Applied Intelligence 19, 171–187, 2003

A Fuzzy Perceptual Model for Ultrasound Sensors Applied to Intelligent Navigation of Mobile Robots*

EUGENIO AGUIRRE AND ANTONIO GONZÁLEZ
Department of Computer Science and Artificial Intelligence, E.T.S. de Ingeniería Informática, University of Granada, 18071-Granada, Spain
eaguirre@decsai.ugr.es
gonzalez@decsai.ugr.es

Abstract. The way of understanding the role of perception along the intelligent robotic systems has evolved greatly since classic approaches to the reactive behavior-based approaches. Classic approaches tried to model the environment using a high level of accuracy while in reactive systems usually the perception is related to the actions that the robot needs to undertake so that such complex models are not generally necessary. Regarding hybrid approaches is likewise important to understand the role that has been assigned to the perception in order to assure the success of the system. In this work a new perceptual model based on fuzzy logic is proposed to be used in a hybrid deliberative-reactive architecture. This perceptual model deals with the uncertainty and vagueness underlying to the ultrasound sensor data, it is useful to carry out the data fusion from different sensors and it allows us to establish various levels of interpretation in the sensor data. Furthermore, using this perceptual model an approximate world model can be built so that the robot can plan its motions for navigating in an office-like environment. Then the navigation is accomplished using the hybrid deliberative-reactive architecture and taking into account the perceptual model to represent the robot's beliefs about the world. Experiments in simulation and in an real office-like environment are shown for validating the perceptual model integrated into the navigation architecture.

Keywords: robotics, fuzzy models, data fusion, environment modeling, behavior-based systems

1. Introduction

In the area of mobile robots, the hybrid deliberative-reactive solutions are aimed at an efficient integration of deliberative [1] and reactive skills [2]. Some hybrid architectures use different levels of abstraction setting a hierarchical structure. For example, in the three layer architectures [3–5] usually the control is situated at the lowest level whereas the deliberation is situated at the highest level. In the intermediate level a layer executes the actions according to the calculated plan, the robot's state and the perceived world. The plan is previously calculated by a planner situated at the highest level that uses some world modeling which is needed to form and optimize the plan. One outcome of a 1995 workshop on robot architectures [6] was the observation that the interface or middle layer between the two components of such an architecture is a main function since it links rapid reaction and long-range planning. In order to carry out this link, it is important to consider the role that the perception plays. For instance, the Autonomous Robot Architecture (AuRA) [7] incorporates a conventional planner that could reason over a modular behavior-based control system, integrating perceptual and a priori environmental knowledge. Here, the intermediate level was called the *plan sequencer* which translates each path leg into a set of motors behaviors for execution. Each motor behavior or schema is associated with a perceptual schema capable of providing

*This work has been supported by the CICYT under Project TIC2002-04196-C05-02.

the stimulus required for that particular behavior. Thus the action-oriented perception was the basis for the navigation [8]. In Atlantis architecture [4] the intermediate layer or *sequencer* handles initiation and termination of low-level activities and addresses possible failures to complete the task. The notion of cognizant failure is introduced [9] referring to the robot's ability to recognize on its own when it cannot complete its task. To achieve this, monitor routines are added to the architecture and these routines usually use perceptual information to determine if things are not going as they should go. In Saphira architecture [5] the center is the Local Perceptual Space (LPS), which is a geometric representation of space around the robot. This representation accommodates various levels of interpretation of sensor information, as well as a priori information from maps. Several semantic descriptions of the world or artifacts, such as corridors, walls or doorways are built from interpretation of sensors readings and are associated to LPS. These artifacts represents the Saphira's beliefs about the world and the most actions are planned and executed with respect to these beliefs.

Additionally, in these systems in order to carry out the actions of the plan, it is usual to set somehow, a relationship between the concepts of the abstract plan with the actual objects of the environment, which are sensed by the robot sensors. Thus, the perception in the context of behavior-based robotic systems is usually conducted on a need-to-know basis, that is the perception is strongly related to the actions that the robot needs to undertake. For instance the sensor-based behaviors implement control policies based on external sensing and then the robot moves with respect to features in the environment.

