reflections where appropriate, by using information from all sensors to recognize cells with strong occupancy odds, and then applying the local specular criterion.

Some modifications of the MURIEL method, which we have not investigated, might prove useful. One is a global assessment of specularity, based on the sonar penetration condition. Any new reading could be checked in this way; but it would be hard to keep track of older readings and re-evaluate them whenever a relevant cell is modified.

By using just one reading per pose bucket, a lot of readings are discarded. Although much of this information is redundant, in some cases it may help to categorize a cell state. For example, given very small objects, or ones that are far away, the frequency of the sensor response at a given pose is a measure of confidence in the presence of the object from that pose. It would be relatively easy to incorporate such refinements into the MURIEL algorithm, although given the overwhelming contribution of geometric error, this refinement will probably not contribute a great deal to the final result. One exception, however, would be to use multiple readings and temporal information to "decay" information in the grid, similar to ideas proposed for the VFH method (see Section 2).

Acknowledgments

The reviewers of a preliminary version of this paper provided many helpful comments, leading to substantial clarifications of some sections.

Notes

1. Probability densities are written with lowercase p, and probabilities with uppercase P.
2. Many sonar and radar sensors can detect more than one echo from a single output pulse. More complicated processing schemes can make use of these multiple echoes, but typical mobile robot applications will look at just the first one.
3. The generalization to three dimensions is straightforward, but would complicate this exposition. Also, indoor mobile robots are generally happy with two-dimensional representations, since they typically operate on a fixed, level ground plane. At most they might need several such plane representations for obstacles or depressions at different heights.
4. Leonard and Durrant-Whyte (1992) point out that most strong readings from sonars are actually specular, e.g., from corner reflectors or from surfaces normal to the sonar beam. These returns aren't "specular" in the sense used here: rather, we are using "specular" to refer to any sensor reading that has reflected off multiple nonlocal surfaces before returning to the sensor.
5. The specular/diffuse distinction makes little difference to $p(r = D \mid \overline{C_i})$. Specular reflections might have reduced probabilities for detecting random targets at short ranges, but we won't deal with this complication.

References

Borenstein, J. and Koren, Y. 1991. The vector field histogram—fast obstacle avoidance for mobile robotics. *IEEE Journal of Robotics and Automation*, 7(3):278–288.

Cox, I.J. and Leonard, J.J. 1994. Modeling a dynamic envrionment using a bayesian multiple hypothesis approach. *Artificial Intelligence*, 66(2):311–344.

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

Crowley, J.L. 1989. World modeling and position estimation for a mobile robot using ultra-sonic ranging. In *Proceedings IEEE International Conference on Robotics and Automation*, Scottsdale, Arizona, pp. 674–681.

Drumheller, Michael. 1985. Mobile robot localization using sonar. A.I. Memo 826, Massachusetts Institute of Technology.

Elfes, A. 1990. Occupancy grids: A stochastic spatial representation for active robot perception. In *Sixth Conference on Uncertainty in AI*.

Elfes, A. 1992a. Dynamic control of robot perception using multi-property inference grids. In *International Conference on Robotics and Automation*.

Elfes, Alberto. 1992b. *Multi-source Spatial Data Fusion Using Bayesian Reasoning*, Academic Press, pp. 137–163.

Konolige, Kurt. 1995. Erratic competes with the big boys. *AI Magazine*, Summer.

Leonard, J., Durrant-Whyte, H., and Cox, I.J. 1990. Dynamic map building for an autonomous mobile robot. In *IROS*, pp. 89–95.

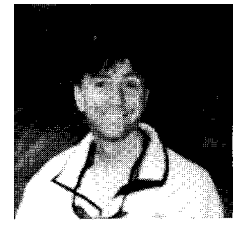
Leonard, J. and Durrant-Whyte, H. 1992. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers: Boston.

Matthies, L. and Elfes, A. 1988. Integration of sonar and stereo range data using a grid-based representation. In *International Conference on Robotics and Automation*.

Moravec, H.P. and Elfes, A.E. 1985. High resolution maps from wide angle sonar. In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, Washington, D.C., pp. 116–121.

Moravec, H.P. and Blackwell, M. 1992. Learning sensor models for evidence grids. In *Robotics Institute Research Review*, Pittsburgh, PA.

Polaroid Ultrasonics Group. 1992. Ultrasonic ranging system manual. Technical report, Poloroid Corporation, Atlanta, GA.



Kurt Konolige is a Senior Computer Scientist at the Artificial Intelligence Center of SRI International, and a Consulting Professor at Stanford University. He received his Ph.D. in Computer Science from Stanford University in 1984; his thesis, "A Deduction Model of Belief and its Logics," develops a model of belief based on the resource-bounded inferential capabilities of agents. His research interests are broadly based on issues of commonsense reasoning, including introspective reasoning, defeasible reasoning, and reasoning about cognitive state, especially in the context of multiagent systems. More recently, he has conducted research in fuzzy control for reactive systems; in constraint-based planning and inference systems; in reasoning about perceptual information; and in realtime robotics and vision systems.