Fuzzy Maps: A New Tool for Mobile Robot Perception and Planning

Giuseppe Oriolo[*], Giovanni Ulivi[†], and Marilena Vendittelli[*]

Abstract

An essential component of an autonomous mobile robot is the heteroceptive sensory system. Sensing capabilities should be integrated with a method for extracting a representation of the environment from uncertain sensor data and with an appropriate planning algorithm. In this paper, fuzzy logic concepts are used to introduce a tool useful for robot perception as well as for planning collision-free motions. In particular, a map of the environment is defined as the fuzzy set of unsafe points, whose membership function quantifies the possibility for each point to belong to an obstacle. The computation of this set is based on a specific sensor model and makes use of intermediate sets generated from range measures and aggregated by means of fuzzy set operators. This general approach is applied to a robot with ultrasonic rangefinders. The resulting map building algorithm performs well, as confirmed by a comparison with stochastic methods. The planning problem on fuzzy maps can be solved by defining various path cost functions, corresponding to different strategies, and by searching the map for optimal paths. To this end, proper instances of the $A^\star$ algorithm are devised. Experimental results for a Nomad 200^{TM} robot moving in a real-world environment are presented.

[*]Dipartimento di Informatica e Sistemistica, Università degli Studi di Roma "La Sapienza", Via Eudossiana 18, 00184 Rome, Italy

[†]Dipartimento di Meccanica e Automatica, Terza Università degli Studi di Roma, Via Ostiense 157, 00154 Rome, Italy

1 Introduction

In the last few years, the research activity on mobile robots has gained considerable impetus, as witnessed by a constantly growing literature, e.g., see [1, 2, 3]. The central topic of this investigation is *autonomy*, i.e., the capability of planning and executing motion tasks without human guidance. Such a faculty is mandatory for advanced robotic systems acting in environments where human operation is difficult or dangerous, such as space, sea depths, and contaminated habitats. A certain degree of autonomy will also be required in the forthcoming field of *service robotics* applications, including waste management, cleaning of public areas, luggage transfer in airports and train stations, disabled people assistance, and many others.

Autonomy is realized through the integration of *sensing* and *intelligence*. While on-board sensory systems allow the gathering of information about the environment in the absence of exogenous knowledge, machine intelligence is essential both for building a convenient representation of the environment from sensor data, and for planning appropriate robot actions. While sophisticated sensors and high computing power are available on the market, to make the use of robots appealing in real-life applications it is necessary to reach a trade-off between costs and benefits. Often, this prevents the use of expensive sensors (e.g., video cameras) in favor of cheaper sensing devices. Smarter algorithms are then needed to extract significant information from data that may be insufficient or conflicting.

The purpose of this paper is to show that fuzzy set theory provides a natural framework in which perception and planning problems can be formulated and solved. Rather than trying to reconstruct a deterministic model of the environment, we have chosen to adopt an intrinsically uncertain map, defined as a *fuzzy set*: a real number is associated to each point, quantifying the possibility that it belongs to an obstacle. The resulting representation is similar to an *occupancy grid*, commonly obtained using stochastic techniques [4, 5, 6]. In our experience, fuzzy logic provides a more robust and efficient tool for managing the uncertainty introduced by the ultrasonic sensing process. In fact, the underlying theory is developed from less conservative axioms than probability theory, so that a wider choice of operators is available for modeling uncertainty and aggregating information coming from multiple sources [7].

We shall apply the above approach in order to devise a method for building a map of an unknown environment, starting from a series of measures obtained with ultrasonic rangefinders. These sensors offer a satisfactory trade-off between cost and accuracy, together with a remarkable simplicity of use. However, due to their poor angular resolution and to the phenomenon of multiple reflections, the obtained measures can be affected by a large amount of uncertainty [8].

To show that the proposed representation provides an effective tool for generating collision-free robot motions, we address the problem of planning safe paths on fuzzy maps. As the map is uncertain, it is not possible to utilize algorithms relying on a sharp distinction between the free space and the obstacles. Our solution is to characterize the risk of collision along a path by proper cost functions, that correspond to different planning attitudes. The $A^\star$ class of graph search algorithms, which perform a local search driven by a suitable heuristic, has been chosen to build minimum-cost paths, because of its flexibility and efficiency. For each cost function, it is necessary to identify a proper instance of $A^\star$.

One constraint we committed to at the beginning of this research was that the developed tools should be useful for *sensor-based navigation*. The problem of autonomous motion in unknown environments is very difficult, and calls for sophisticated sensing strategies [2, Ch. 5] and sophisticated control architectures [9]. Several approaches have been investigated, ranging from *reactive* to *deliberative* behaviors. In the former, there is a stimulus-response relationship between sensors and actuators, with as little as possible world modeling [10]; for example, see [9, 11, 12, 13]. In deliberative approaches, a world model is used to plan actions, to which the robot is then more or less committed; examples of this kind are given in [14, 15].

The fuzzy map concept is well suited for use in deliberative or mixed approaches, which are still preferable for finding good solutions to complex tasks. Both the map building and the planning algorithm to be presented here lend themselves very naturally to be used on-line. In fact, the map building method is memoryless and requires no post-processing phase (e.g., cell clustering, feature extraction). As a consequence, the same final map is obtained regardless of the order in which the range readings are processed. As for the planning algorithm, the adoption of $A^\star$, which is a global planning method but takes full advantage of all the available local information, is particularly convenient. To support this claim, we have presented in [16] a method for on-line map building and navigation based on the tools introduced here.

The paper is organized as follows. We begin with a short review of the basic concepts of fuzzy set theory. In Sect. 3, the various phases of the fuzzy map building algorithm from ultrasonic measures are reviewed, and experimental results for a Nomad 200$^{\text{TM}}$ robot are presented. The performance of the proposed method is compared to that of stochastic techniques in Sect. 4, by means of a simple example. In Sect. 5, three cost functions are introduced for characterizing safe paths on fuzzy maps. In Sect. 6, we apply the $A^\star$ algorithm for finding minimum-cost paths and report some simulation results. The concluding section offers some comments about potential applications of the approach that are currently under investigation.

2 A survey of fuzzy set theory

Studies on the theory of fuzzy sets started in the early 70's, with the seminal papers of Zadeh (e.g., see [17]). Here, only the basic concepts will be reviewed. The interested reader may refer, for example, to [7, 18, 19].

Fuzzy sets may be easily introduced as extensions of standard crisp sets. For a *crisp set* A defined over the universal set U, the *membership function*

$$\mu_A : U \mapsto \{0,1\}, \quad \mu_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

identifies those elements of U that belong to A.

For a *fuzzy set* $\mathcal{A}$ defined over U, the membership function

$$\mu_{\mathcal{A}} : U \mapsto [0,1]$$

may assume any real value within the interval $[0,1]$, expressing the degree of membership of any element of U to $\mathcal{A}$. Hereafter, calligraphic capital letters denote fuzzy sets.

Basic fuzzy set operators (i.e, complementation, intersection, and union) are defined as generalizations of the classical crisp set operators. In particular, they must satisfy proper sets of axioms [18], that, however, do not uniquely define the operators. As a consequence, several options for the same operation are available. This contributes to the richness and flexibility of fuzzy logic, but on the other hand the selection of the most suitable operators requires special care.

Any monotonic function

$$c : [0,1] \mapsto [0,1]$$

that satisfies the boundary conditions $c(0) = 1$ and $c(1) = 0$ may be used to define a *complementation* operator. In particular, the membership function of the complement $\bar{\mathcal{A}}$ of the fuzzy set $\mathcal{A}$ is denoted by

$$\mu_{\bar{\mathcal{A}}}(x) = c(\mu_{\mathcal{A}}(x)).$$

In general, c is also required to be continuous. The most common complementation operator is obtained by letting

$$c(\mu_{\mathcal{A}}(x)) = 1 - \mu_{\mathcal{A}}(x). \tag{1}$$

Similarly, set *intersection* operators are defined through functions

$$i : [0,1] \times [0,1] \mapsto [0,1]$$

that are commutative, associative, and monotonic. Furthermore, i must satisfy the four boundary conditions $i(0,0) = i(1,0) = i(0,1) = 0$ and $i(1,1) = 1$. It is often required that i is continuous and, sometimes, that it is *idempotent*, i.e., $i(a,a) = a$. Standard intersection operators are

$$i_1(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cap\mathcal{B}}(x) = \min(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)),$$

the *algebraic product*

$$i_2(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cap\mathcal{B}}(x) = \mu_{\mathcal{A}}(x) \cdot \mu_{\mathcal{B}}(x),$$

and the *bounded product*

$$i_3(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cap\mathcal{B}}(x) = \max(0, \mu_{\mathcal{A}}(x) + \mu_{\mathcal{B}}(x) - 1). \tag{2}$$

Finally, set *union* operators are obtained from functions

$$u : [0,1] \times [0,1] \mapsto [0,1]$$

that are commutative, associative, monotonic, and satisfy the boundary conditions $u(0,0) = 0$ and $u(1,0) = u(0,1) = u(1,1) = 1$. Again, one may require u to be continuous and idempotent. Typical choices include

$$u_1(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cup\mathcal{B}}(x) = \max(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x))$$

the *algebraic sum*

$$u_2(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cup\mathcal{B}}(x) = \mu_{\mathcal{A}}(x) + \mu_{\mathcal{B}}(x) - \mu_{\mathcal{A}}(x) \cdot \mu_{\mathcal{B}}(x),$$

and the *bounded sum*

$$u_3(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \mu_{\mathcal{A}\cup\mathcal{B}}(x) = \min(1, \mu_{\mathcal{A}}(x) + \mu_{\mathcal{B}}(x)). \tag{3}$$

It is interesting to note that the triples (c, i_1, u_1), (c, i_2, u_2), and (c, i_3, u_3) satisfy De Morgan's law.

A whole class of union operators, which we shall use for aggregating information, has been introduced by Dombi [20]:

$$u_\lambda(\mu_{\mathcal{A}}(x), \mu_{\mathcal{B}}(x)) = \frac{1}{1 + \left[\left(\frac{1}{\mu_{\mathcal{A}}(x)} - 1\right)^{-\lambda} + \left(\frac{1}{\mu_{\mathcal{B}}(x)} - 1\right)^{-\lambda}\right]^{-\frac{1}{\lambda}}}, \tag{4}$$

with $\lambda \in (0, \infty)$. One has

$$\lambda_1 < \lambda_2 \Longrightarrow u_{\lambda_1}(a,b) > u_{\lambda_2}(a,b).$$

Defining [18] the *cardinality* of a fuzzy set $\mathcal{A}$ over U as $|\mathcal{A}| = \sum_{x\in U} \mu_{\mathcal{A}}(x)$, the above inequality indicates that the Dombi operator produces 'larger' union sets as λ is decreased—one may say that 'weaker' unions are obtained for smaller values of λ. This behavior is illustrated in Fig. 1, where we show the result of the iterated application of u_λ to the same number $a = 0.25$, for different values of λ. Note that the Dombi operator is not idempotent.