In this work we propose a fuzzy *perceptual model* that allows us to build an approximate world model giving us various levels of interpretation, the possibility for reasoning and planning about the robot motion in the world and allowing behavior-based navigation using a hybrid deliberative-reactive architecture. Firstly, the influence of multiple error sources is studied to establish a process that reduces the influence of these phenomenon and to define a new fuzzy *model sensor* that gives us a belief degree about the possible existence of a piece of wall that is being sensed perpendicularly in the direction to the sensor. Then fusing information gathered from different sensors, the contours around the robot can be determined and classified in *perceptual objects* like *walls*, *corridors*, *corners*, *hallways* and others distinguished places using a incremental process. Fuzzy logic [10] is the tool that is used to manage the uncertainty and vagueness of the sensor data and for modeling the different perceptual objects that will be detailed along this work. Taking into account these objects, a topological map of the environment can be built and then a planner will be able to find the best path that links the initial position to the final localization.

The uses of fuzzy logic in robotic systems to connect perception to action have been numerous [11–16] and several autonomous robots have been equipped with a wider set of fuzzy behaviors as the cases of autonomous robots FLAKEY [17] in which the architecture Saphira has been developed, and MARGE [18]. Regarding environment modeling using fuzzy logic some approaches deal with the modeling of the uncertain geometric robot environment [19] using points and lines whereas others built sonar map using segments whose width and length are trapezoidal fuzzy sets [20]. Also there are approaches that have proposed variants of Moravec and Elfe's occupancy grids [21] in which each cell is associated with a possibility distribution that expresses if a cell is free or occupied [22].

This paper has been organized in the following way. First, we briefly introduce the hybrid deliberative-reactive architecture and the set of behaviors that are used for navigating. Then, the process to reduce the influence of the noise and the fuzzy sensor model for modeling the possible existence of perpendicular walls are properly detailed. Upon this fuzzy sensor model then we develop the procedures to determine the perceptual objects or distinguished places setting the description of these places by means of fuzzy logic. In order to test the validity of this perceptual model in the real world several trials of distinguished places detection are carried out in an office-like environment. Also the model has been applied for navigating using our own deliberative-reactive architecture obtaining a safe behavior-based navigation, first in simulation and afterwards in a real environment. Finally some conclusions and future work lines are shown.

2. Overview of the Architecture

Our system is along the lines of behavior-based robots [23] and uses a three-level architecture to integrate deliberative techniques and reactive behaviors. This architecture is composed of three hierarchical layers: a planning, an executive and a control level. Figure 1 shows the relationship among these layers and the map of the environment.

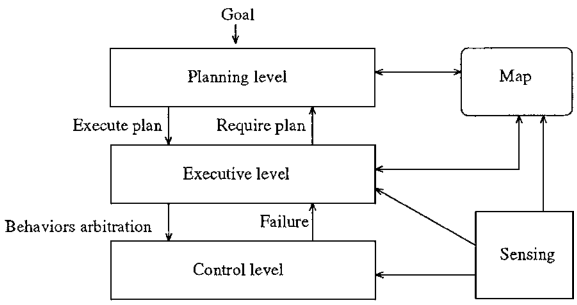

Figure 1. Three-level architecture and map.

The highest level must search for a safe and minimum-cost path from an initial position to a final desired position across an office-like environment which is expressed by means of a map that contains topological and geometrical information about the environment [24]. This map is constructed performing an exploratory task in order to discover several kinds of distinctive places, which are regions in the world that have characteristics that distinguish them from their surroundings. Formally the topological map consists of a graph $G = (V, E)$, where $V = \{v_1, \ldots, v_n\}$ is the set of N nodes, and $E = \{e_{ij}, i = 1 \ldots N, j = 1 \ldots N\}$, where $e_{ij} = (v_i, v_j)$, is the set of M edges. The nodes can be classified in one of the next five kinds of detectable objects: corner (c), door (d), hallway (h), end of corridor (ec) and a default object type corresponding to a long irregular boundary (i). The arcs are the walls (w) or corridors (co) that connect the previous nodes. Additionally, the arcs can only express connectivity among different nodes, as for example the arcs that link the doors, link doors to hallways, or link the irregular type nodes to any other kind of node. Both nodes and arcs have a descriptor which contains information about the object type, and if it is necessary, a fuzzy estimation of the object's length. An example of a topological map of the environment is shown in the experimental section by Fig. 12.

This topological map provides information directly gleaned from the robot's experiences within the world, but it can additionally contain any information obtained independently from the robotic agent itself, such as maps obtained from floorplans. This kind of information resolves, for example, the problem of detecting the presence of a staircase. The topological map can be used by the planner to compute a minimum-cost path from the current position to the desired goal taking into account the estimated length of each arc and using a standard graph search algorithm, such as Dijkstra's shortest path algorithm or A* algorithm.