3 Building fuzzy maps from ultrasonic measures

Fuzzy logic finds its natural application in handling two different kinds of uncertainty, namely *(i) vagueness*, associated with the difficulty in characterizing a particular concept or property with a crisp set, or *(ii) lack of evidence*, that does not indicate whether a given element is a member of a particular crisp set. Map building from ultrasonic readings provides an example of the second kind of uncertainty. In principle, map building could be formulated as a decision problem; i.e., determining for each point of the area of interest (the universal set U) whether it belongs to the *empty* space or to the *occupied* space, two crisp sets giving a partition of U. However, due to the large amount of uncertainty introduced by the ultrasonic sensing process, enforcing such a sharp dichotomy may not be convenient. Depending on the decision method, the resulting map may be too conservative or, on the other hand, overconfident.

Here, the empty and the occupied space are defined as fuzzy sets over the universal set U, that is assumed to be a two-dimensional subset of $\mathbb{R}^2$. Denoting these sets by $\mathcal{E}$ and $\mathcal{O}$, their membership functions are $\mu_{\mathcal{E}}$ and $\mu_{\mathcal{O}}$, respectively. Thus, the task of the map building module will include the computation, for each point $x \in U$, of the two values $\mu_{\mathcal{E}}(x)$ and $\mu_{\mathcal{O}}(x)$. Note that, in the fuzzy context, the two sets $\mathcal{E}$ and $\mathcal{O}$ are no longer complementary: for a given point, partial membership to *both* $\mathcal{E}$ and $\mathcal{O}$ is possible. In other words, the principle of *tertium non datur* does not hold.

Hereafter, the procedure for building a fuzzy map from ultrasonic measures is presented. We will assume that a stream of range readings r_i, obtained at known sensor locations, is available as input. It is intuitive that the obtained map will be more accurate if the measurement locations (namely, the locations where the robot collects a series of range readings) are numerous and well distributed.

The objective of the map building phase is to produce a fuzzy set $\mathcal{M}$, called the *fuzzy map*, that conveys information about the risk of collision for each point of the environment. The fundamental steps of the algorithm are shown in Fig. 2. In the discussion below, each step is described in detail.

3.1 Range measures processing

Ultrasonic rangefinders work according to a simple principle: a packet of ultrasonic waves is generated and the resulting echo is detected. The time elapsed between transmission and reception is assumed to be proportional to the distance of the sensed obstacle. Hereafter, we will refer to the Polaroid Ultrasonic Rangefinder, a very common device that is described in detail in [21]. The mobile robot Nomad 200$^{\text{TM}}$, used to perform experiments in our Laboratory, is equipped with a ring of 16 such sensors.

The multi-lobed beam pattern of the transmitter may be obtained from the radiation directivity function of a plane circular piston

$$D(\vartheta) = 2\frac{J_1(kp\sin\vartheta)}{kp\sin\vartheta}, \tag{5}$$

where $J_1(\cdot)$ is the Bessel function of the first order, $k = 2\pi/\ell$ depends on the wavelength ℓ, p is the piston radius and ϑ is the azimuthal angle measured with respect to the beam central axis. For the Polaroid sensor it is $p = 0.01921$ m and $\ell = c/f$, where c is the sound speed in air and $f = 49.410$ kHz. For practical purposes it is sufficient to take into account only the principal lobe of the pattern. As a consequence, the waves are considered to be diffused over a radiation cone of 25° width.

There are basically three sources of uncertainty when using an ultrasonic rangefinder for determining the position of an object inside the radiation cone. First, the measured distance r is affected by an error. The standard Polaroid Rangefinder can detect distances from 0.12 to 6.5 m with 1% accuracy over the entire range. Second, it is impossible to determine the angular position of the object that originated the echo; for example, all the three obstacles of Fig. 3 give the same distance reading. Finally, *multiple reflections* may occur: the sonar beam is scattered only by surfaces whose irregularities are comparable in size to the wavelength of the emitted pulse, that is about 0.0068 m for the Polaroid sensor. Therefore, it may bounce off when the incidence angle is larger than a critical value α (see Fig. 3); in this case the sensor reading is not significant because the beam typically reaches the receiver after being reflected by other surfaces, or may even get lost. The angle α depends on the surface characteristics, ranging from 7–8° for smooth glass to almost 90° for very rough materials.

The above qualitative description can be directly used to set up a certainty model for the state of points belonging to the radiation cone. Unless a multiple reflection occurred, a single reading r provides the information that one or more obstacles are located somewhere along the 25° arc of circumference of radius r. Hence, there is evidence that points located in the proximity of this arc are 'occupied'. On the other hand, points well inside the circular sector of radius r are likely to be 'empty'. To model this knowledge, we introduce the two functions

$$f_{\mathcal{E}}(\rho, r) = \begin{cases} k_{\mathcal{E}} & 0 \leq \rho < r - \Delta r \\ k_{\mathcal{E}} \left(\frac{r-\rho}{\Delta r}\right)^2 & r - \Delta r \leq \rho < r \\ 0 & \rho \geq r \end{cases} \tag{6}$$

$$f_{\mathcal{O}}(\rho, r) = \begin{cases} 0 & 0 \leq \rho < r - \Delta r \\ k_{\mathcal{O}} \left[1 - \left(\frac{r-\rho}{\Delta r}\right)^2\right] & r - \Delta r \leq \rho < r + \Delta r \\ 0 & \rho \geq r + \Delta r, \end{cases} \tag{7}$$

that describe, respectively, how the degree of certainty of the assertions 'empty' and 'occupied' vary with ρ for a given range reading r. Here, ρ is the distance from the sensor, $k_{\mathcal{E}}$ and $k_{\mathcal{O}}$ are two constants corresponding to the maximum values attained by the functions, and $2 \cdot \Delta r$ is the width of the area considered 'proximal' to the arc of radius r. The profile of $f_{\mathcal{E}}$ and $f_{\mathcal{O}}$ is displayed in Fig. 4. The choice of $k_{\mathcal{E}}$, $k_{\mathcal{O}}$, and Δr, as well as other parameters to be introduced, will be detailed in Sect. 3.4.

Since the intensity of the waves decreases to zero at the borders of the radiation cone, the degree of certainty of each assertion is assumed to be higher for points close to the beam axis. This is realized by defining an *angular modulation* function

$$m_1(\vartheta) = \begin{cases} D(\vartheta) & 0 \leq |\vartheta| \leq 12.5° \\ 0 & |\vartheta| > 12.5°, \end{cases} \tag{8}$$

where $D(\vartheta)$ is the radiation directivity function (5). For ease of computation, the latter can be accurately approximated inside the principal lobe by a 4-th order polynomial $\widetilde{m}_1(\vartheta)$, shown in Fig. 5.

Finally, a *radial modulation* function was defined as

$$m_2(\rho) = 1 - \frac{1 + \tanh(2(\rho - \rho_v))}{2}, \tag{9}$$

in order to weaken the confidence of each assertion as the distance from the sensor increases (see Fig. 5). The parameter ρ_v plays the role of a 'visibility radius', where a smooth transition occurs from certainty to uncertainty. The motivation for introducing this function is twofold. First, as the possibility of multiple reflections increases as the beam makes a longer fly, the use of $m_2(\rho)$ reduces the undesirable effects of multiple reflections. Besides, narrow passages and doors appear to be obstructed if seen from a large distance, due to the sensor wide radiation angle. By varying the visibility radius according to the characteristics of the environment, it is possible to obtain a more correct detection behavior (see Sect. 3.4).

For each range measure r_i, two *local* fuzzy sets $\mathcal{E}_i$ and $\mathcal{O}_i$ are generated by combining the previously introduced certainty functions

$$\begin{aligned}\mathcal{E}_i(\rho,\vartheta) &= f_{\mathcal{E}}(\rho,r_i)\widetilde{m}_1(\vartheta)m_2(\rho)\\ \mathcal{O}_i(\rho,\vartheta) &= f_{\mathcal{O}}(\rho,r_i)\widetilde{m}_1(\vartheta)m_2(\rho).\end{aligned}$$

These sets represent, respectively, the certainty ascribed to the assertions 'empty' and 'occupied' in the radiation cone, on the basis of the sensor reading r_i. Note that the above membership functions are expressed in polar coordinates with respect to the sensor position, and assume nonzero values only inside the radiation cone. Figure 6 shows the typical shape of the two sets.

3.2 Aggregation

The local information contained in $\mathcal{E}_i$ and $\mathcal{O}_i$, which correspond to the range reading r_i, is aggregated in two *global* fuzzy sets $\mathcal{E}$ and $\mathcal{O}$, respectively the set of *empty* and *occupied* points, by using a fuzzy union operator

$$\mathcal{E} = \cup_i \mathcal{E}_i \tag{10}$$

$$\mathcal{O} = \cup_i \mathcal{O}_i. \tag{11}$$

The Dombi union operator was chosen for the above computation, due to its flexibility. By choosing λ in eq. (4), we may tune the aggregation strength. In principle, one could use different values of λ for $\mathcal{E}$ and $\mathcal{O}$.

During the aggregation procedure, it is necessary to perform a conversion from polar coordinates, relative to each sensor location, to absolute cartesian coordinates. For computational reasons, the universal set $U \subset I\!R^2$ is discretized as a matrix of $\beta = \sigma_1 \times \sigma_2$ square cells of side δ. As a result, two numerical values $\mu_{\mathcal{E}}(C)$ and $\mu_{\mathcal{O}}(C)$ are associated with each cell C, respectively quantifying its degree of membership to $\mathcal{E}$ and $\mathcal{O}$.

3.3 Computation of the fuzzy map

As already noted, the two sets $\mathcal{E}$ and $\mathcal{O}$ do *not* contain complementary information, due to the choice of a fuzzy logic framework. As a consequence, they can be combined to identify areas where the available information is either conflicting or insufficient.

The intersection of $\mathcal{E}$ and $\mathcal{O}$ is the set of *ambiguous* cells, with the corresponding membership value representing the degree of contradiction:

$$\mathcal{A} = \mathcal{E} \cap \mathcal{O}, \tag{12}$$

while the *indeterminate* cells are those neither *empty* nor *occupied*

$$\mathcal{I} = \bar{\mathcal{E}} \cap \bar{\mathcal{O}}. \tag{13}$$

A conservative map $\mathcal{S}$ of the *safe* cells is obtained by 'subtracting' the *occupied*, the *ambiguous* and the *indeterminate* cells from the *very empty* ones:

$$\mathcal{S} = \mathcal{E}^2 \cap \bar{\mathcal{O}} \cap \bar{\mathcal{A}} \cap \bar{\mathcal{I}}. \tag{14}$$

In fact, by squaring the value of the membership function of $\mathcal{E}$, the difference between low and high values is emphasized: according to the fuzzy logic terminology, we are applying the linguistic modifier 'very' to the 'empty' concept.