The executive level must ensure the fulfillment of the plan by selecting which basic behaviors should be activated at a given time depending on the environment, the robot's state and the current goal. Thus, this level defines the context of applicability of certain behaviors using a set of metarules that address the perceptual objects that form the current context. For example, the plan computed to move the robot from a corridor to a corner at the end of a wall (the goal), may be defined as the robot having to follow the corridor *Corridor X* and after that following the wall *WallY* on the right until the final goal is reached, of course avoiding the possible obstacles. The translation to metarules is carried out by the executive level directly, coupling perceptual objects to behaviors. Thus the set of metarules will be as follows:

$$\begin{gathered}
\textit{If obstacle then Avoid obstacle}\\
\textit{If } \neg\textit{obstacle} \wedge \textit{In}(\textit{Corridor X}) \wedge \neg\ \textit{In}(\textit{WallY})\\
\textit{then Follow corridor}(\textit{Corridor X})\\
\textit{If } \neg\textit{obstacle} \wedge \textit{In}(\textit{WallY}) \wedge \neg\ \textit{In}(\textit{Goal})\\
\textit{then Follow wall}(\textit{WallY})\\
\textit{If } \neg\textit{obstacle} \wedge \textit{In}(\textit{Goal}) \textit{ then Stop}.
\end{gathered}$$

The last metarule is just to indicate the stop condition since *Stop* is not actually a behavior. In this example the contexts are given either by the presence or absence of an obstacle and by the detection of the corridor *Corridor X* or the wall *WallY*. The detection of the perceptual objects is based on the perceptual model proposed in this work which is used to generate the level of belief in the existence of the different objects.

In our architecture, reasoning is based on the path planning process because the relation between the perceptual objects and behaviors is natural and direct. This allows the robot to compute the best path in order to achieve the final goal, although the environment may have several paths joining the current position and the goal. This question is an important difference between our system and the work of Saffiotti [5], since it is not as easy to optimize the cost of the computed path in the behavioral plans as it is in our proposal.

With these metarules different behaviors can be activated at the same time but each behavior only makes a contribution if its own context of applicability is true to some degree. For example, an extreme case would be the activation of *Avoid obstacle*, *Follow corridor* and *Follow wall* at the same time. The final output will be the result of combining each behavior output previously modified by the truth value of its context of applicability.

The robot, by means of the activation of its reactive behaviors, will be usually able to accomplish the plan in spite of the presence of unexpected obstacles along its path. Additionally, the executive level monitors the performance of the robot so that possible failures can be detected. These failures may be given by changes in the environment which are not present in the current map. For example a door that was opened, now it is closed or some unexpected obstacle is blocking the path in the middle of a room or the robot becomes trapped among several unexpected obstacles. In these cases the execution of the original plan is interrupted by the execution level and if it is necessary, one behavior specifically designed for these situations is activated to lead the robot to a safe area. The new location of the robot is computed by the execution level and the map is updated to show the new state of the environment. Then the planning level considers the new situation to decide if it is possible to repair the current plan or to generate a new plan or to abort the mission after informing the user.

The robot localization is also a function of the intermediate level and this problem is resolved using an approach based on map matching [25–27] so that the robot can know its position in the world but taking into account a reasonable level of uncertainty.

The lowest level deals with the control of the robot motion, coupling sensors to actuators. The control level is composed of several rule-based basic behaviors which can be combined to generate a more complex observable behavior [28]. Fuzzy logic is also used for designing the rules of the behaviors and to obtain the preferred action from each behavior and then to fuse these actions.

2.1. The Behaviors

The design of the behaviors follows a methodology [29] by the authors, which is based on fuzzy control and fundamentals of regulatory control. This methodology sets a classification of the behaviors according to the use of several abstraction levels on the information. That is, we classify the behaviors according to the kind of input information that is used. Thus, the input data can be:

– Input data from the robot sensors with a simple preprocessing to avoid noisy data. Within this kind of behaviors we distinguish between:

1. Behaviors addressed to reach and maintain an objective such as the following of a wall. We call these *Objective-oriented* behaviors and they are:
   - *Follow wall*. It follows the right or left wall to a certain distance and maintaining the robot aligned with the wall.
   - *Follow Corridor*. It keeps the robot close to the middle of a corridor and in line with it.
   - *Face object*. This behavior moves the robot according to a certain orientation so that the robot aligns itself with the corresponding object, for example wall, corridor or door.
2. Behaviors that tightly couple perception to action such as *Avoid obstacle*. These are *Purely-reactive* behaviors.