The fuzzy map $\mathcal{M}$ built by complementing $\mathcal{S}$ identifies *unsafe* cells, that must be avoided as much as possible during the robot motion:

$$\mathcal{M} = \bar{\mathcal{S}}.$$

For compactness, the membership degree of a cell C to the fuzzy set $\mathcal{M}$ will be simply denoted by $\mu(C)$.

The complement operator (1) and the bounded product intersection operator (2) are used to perform all the above computations. Note that the time complexity of the resulting fuzzy map building algorithm is linear in β, that is the number of cells of the bitmap representation.

3.4 Choice of the parameters

The values of the parameters of the certainty functions and of the Dombi union should be chosen on the basis of *(i)* the spatial distribution of the measurement locations *(ii)* the width of the sensor radiation cone, and *(iii)* the characteristics of the environment. However, a small amount of qualitative information is sufficient for successful tuning.

Typically, range readings are collected by a mobile robot through a ring of ultrasonic sensors. As already mentioned, we used the Nomad 200$^{\mathrm{TM}}$, equipped with 16 sonars. To increase measurement accuracy, at each position the ring is rotated twice by an angle of 7.5°: as a result, 48 range readings are obtained at each measurement location, and each cell falls at least in 3 radiation cones. Moreover, overlapping may occur between radiation cones corresponding to different measurement locations, so that often several readings are available for the same cell. Consider a particular cell C, and assume that it belongs to the 'empty' area of ν cones. Hence, each set $\mathcal{E}_i$ $(i = 1, \ldots, \nu)$ will contribute a nonzero value to $\mathcal{E}(C)$, that is computed according to eq. (10). Immediate saturation of $\mathcal{E}(C)$ should be avoided, so as to require the concordance of multiple measures to build up certainty. To this end, it is necessary

1. To set an upper bound on the membership function of $\mathcal{E}_i$ by choosing a 'small' value for $k_{\mathcal{E}}$ in eq. (6).

2. To select accordingly an appropriate λ for the Dombi union. Typically, λ is chosen in such a way that 3 or 4 concordant readings produce a high certainty value (see the plots in Fig. 1).

Similar considerations can be repeated for the choice of $k_{\mathcal{O}}$ in eq. (11).

As for the value of Δr in eqs. (6–7), it should be selected in such a way to 'augment' slightly the occupied area, providing a convenient safety margin for the planning phase.

Finally, the choice of the visibility radius ρ_v in eq. (9) can be made on the basis of a qualitative knowledge of the environment. For example, in an office-like area (as in our experiments) it is reasonable to ask that the map building algorithm correctly detects passages of the size of a door, say 0.7 m. To this end, the measure point must be sufficiently close to the opening, so that the 25° radiation cone does not intersect the door features. A simple geometric construction shows that the maximum admissible distance is 1.5 m. In this case, it would be therefore necessary to choose $\rho_v < 1.5$ m.

In our experience, the flexibility provided by these parameters largely overcomes the load associated with the tuning procedure.

3.5 Map building: Experimental results

As a testbed for the proposed approach, we have used the Nomad 200$^{\mathrm{TM}}$ mobile robot produced by Nomadic Technologies. As shown in Fig. 7, Nomad has a cylindrical shape with an approximate radius of 0.23 m, and a kinematic model equivalent to a unicycle. The upper turret, which carries the 16 Polaroid ultrasonic sensors, may be independently rotated. The robot control software runs under UNIX on an IBM RISC 6000 that communicates with Nomad through a radio link. The algorithms for map building and path planning have been implemented in the C language.

The experiment area was a rectangle of 18×12 m, containing a corridor and a room of our Department. To define the fuzzy map $\mathcal{M}$, this area was discretized into a matrix of 180×120 square cells of side $\delta = 0.1$ m. Halfway along the corridor there is an intersection, where a small obstacle is present. The open space is delimited by smooth surfaces (walls and closed glass cabinets) with poor diffraction properties, an adverse condition for ultrasonic sensing. Nomad has been driven to 43 measure points, whose position was determined a priori. At each measure point, a complete sensing procedure was performed, providing $16 \times 3 = 48$ range readings.

We have used the following set of parameter values: $k_{\mathcal{E}} = 0.1$, $k_{\mathcal{O}} = 0.25$, $\Delta r = 0.15$ m, $\rho_v = 1.2$ m, and $\lambda = 0.4$. Figure 8 shows the density plot of the resulting fuzzy map $\mathcal{M}$. The total time necessary to compute $\mathcal{M}$ from the range readings was approximately 13 s. Note the satisfactory accordance of the map with the actual boundary of the open space. The small gray areas extending beyond the corridor walls are due to multiple reflections occurring for large angles of incidence. Nevertheless, the map building algorithm was able to reconstruct accurately the profile of the walls, by incorporating the range readings obtained for incidence angles smaller than the critical value. To take into account the actual Nomad dimensions, the augmentation procedure described at the end of the previous section was performed with $\gamma = 0.23$ m, giving $\eta = 5$. The density plot of the augmented map $\mathcal{M}_a$ is reported in Fig. 9.

4 Fuzzy vs. probabilistic map building

At this point, a comparison with other map building techniques is in order. As already mentioned, existing methods for map building from ultrasonic sensor measures are often based on stochastic techniques. This approach was introduced in [4] and subsequently refined by Elfes, who provided a more firm theoretical foundation [5]. Related methods have been proposed in [8, 6].

In the discussion below, we shall use a simple example to compare the performance of our map building algorithm with the method of [5], whose basic features will be briefly recalled. This comparison is not intended as a contribution to the long-standing debate on fuzzy set theory vs. probability theory, for which the reader may consult, for example, [22, 23, 24] and the references therein. Our purpose is simply to emphasize the different behavior of the two methods with respect to a problem often encountered in practice, that is the occurrence of *outliers* in the measuring process.

In [5], the environment map is represented as an *occupancy grid*, containing probabilistic estimates of the occupancy state of the cells. In particular, the random variable $s(C)$ describing the state of cell C can assume two values, namely EMP (empty) and OCC (occupied). The occupancy grid is modeled as a discrete-state Markov field, a realization of which consists in the estimation, for each cell C, of the conditional probability $P[s(C) = \text{OCC} \,|\, r]$, i.e., the probability of cell C being occupied, given that a range reading r has occurred. In turn, $P[s(C) = \text{OCC} \,|\, r]$ is computed by using Bayes theorem on the basis of *(i)* a stochastic model of the sensor in the form of a probability density function $p[r \,|\, z]$, characterizing the probability of measuring r given that the closest obstacle to the sensor is located at a distance z, and *(ii)* a priori estimates of the probabilities $P[s(C) = \text{EMP}]$ and $P[s(C) = \text{OCC}]$, with the constraint

$$P[s(C) = \text{EMP}] + P[s(C) = \text{OCC}] = 1.$$

As successive range readings $r_1, r_2, \ldots, r_n$ become available, the conditional probabilities $P[s(C) = \text{OCC} \,|\, \{r_1, r_2, \ldots, r_n\}]$ are determined via the sequential Bayes formula, and used to update the occupancy grid.

Two general remarks are in order. First, to keep the problem tractable, one has to make the assumption that the random variables $s(C)$ are independent (zero-order Markov field), i.e., that no relationship whatsoever exists between the states of two cells C_i and C_j, even if they are adjacent. Second, the prior probabilities $P[s(C) = \text{EMP}]$ and $P[s(C) = \text{OCC}]$ are typically estimated with the *maximum entropy* assumption, or

$$P[s(C) = \text{EMP}] = P[s(C) = \text{OCC}] = 1/2.$$

Most criticisms raised to the Bayesian approach proceed from the discussion of the validity of these two assumptions. As for the independence assumption, it has been observed that it may induce large errors even in the presence of a slight degree of dependence between the random variables (see [23]). We believe that this is the case for practical map building, since the occupied cells are not evenly distributed, but concentrated in clusters (obstacles). Furthermore, in the absence of a reliable estimate of the prior probabilities, and especially with the maximum entropy assumption, the convergence of the Bayesian updating procedure towards an acceptable characterization of the occupancy grid requires a large number of measures.

Consider now a simple one-dimensional environment, that we attempt to reconstruct using a range finder located in a fixed position. In Figs. 10 and 11, the environment is the positive x axis, and the sensor is located at the origin. We shall assume that the rangefinder provides a succession of ten range readings $\{r_1, r_2, \ldots, r_{10}\}$, with $r_1 = r_2 = \ldots = r_9 = 0.6$ m, and $r_{10} = 1.2$ m. In our experience, similar measure patterns often occur, due to the aforementioned phenomenon of multiple reflections. While detecting and rejecting the outlier value $r_{10} = 1.2$ m would seem straightforward in the considered example, it is far from being so in practical situations, especially when the available measures are obtained from different positions.

We have first applied the stochastic map building method, choosing the sensor model $p[r \mid z]$ as a Gaussian density function with mean value z and variance 0.1. Figure 10 shows the profiles of the occupancy probabilities after 1,3,5,7,9, and 10 readings (the latter is the final occupancy grid). Note that the method correctly uses the first nine readings to build up certainty about the occupancy state of cells located in the vicinity of 0.6 m. However, when the tenth reading is processed, all the previous knowledge is erased and the final occupancy grid takes into account only the last range reading r_{10}. This fact, due to the structure itself of the sequential Bayes updating formula, shows that the stochastic map building method is prone to serious errors in the presence of multiple reflections. In principle, it should be possible to make the method more robust by using more complicated probability density functions to model the sensor [1, 5]. However, this requires a careful calibration of the model through several preliminary experiments to be performed in the considered environment.

The fuzzy map $\mathcal{M}$ built with our method after 1,3,5, and 10 range readings is shown in Fig. 11. To allow a fair comparison, we defined the set of *safe* cells simply as

$$\mathcal{S} = \mathcal{E} \cap \bar{\mathcal{O}},$$

in place of the complete expression (14), while $\mathcal{M} = \bar{\mathcal{S}}$ as before. Besides, both the radial and—of course—the angular modulation functions $\widetilde{m}_1(\vartheta)$ and $m_2(\rho)$ have been excluded. The parameters in $f_{\mathcal{E}}(\rho, r)$ and $f_{\mathcal{O}}(\rho, r)$ have been chosen as $k_{\mathcal{E}} = k_{\mathcal{O}} = 0.2$ and $\Delta r = 0.15$ m, while $\lambda = 0.4$ for the Dombi union. The behavior of our method is very different from that of the stochastic method. In fact, cells located in the vicinity of 0.6 m are considered to be unsafe already from the very first readings, while the area closer to the sensor becomes increasingly safe as successive measures confirm the obstacle position. Note also that, contrary to the stochastic method, the cells beyond the detected obstacle are considered to be unsafe. This globally cautious attitude is consistent with our final goal, that is to use the obtained map for planning the motion of the mobile robot. Besides, the method appears to be robust with respect to outliers, as the tenth reading does not modify substantially the certainty about the existence of an obstacle at 0.6 m.