– Input data from a sensor-derived world modeling. There is a temporary world representation but only the information necessary for the performance of a specific behavior is represented. For example *Cross door* is a behavior of this kind and we call these *Short-memory* behaviors.

More details about the behaviors, metarules and control level can be seen in [29] whereas details about the planning level and map building can be viewed in [24]. Likewise, overall architecture is profusely detailed in [30]. In the rest of the work the attention is focused on the fuzzy perceptual model, which plays an important role into our system.

3. The Fuzzy Perceptual Model

The proposed perceptual model, allows us to build an approximate environment model by setting various perceptual objects or distinguished places that represent different levels of interpretation of the sensor data. Besides, this fuzzy perceptual model is used to determine the robot's beliefs about the perceptual objects present in the environment and it allows the behavior-based navigation using the deliberative-reactive architecture briefly introduced in Section 2. In order to define the model, firstly the influence of multiple error sources must be taken into account so that a process to reduce the negative influence of these phenomenon has been developed. The idea is to define a sensor model for computing a level of belief about the possible existence of a straight contour around the robot and situated perpendicularly in the direction of the sensor which is sensing that contour. These contours belong to objects and walls of the environment but from the point of view of the robot the contours will be considered coming only from walls so that the concept of wall must be understood in a flexible way. Precedents of this way of understanding the walls can be found in [31] where the integration of qualitative maps and behavior-based robotic systems is demonstrated. Once we have a belief degree of the existence of straight contour that is being perceived perpendicularly to some sensor, then different perceptual objects are defined like *wall*, *corner*, and *corridor*. Others distinguished places like *door* and *hallway* will need the contribution of information stored in a environment map. Besides, if the contour is not enough regular then it is considered as a special kind of object that we call *irregular contour*. These perceptual objects can be used for both building a qualitative map of the environment and navigating using the set of suitable behaviors.

3.1. The Fuzzy Sensor Model

The sensing system of the robot is under the influence of multiple error sources that depend on both environment features and the kind of sensor. These errors concern to the computation of distance and position of the sensing objects by the perceptual system. In our system the main sensors that we use to navigate are sonar and infrared sensors, but actually most measures are provided by the sonar sensors since, infrared sensors have a range limited to approximately 35 centimeters, although on the other hand, infrared sensors have a better angular resolution than sonar sensors. To consider the presence of uncertainty and vagueness in the sensing information, mainly provided by the sonar system, we have follow the process that is below explained. This process has been developed thinking in our robot, a mobile robot Nomad 200 [32], but notice that the underlying ideas can easily apply to any robot that uses ultrasound sensors.

Before we explain the process, some words regarding the error sources that can affect to the sonar system are necessary. The most important errors are summarized in the following aspects:

– The ultrasound signal generated by a sensor can be lost because the incidence angle on the object to detect is too wide so that the echo does not return to the sensor and therefore the object is not sensed.
– Depending on the surface and the form of the object, the echo of the ultrasound beam can present different features and some sensors can detect the echoes of signals that have been generated by other different sensors.
– The ultrasound beam can be affected by environmental conditions as for example, the temperature and the level of humidity of the air which affect to the sound propagation velocity. Likewise, if the distance between the object and the sensor increases then the error in the determination of the distance increases too.
– The ultrasound sensor can sense any object within a cone approximately 30 degrees wide and they can supply erroneous measures depending on the position of the perceived object into the perception cone.

In order to reduce the negative influence of these phenomenon, we have developed the following process. First, our sensing routines take into account the factors below listed:

– Several redundant measures are taken for computing an average value. From this average the clearly noise measure can be eliminated.
– Sensors are activated in a suitable sequence according to its position in the sensor ring so that the possible mutual interference can be avoided.
– The belief in the correctness of the measure that has been sensed by a sensor, will be higher if the echo of the emitted signal is returning of a perpendicular surface to the direction of that sensor. In this case it is well know that the computed measure is very accurate.