5 Safe paths on fuzzy maps

In order to prove the utility of fuzzy maps in autonomous mobile robotics, we need to present effective algorithms for planning paths on them. For clarity of exposition, we consider first the case of a point robot; however, this assumption will be removed later.

Assume that a fuzzy map $\mathcal{M}$ of the environment has been reconstructed from sensor readings following the approach outlined in the last section (as a matter of fact, the planning technique presented in this section can be applied to any gray-level map, independently from the chosen map building method). A start cell S and a goal cell G are given, that specify respectively the initial and the desired final position of the robot. A path from S to G is a sequence of adjacent cells $\{S, \ldots, G\}$.

A distinctive feature of the problem, which our planner must explicitly take into account, is the uncertain nature of fuzzy maps, that do not provide a separation between the free and the occupied space. In principle, all cells are admissible for planning; a natural strategy is then to avoid cells with large values of μ, to minimize the risk of traversing cells corresponding to detected obstacles. This may be achieved by defining proper *cost functions* for a path P, and then searching for minimum-cost paths.

In the following, we introduce three different cost functions. The first, and most intuitive, is defined as

$$g_1(P) = \sum_{C_i \in P} \mu(C_i), \tag{15}$$

that is a measure of the integral risk along the path.

As a second cost function, we propose

$$g_2(P) = \sum_{C_i \in P} \mu^2(C_i). \tag{16}$$

Again, this corresponds to applying the modifier 'very' to the 'unsafe' concept. As a consequence, the inclusion of cells with high values of μ in a minimum-cost path is less likely.

Finally, consider

$$g_3(P) = \max_{C_i \in P} \mu(C_i), \tag{17}$$

that is the maximum risk encountered on the path.

While the minimization of g_1, g_2 and g_3 is intuitively appealing, it can also be given a formal significance. In fact, all these cost functions may be interpreted as *energy measures* [7] of the fuzzy set $\mathcal{M}$ over the path P; namely, they characterize the distance of P from an ideally safe path, i.e., a path for which $g_i = 0$ $(i = 1, 2, 3)$. In particular, according to the terminology of [25], g_1 and g_3 are respectively the *power* and the *height* of the fuzzy set $\mathcal{M}$ over P.

The three cost functions measure the above distance according to different metrics, so that each of them has a peculiar rationale for motion planning. For example, assume that S and G are far apart on $\mathcal{M}$, so that any admissible path connecting them consists of a large number of cells. Since the value of g_1 depends on the length of the path through the number of addends, a path minimizing g_1 will yield a trade-off between length and risk, and may traverse a cell with a large μ, if in this way its length were significantly reduced. A more conservative strategy would be to minimize g_3, even if this might result in a longer path. The use of g_2 is expected to generate paths with somehow intermediate attributes between those produced using g_1 and g_3.

On the other hand, consider a situation in which S is located in a zone of $\mathcal{M}$ with uniform high values of μ. A path that is optimal with respect to g_3 might be unsatisfactory, since the maximum value of μ is likely to be attained in proximity of S. From there on, any subpath leading to G is admissible, as long as it does not increase the value of g_3. Minimization of g_1 or g_2 should produce better results, since both these cost functions depend on the *whole* path from S to G. The results of Sect. 6.2 will further clarify the above discussion.

6 Path planning on fuzzy maps

After the discussion of last section, our version of the motion planning problem can be stated as 'Find a minimum-cost (g_1, g_2 or g_3) path from S to G on $\mathcal{M}$'. As a planning method, we have adopted the $A^\star$ algorithm [26]. Under a mild condition, this algorithm is *complete*, that is to say, it is guaranteed to return a minimum-cost path whenever one exists, and to return failure otherwise. It also allows to incorporate heuristic information when available, resulting in an efficient search.

While the application of $A^\star$ for determining minimum-g_1 or g_2 paths is straightforward, its use to minimize g_3 is somewhat non-standard, and deserves some comments. In the following, we briefly recall the fundamental features of the basic algorithm, and then discuss how it can be used to solve the three instances of our planning problem.

6.1 Planning safe paths via the $A^\star$ algorithm

Given a graph whose arcs are labeled by costs, the cost of a path from S to G is defined as the sum of the costs of its arcs. $A^\star$ makes use of an evaluation function $e(N)$ associated with each node N, that estimates the cost of the optimal path from S to G constrained to go through N. The estimate $e(N)$ is the sum of two parts:

$$e(N) = g(N) + h(N),$$

where $g(N)$ is the cost of the *current* path from S to N, and $h(N)$ is a *heuristic* estimate of the cost of the optimal path from N to G. The graph is iteratively searched from

S. Any node that has been visited but not expanded (i.e., whose successors have not been all visited *from that node*) is placed in a OPEN list, sorted by increasing values of e. At each iteration, the algorithm removes and expands the first node N of OPEN. The following procedure is repeated for each successor N': if N' has not been visited so far, it is inserted in OPEN; otherwise, it must be checked if the path leading to N' through N is more convenient. This is done using the test

$$g(N') > g(N) + k(N, N'), \tag{18}$$

where $k(N, N')$ is the cost associated with the arc joining N to N'. The algorithm assigns to each visited node N' a pointer to its *parent*, namely its predecessor on the current path from S to N'. If eq. (18) is verified, then the value of $e(N')$ (and hence the list OPEN) is updated and the pointer of N' is redirected to N. In this way, only a spanning tree T of the graph is memorized.

$A^\star$ is complete under the simple condition

$$0 \leq h(N) \leq h^*(N), \quad \forall N, \tag{19}$$

where $h^*(N)$ is the *actual* cost of the minimum-cost path from N to G. A heuristic function that satisfies the condition (19) is said to be *admissible*. When no estimate of $h^*(N)$ is available, it is possible to set $h(N) = 0$, resulting in a *non-informed* $A^\star$, or Dijkstra's algorithm. Besides, if the heuristic function is *locally consistent*, i.e., for any pair of adjacent nodes (N, N') it is

$$0 \leq h(N) \leq h(N') + k(N, N'),$$

then, whenever a node N is selected for expansion, the current path from S to N in T is already optimal. Again, $h(N) = 0$ is trivially consistent.

The use of $A^\star$ to generate paths minimizing g_1 on the fuzzy map $\mathcal{M}$ is straightforward. The cost of an arc joining two adjacent cells C_i and C_j is defined as

$$k_1(C_i, C_j) = \mu(C_j),$$

so that the cost of a path P coincides with $g_1(P)$, except for the additive constant $\mu(S)$. As for the heuristic function, we use

$$h_1(C_j) = d(C_j) \cdot \mu^{min}, \tag{20}$$

in which $d(C_j)$ is the minimum number of cells that compose a subpath from C_j to G, and μ^{min} is the smallest value of μ over $\mathcal{M}$ (a quantity that can be memorized during the map building phase). The heuristic function (20) is clearly admissible.

Two remarks are necessary at this point.

- The value $d(C_j)$ depends on the adjacency definition on the map. If 1-adjacency is used (each cell has four adjacent cells), we have

$$d(C_j) = |x_G - x_j| + |y_G - y_j|,$$

where (x_j, y_j) and (x_G, y_G) are respectively the coordinates of C_j and G. When using 2-adjacency (each cell has eight adjacent cells), one has

$$d(C_j) = \max(|x_G - x_j|, |y_G - y_j|).$$

- In order to obtain an *informed* $A^\star$, it must be $\mu^{min} > 0$. Hence, it may be advisable to offset all values of μ by a small positive constant.

The resulting version of the $A^\star$ algorithm will be denoted by $A_1^\star$. The algorithm $A_2^\star$ for minimization of g_2 is obtained by trivial modifications of $A_1^\star$.

Consider now the case of the cost function g_3. The difficulty in applying $A^\star$ comes from the non-additive nature of this function. First, it is necessary to define the arc cost $k_3(C_i, C_j)$ in such a way that, adding the costs of all arcs of a path, the result is g_3. To this end, let

$$k_3(C_i, C_j) = \begin{cases} \mu(C_j) - \hat{\mu}(C_i) & \text{if } \mu(C_j) > \hat{\mu}(C_i) \\ 0 & \text{else,} \end{cases}$$

where $\hat{\mu}(C_i)$ denotes the maximum value of μ encountered on the optimal path from S to C_i. Assuming that a locally consistent heuristic function h is used, when C_i is expanded, the current path from S to C_i is optimal: hence, it is $\hat{\mu}(C_i) = g(C_i)$. Thus, even if the arc costs are not known *a priori*, they are uniquely defined and computable as the algorithm proceeds.

The choice of the heuristic function is constrained by the admissibility condition. In this case, it is necessary to set $h_3(C_j) = 0$, since it may happen that $g(G) = g(C_j)$ (the maximum value of μ on the path is attained on the subpath from S to C_j). However, a non-informed $A^\star$ may be dramatically inefficient, due to the large number of *ties* that may occur in OPEN. To cope with this problem, we have used a tie-resolution strategy privileging cells whose distance $d(\cdot)$ from G is smaller. With this modification, $A^\star$ works like a *depth-first* search method whenever this is possible without increasing the value of g_3. The algorithm just described will be indicated by $A_3^\star$.

At this point, we can remove the point robot assumption and take into account the actual robot dimensions with a simple modification. Assume that the robot can be approximated by a circle of radius γ, whose center is located at (the center of) cell C. Since each bitmap cell has side δ ($\delta < \gamma$), the robot body will be contained in a square of $\eta \times \eta$ cells centered at C, being η an odd number computed as

$$\eta = 2\left(\text{round}\left(\frac{\gamma}{\delta}\right) + 0.5\right),$$

where round(x) is the nearest integer to x. Hence, we can build an *augmented* map $\mathcal{M}_a$ as follows: for each cell C, $\mu_a(C)$ is defined as the maximum value of μ attained in the square of $\eta \times \eta$ cells centered at C. Planning a path for a point in $\mathcal{M}_a$ is equivalent to plan paths for the actual robot in $\mathcal{M}$. Such procedure may be implemented as a pre-processing phase on the map $\mathcal{M}$, but this is not necessary. In fact, it is sufficient to modify $A^\star$ so that the value of $\mu_a(C)$ is computed only when cell C is visited.