Now we focus our attention in the last factor so that a new fuzzy sensor model is developed in order to consider the influence of the uncertainty and vagueness in the measures. The idea of this sensor model is to increase our belief in a measure if the echo is returned from a surface perpendicular to the direction of the sensor. This sensor model makes sense because our architecture can use several behaviors that allow to the robot the following of the contours of objects, in fact *Follow wall* and *Follow corridor* behaviors move the robot following the right or left wall or follow a corridor in the middle and line to its walls. In such situations, the sensor situated in the number 4 position of the sensor ring (see Fig. 2), if the robot is following the wall on the left, or in the number 12 position (see Fig. 2), if it is following the right wall, are going to

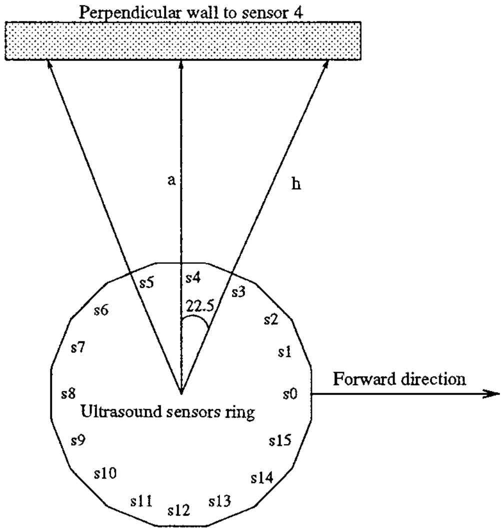

Figure 2. Sensing a perpendicular wall to sensor 4.

be in very good situation for sensing the distance to the object with a high accuracy. However the echoes are not always returned from perpendicular surfaces so that some measure of belief is needed to consider the accuracy of the values computed by any sensor. Thus, we propose the following system.

The first step is to model the situation of sensing a perpendicular echo and afterwards consider the influence of the error source. Figure 2 depicts the ideal situation when the sensor 4 is sensing a perpendicular wall. In this ideal situation, we are supposing the absence of noise in the measures and that the echoes have a good angular resolution. Both features will be more realistic in the case of the echo from sensor number 4 but not for sensors number 3 and 5. This fact will be dealt in a second phase.

If the absence of noise is supposed then the relationship d between the value of $h - a$ and the value of a can be computed by the Eq. (1)

$$d = \frac{h}{a} - 1. \tag{1}$$

Considering the situation shown in Fig. 2, a rectangle triangle exists among a, h and the wall, so that:

$$a = h \cdot \cos 22.5 \tag{2}$$

and replacing the value of a in Eq. (1) then:

$$d = \frac{h}{h \cdot \cos 22.5} - 1; \quad d = \frac{1}{\cos 22.5} - 1; \quad d = 0.0823. \tag{3}$$

Therefore the value of $h - a$ can be said that it is the 8.23% the value of a. So far we have supposed the absence of noise and that the signal echoes return to the sensor describing straight trajectories. This assumption can be considered generally valid for sensor 4, since this sensor is perpendicular to the wall, but for sensor 3 and 5 a more complex model is needed. In these cases, a 30 degree cone is used to model the wide of the range of the sonar, so that the returned echo from these sensors can come from some point of the wall different to the one shown in Fig. 2 and that should be, probably, near of the point that has been sensed by the sensor 4. In order to take into account this aspect, our sensor model considers that the echo can come from a point such that the angle between lines h and a will be a value belonging to the interval [7.5, 22.5] (degrees), since the influence of the half sensing cone is considered (15 degrees). The value of d if the angle is 22.5 degrees, has been already computed and we call it $d_{\max}$, which is considering by percentage as $d_{\max} = 8.23$. Through a similar process for the value of angle 7.5 degrees, we obtain in percentage, $d_{\min} = 0.86$. In order to deal with the vagueness underlying to the assumed suppositions, including the fact of that the sonar measures have an error of 1%, the fuzzy set shown in Fig. 3 is used to get a soft representation for the value of d. The support of this fuzzy set, is defined so that both $d_{\min}$ and $d_{\max}$ will have maximum membership values to the fuzzy set and the values higher than $d_{\max}$ will have a lower membership value gradually. The gradient of the right part of this fuzzy set has been experimentally fixed.