Finally, a comment about the complexity of the planning algorithm. For a graph with a arcs and n nodes, the time complexity of $A^\star$ is $O(a \log n)$. Since our representation of the environment is a bitmap of β cells, the number of arcs is $a = 2\beta^2 - 3\beta + 1$ when 2-adjacency is used. The resulting time complexity is $O(\beta^2 \log \beta)$.

6.2 Path planning on fuzzy maps: Simulation results

The proposed path planning method has been applied to the map of Fig. 8. The results for a first trial case are reported in Fig. 12, while details on the solution paths are given in Tab. 1. The start cell S is located inside the room, while the goal cell G is at the end of the left branch of the corridor. The straight line distance between S and G is approximately 12 m. Note that, even if the paths produced by $A_1^\star$ and $A_2^\star$ coincide, the computing time differs substantially for the two algorithms. As for $A_3^\star$, the solution path is more dangerous in the integral sense (the value of g_1 is roughly doubled with respect to $A_1^\star$ and $A_2^\star$), but the maximum risk along the path is 0.5, as before. This value is encountered in correspondence of the narrow passage between the room and the corridor (recall that planning is performed on the augmented map $\mathcal{M}_a$).

The algorithm $A_3^\star$, although non-informed, is much faster than the other two algorithms. This is due to the tie-resolution strategy, that forces a depth-first behavior when possible. As a drawback, while 2-adjacency was chosen for $A_1^\star$ and $A_2^\star$, we had to use 1-adjacency for $A_3^\star$, to prevent the path from grazing the walls.

To better illustrate the different behavior of the algorithms, we have considered a second case, with the start cell S and the goal cell G located on the opposite sides of the small obstacle, as shown in Fig. 13 on a magnified view of $\mathcal{M}$. Here, planning is performed directly on $\mathcal{M}$, i.e., for a point robot. The risk value $\mu(S)$ associated with the start cell is 0.5. Details on the performance of the algorithms are given in Tab. 2. Note that the path produced by $A_1^\star$ turns around the obstacle in the counterclockwise direction, while those generated by $A_2^\star$ and $A_3^\star$ take the clockwise direction. In particular, $A_1^\star$ returns a (shorter) path with the lowest possible value of integral risk g_1, but traversing more dangerous cells, as indicated by the value of $g_3 = 0.6$. Instead, the risk over the $A_3^\star$ path does never increase beyond the initial value $\mu(S) = 0.5$. Interestingly, $A_2^\star$ displays in this case an intermediate behavior between the other two algorithms. In fact, while the maximum risk attained along the path is again $\mu(S) = 0.5$, the integral risk is lower than for $A_3^\star$, since each cell in the path contributes to the cost function g_2.

The time necessary to compute a solution path may vary remarkably. Indeed, time efficiency was not stressed in our implementation of $A^\star$. A considerable increase in speed could be obtained by using a sub-optimal version of $A^\star$, as proposed in [27]. In general, $A_3^\star$ is faster in returning a solution path, provided that no backtracking phase is necessary. On the other hand, $A_1^\star$ and $A_2^\star$ produce generally safer and smoother paths.

7 Conclusion

The concept of fuzzy map has been presented as a new tool for perception and planning with autonomous mobile robots. The main points of our presentation are highlighted below.

- An intrinsically uncertain representation of the environment is used. In particular, fuzzy set operators are used to process ultrasonic sensor data, obtaining a grey-level bitmap $\mathcal{M}$ that conveys a risk information for each cell.
- Three cost functions g_1, g_2 and g_3 have been introduced to characterize the risk of collision along a path P. All these may be interpreted as energy measures of $\mathcal{M}$ on P, therefore characterizing the distance of P from an ideally safe path. However, each cost function has a different rationale for motion planning.
- Planning on a fuzzy map $\mathcal{M}$ can be performed by searching for optimal paths. Proper instances of the $A^\star$ class of search algorithms have been identified for the minimization of g_1, g_2 and g_3.

Experimental results have been reported to illustrate the satisfactory performance of the proposed method, showing the applicability of fuzzy logic methodologies in the field of autonomous mobile robots. The obtained maps were quite accurate even in hostile environments, where multiple reflections often occur. In turn, the $A^\star$-based planner is efficient and produces safe paths. Space for adaptation is provided by the possibility of tuning various parameters as well by the choice of the path cost function.

Among the current research directions, we mention *(i)* the integration of information coming from other sensors (e.g., laser and infrared rangefinders), that is straightforward within the fuzzy logic framework *(ii)* a localization procedure with respect to the reconstructed map $\mathcal{M}$, in the spirit of [28] *(iii)* the possible use of a different planning method. In [29] it was shown how to define *navigation functions* on fuzzy maps to perform potential-based motion planning, following the approach of [30].

Finally, we are applying the presented techniques to perform *on-line* map building and navigation: local maps of the environment are built and integrated as the robot moves towards the goal as prescribed by the $A^\star$ algorithm. Two processes are repeatedly performed in sequence: perception and navigation. In the former, the robot acquires data from its sensors and updates accordingly its environment maps. In the latter, the planning module proposes a path to the goal that is safe within the area explored so far, and the robot follows it until a new perception phase is necessary. The preliminary results of this investigation are encouraging, as reported in [16].

Acknowledgments

This work was partially supported by CNR-PF Robotica funds and by ESPRIT BR Project 6546 (PROMotion).

References

[1] H. F. Durrant-Whyte, *Integration, Coordination and Control of Multi-Sensor Robot Systems*, Kluwer Academic Publishers, Norwell, MA, 1988.

[2] I. J. Cox and G. T. Wilfong, Eds., *Autonomous Robot Vehicles*, Springer-Verlag, New York, NY, 1990.

[3] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, Boston, MA, 1991.

[4] A. Elfes and H. P. Moravec, "High resolution maps from wide angle sonar," *Proc. 1985 IEEE Int. Conf. on Robotics and Automation*, St. Louis, MO, 1985, pp. 116–121.

[5] A. Elfes, "Occupancy grids: A stochastic spatial representation for active robot perception," in *Autonomous Mobile Robots: Perception, Mapping, and Navigation*, S. S. Iyengar and A. Elfes, Eds., IEEE Computer Society Press, Los Alamitos, CA, 1991, pp. 60–71.

[6] D. W. Cho, "Certainty grid representation for robot navigation by a Bayesian method," *Robotica*, **8**, pp. 159–165, 1990.

[7] H. Bandemer and W. Näther, *Fuzzy Data Analysis*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1992.

[8] J. J. Leonard and H. F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*, Kluwer Academic Publishers, Norwell, MA, 1992.

[9] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. of Robotics and Automation*, **RA-2**, 1, pp. 14–23, 1986.

[10] R. C. Arkin, "Integrating behavioral, perceptual, and world knowledge in reactive navigation," *Robotics and Autonomous Systems*, **6**, pp. 105–122, 1990.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. of Robotics Research*, **5**, 1, pp. 90–98, 1986.

[12] D. W. Payton, "Internalized plans: A representation for action resources," *Robotics and Autonomous Systems*, **6**, pp. 89–103, 1990.

[13] J. Borenstein and Y. Koren, "The vector field histogram—fast obstacle avoidance for mobile robots," *IEEE Trans. on Robotics and Automation*, **7**, 3, pp. 278–288, 1991.

[14] A. Stentz, *The Navlab system for mobile robot navigation*, Ph. D. Thesis, Carnegie Mellon University, School of Computer Science, 1990.

[15] E. Gat, "Integrating planning and reacting in a heterogeneous asynchronous architecture for mobile robot navigation," *Proc. 10th Nat. Conf. on Artificial Intelligence*, San Jose, CA, 1992.

[16] G. Oriolo, G. Ulivi, and M. Vendittelli, "On-line map-building and navigation for autonomous mobile robots," *Proc. 1995 IEEE Int. Conf. on Robotics and Automation*, Nagoya, J, 1995, pp. 2900–2906.

[17] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision process," *IEEE Trans. on Systems, Man, and Cybernetics*, **SMC-3**, 1, pp. 28–44, 1973.

[18] G. J. Klir and T. A. Folger, *Fuzzy Sets, Uncertainty and Information*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[19] H.-J. Zimmermann, *Fuzzy Set Theory—and Its Applications*, Kluwer Academic Publishers, Norwell, MA, 1991.

[20] J. Dombi, "A general class of fuzzy operators, the De Morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators," *Fuzzy Sets and Systems*, **8**, pp. 149–163, 1982.

[21] Polaroid Corporation, *Ultrasonic Ranging System*, 1987.

[22] J. R. Quinlan, "Consistency and plausible reasoning," *Proc. 8th Int. Joint Conf. on Artificial Intelligence*, Karlsruhe, Germany, 1983, pp. 137–144.

[23] S. J. Henkind and M. C. Harrison, "An analysis of four uncertainty calculi," *IEEE Trans. on Systems, Man, and Cybernetics*, **SMC-18**, 5, pp. 700–714, 1988.

[24] P. P. Bonissone, "Reasoning, plausible," in *The Encyclopedia of Artificial Intelligence*, 2nd Edition, S. Shapiro, Ed., John Wiley and Sons, New York, NY, 1991, pp. 1307–1322.

[25] A. De Luca and S. Termini, "Entropy and energy measures of a fuzzy set," in *Advances in Fuzzy Sets and Applications*, M. M. Gupta, R. K. Ragade, and R. R. Yager, Eds., North-Holland, 1979, pp. 321–338.

[26] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. on Systems, Science, and Cybernetics*, **SSC-4**, 2, pp. 100–107, 1968.

[27] C. W. Warren, "Fast path planning using modified $A^\star$ method," *Proc. 1993 IEEE Int. Conf. on Robotics and Automation*, Atlanta, GA, 1993, pp. 662–667.

[28] P. Mackenzie, G. Dudek, "Precise positioning using model-based maps," *Proc. 1994 IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, 1994, pp. 1615–1621.

[29] G. Oriolo, G. Ulivi, and M. Vendittelli, "Potential-based motion planning on fuzzy maps," *Proc. Second European Cong. on Intelligent Techniques and Soft Computing (EUFIT'94)*, Aachen, D, 1994, pp. 731–735.

[30] J. Barraquand, B. Langlois, and J.-C. Latombe, "Numerical potential fields techniques for robot path planning," Report No. STAN-CS-89-1285, Stanford University, CA, 1989.

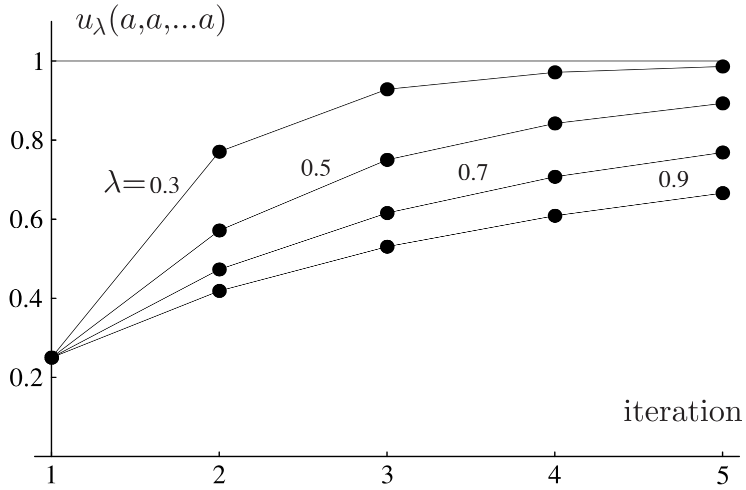

Figure 1: Behavior of the Dombi union operator for different values of λ. The dots show the result of the iterated application of u_λ to $a = 0.25$.

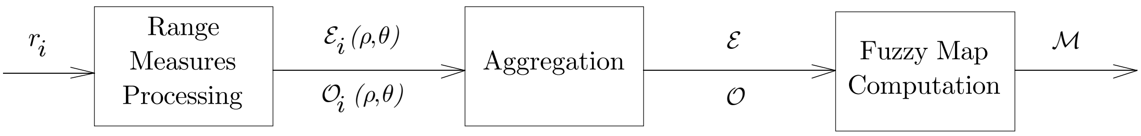

Figure 2: Basic steps of the map building algorithm.

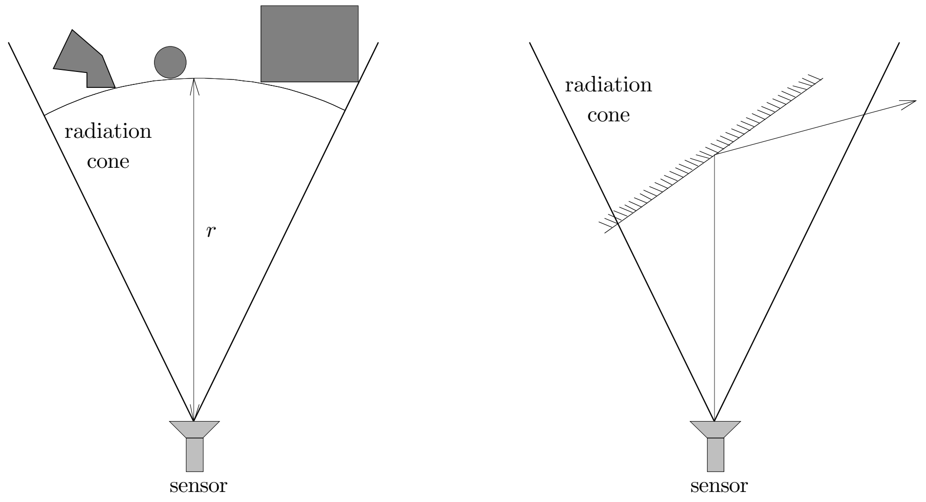

Figure 3: Ultrasonic sensing: objects in different positions can give the same distance reading r (left); multiple reflections may occur for large angles of incidence (right).

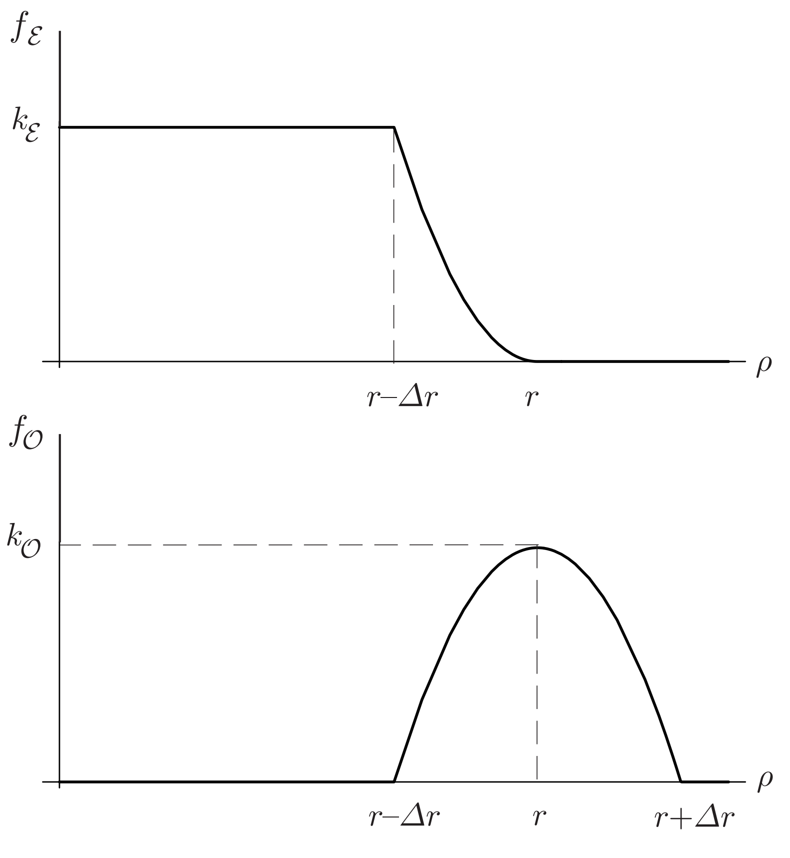

Figure 4: The two certainty functions $f_{\mathcal{E}}$ (above) and $f_{\mathcal{O}}$ (below) for a range reading r.

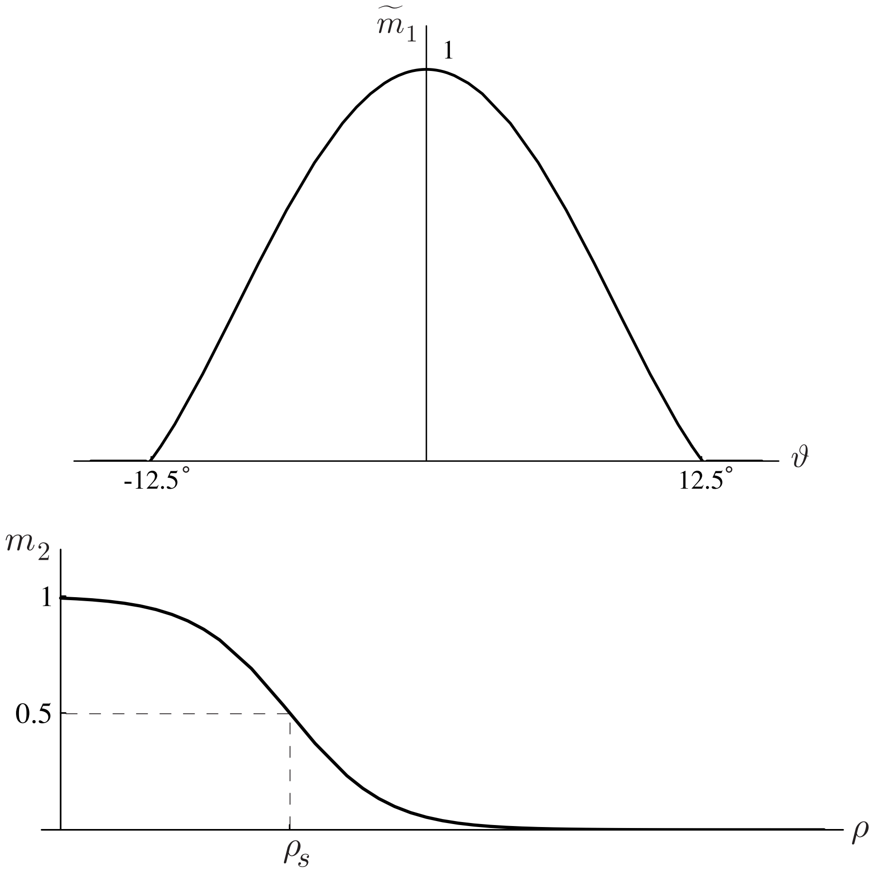

Figure 5: The angular modulation function $\widetilde{m}_1(\vartheta)$ (above) and the radial modulation function $m_2(\rho)$ (below).

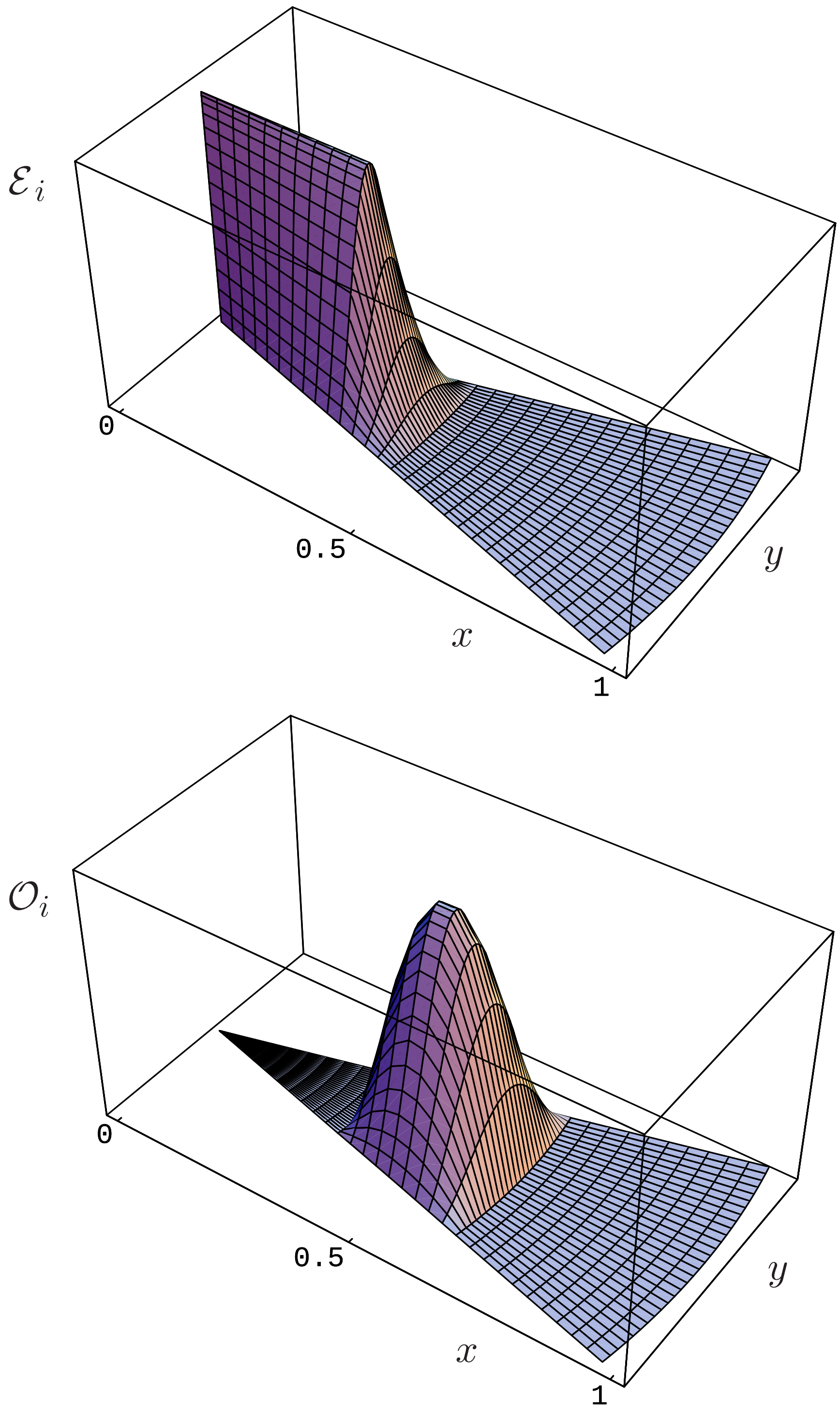

Figure 6: Membership functions for $\mathcal{E}_i$ (above) and $\mathcal{O}_i$ (below) inside the sensor radiation cone. The sensor reading is assumed to be $r_i = 0.5$ m.

Figure 7: The Nomad 200^{TM} mobile robot.

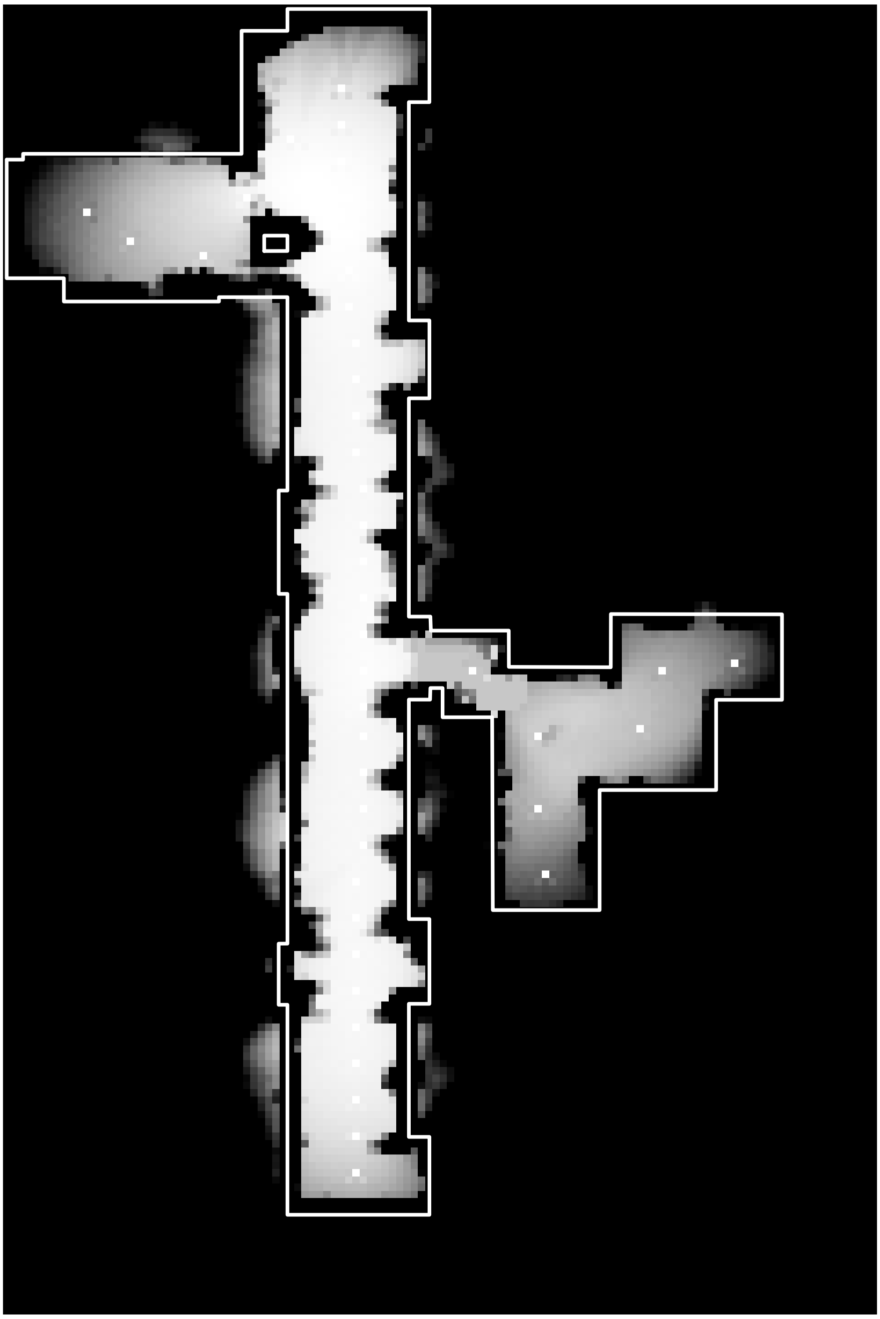

Figure 8: Density plot of the fuzzy map $\mathcal{M}$ obtained from our experiments: darker areas correspond to higher values of μ. The actual profile of the corridor, the room and the obstacle are superimposed. White spots indicate the measure points.

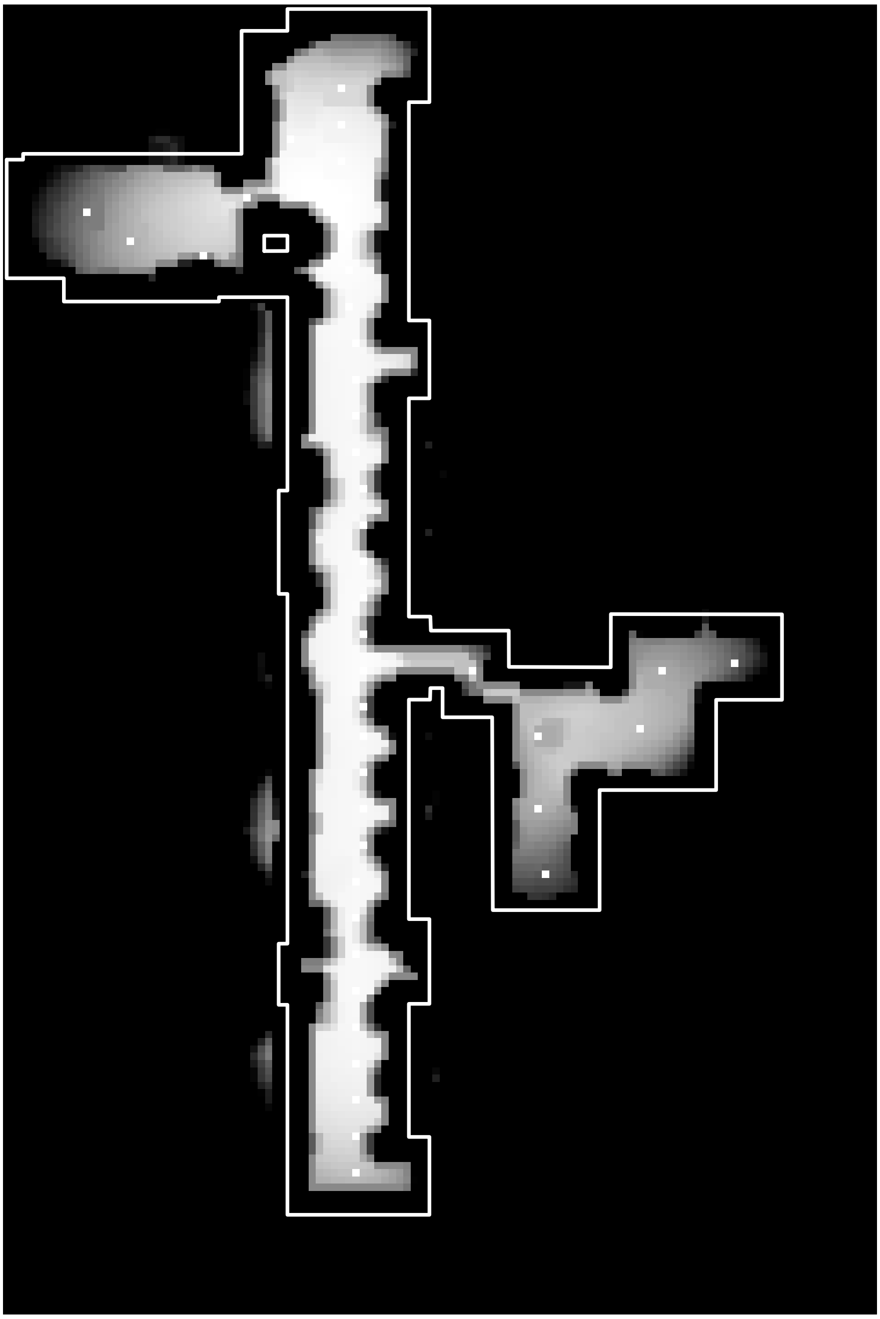

Figure 9: Density plot of the augmented fuzzy map $\mathcal{M}_a$ for $\eta = 5$.

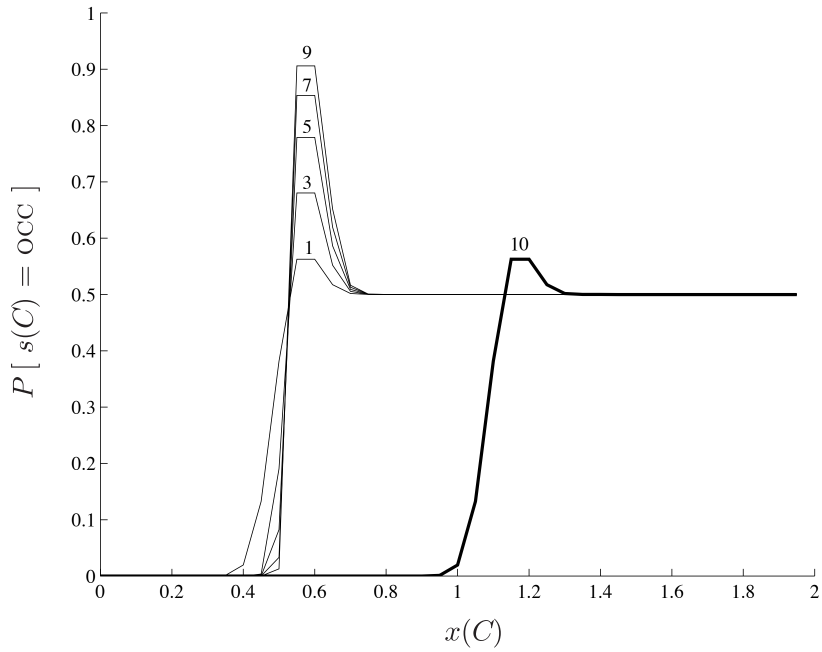

Figure 10: Results of the stochastic map building method for a simple one-dimensional example. The sensor is located at the origin.

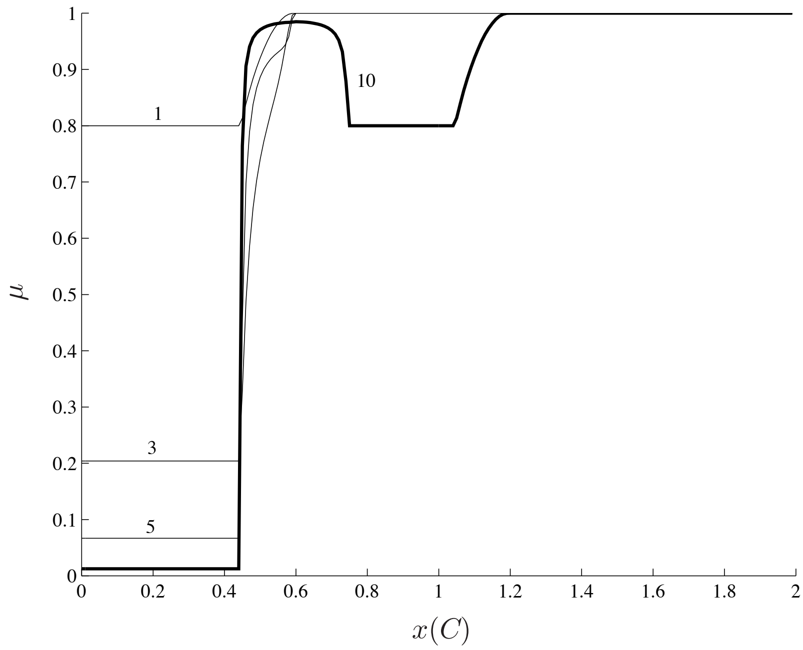

Figure 11: Results of the fuzzy map building method for the same example.

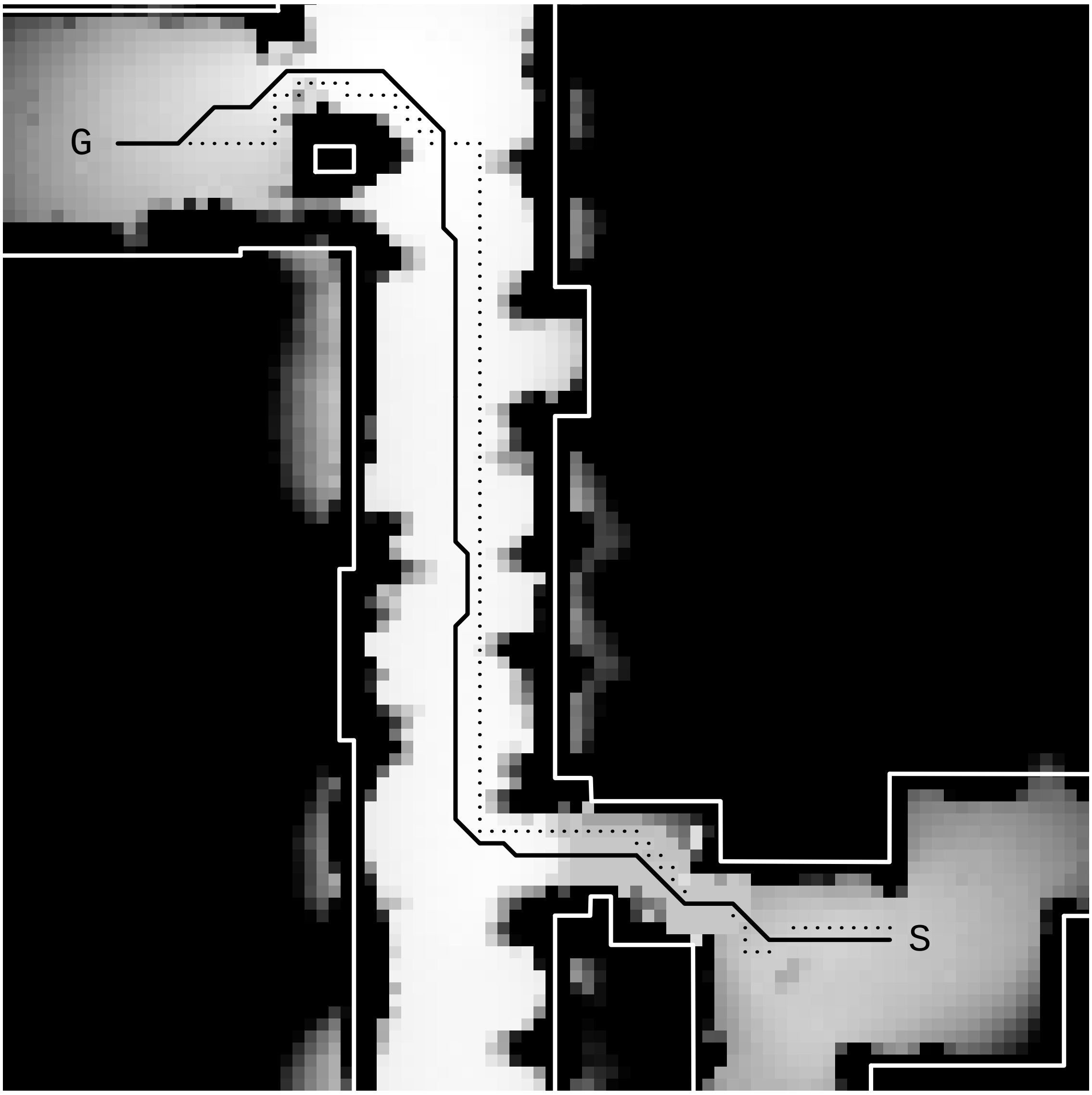

Figure 12: Case 1: Paths generated by $A_1^\star$ and $A_2^\star$ (continuous) and $A_3^\star$ (dotted).

	g_1	g_2	g_3	cells in the path	expanded cells	computing time (sec)
$A_1^\star$	21.80	5.34	0.50	121	2066	3.13
$A_2^\star$	21.80	5.34	0.50	121	2394	6.21
$A_3^\star$	45.50	17.13	0.50	145	378	0.08

Table 1: Results for the first case.

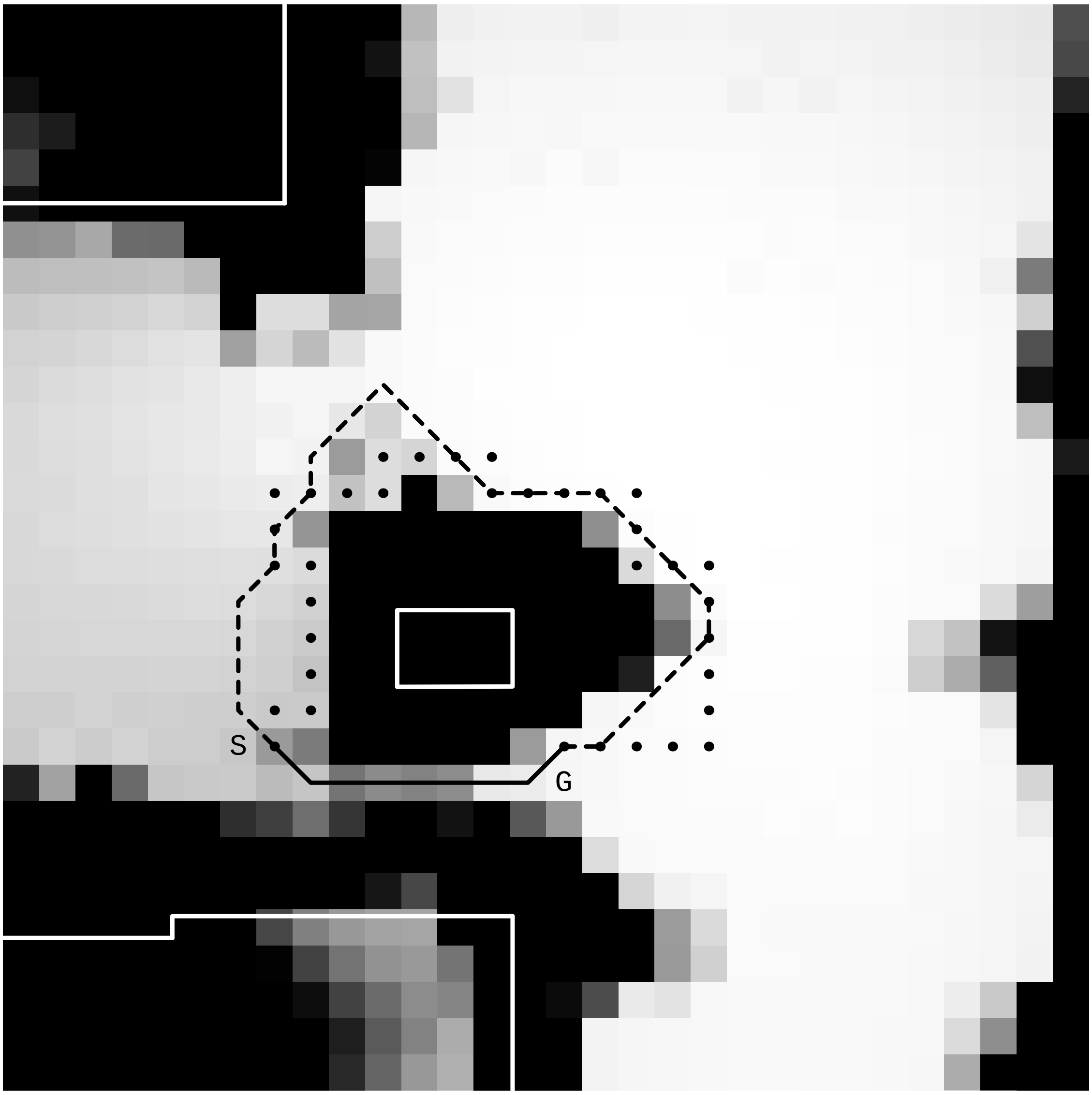

Figure 13: Case 2: Paths generated by $A_1^\star$ (continuous), $A_2^\star$ (dashed) and $A_3^\star$ (dotted).

	g_1	g_2	g_3	cells in the path	expanded cells	computing time (sec)
$A_1^\star$	3.1	1.43	0.60	9	108	0.09
$A_2^\star$	3.7	0.69	0.50	25	168	0.16
$A_3^\star$	5.8	1.22	0.50	35	65	0.01

Table 2: Results for the second case.

List of Figures

List of Tables