This fuzzy set is considered as the *Approximate Subtraction* (*AS*) between the measures of two consecutive sensors s_i, s_j normalized to the value of s_i, needed to establish whether in front of s_i, a perpendicular wall has been detected by the echo of s_i. Thus, the degree of belief in sensing a perpendicular wall to s_i taking into account s_j is defined as the value of membership function of AS in $(s_j - s_i)/100/s_i$, that is:

$$B(s_i/s_j) = \mu_{AS}((s_j - s_i)/100/s_i)$$

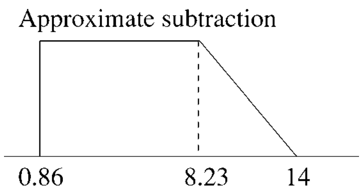

Figure 3. Approximate subtraction between h and a normalized to the value of a.

so that $B(s_i/s_j)$ will be a value that belongs to interval [0,1]. Additionally, the previous sensor in the ring can also be considered in order to carry out a similar process. Thus, let $B(s_1/s_2)$ be the degree of belief of perpendicular wall to sensor s_1 taking into account the value of s_2 and let $B(s_1/s_0)$ be the belief of perpendicular wall to sensor s_1 but considering the measure from sensor s_0, then the final value $B(s_1)$ will be the result of fusing both values. To compute the final value, we consider that a perpendicular wall to s_1 exists if a perpendicular wall to s_1 has been detected depending on the measure of s_2 or s_0. Understanding $B(s_1/s_2)$ as the possibility of first event and $B(s_1/s_0)$ the possibility of second event, then the possibility of the union of both event is the maximum of both possibilities, thus the blended belief will be:

$$B(s_1) = \max\{B(s_1/s_2), B(s_1/s_0)\}. \tag{4}$$

Anyway it is necessary to notice that this process is not avoiding all the possible noise sources since the ultrasound sensors can be affected by several noise sources at the same time and it is usual the presence of non linearity in the data. That is, there may be some situations in which the result of the described operations renders that the belief of a perpendicular wall is 1, but the wall is not actually perpendicular. Therefore to draw more reliable conclusions from this sensor model some redundancy and data fusion are needed. Thus, in the process to determine the perceptual objects several values of this measure along the time must be considered and also the information of different sensors of the sensor ring must be properly blended. Both questions have been taken into account in next subsections in which, both the measures of the sensor model and the distances computed by the ultrasound sensors are understood as variables affected by vagueness and uncertainty. Vagueness refers to the fact that the value of the variable under consideration is only known to belong to some subset of values that is not a singleton. Uncertainty refers to the lack of complete information that precludes a statement as to the certainty that the variable either belongs or does not belong to some subset. In this work, we use the possibility theory [33] for the modeling of information that is both vague and uncertain so that the perceptual information is understood as belief and it is dealt by the robot following the rules of the approximate reasoning [10]. Furthermore, with respect to the design of the behaviors, we use fuzzy control [34] since this kind of control is preferred for non linear systems, systems with no predictable disturbance or low accurate sensors and systems where there is a need to incorporate human experience [35].

3.2. Determining the Perceptual Objects

In this subsection, we are going to show how the sensor model previously defined can be used to determine various perceptual objects or distinguished places that later can be used for both building a qualitative environment map and navigating using the behaviors.

To detect such perceptual objects, the robot has to navigate following the contour of the objects and use the perceptual routines that determine the level of belief of each kind of object. The philosophy that guides the design of these routines is explained bellow.

– First, a linguistic description of the object is given and then this description is represented using expressions with fuzzy sets.
– The fuzzy expressions use certain perceptual features that generate a belief level in the existence of the object.
– If a high level of belief is given during a time then the beginning of the object is considered.
– The perceptual routines are continuously computing the belief level thus if the level becomes lower than a certain threshold during a time then the end of the object is fixed.
– The detection of a different place is also a cause to determine the end of the previous one, although the transition of the old to the new place will be a smooth transition since the levels of belief change gradually.

In next subsection this process is explained in detail for the case of determining the wall existence but when the other objects are explained, the attention will be mainly focused in the fuzzy expressions to determine the perceptual features.

3.3. Determining the Possible Wall Existence

To determine the existence of a wall the information provided by the previously explained sensor model is now used. Through the sensor model a belief value of sensing a perpendicular wall can be associated to each sensor of the robot sensor ring. Summarizing, let s_0 to s_{15} be the sixteen possible positions in the sensor ring then $B(s_i)$ is the belief value, between 0 y 1, of a perpendicular wall exists to the corresponding sensor at

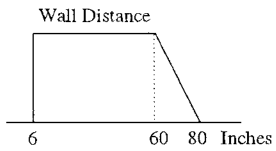

Figure 4. Wall distance.

distance s_i. This information is always available and it is continuously updated. Furthermore, taking into account the direction of the robot motion when it is navigating different kinds of walls can be determined. That is, the process will deal with the possible existence of walls on the right or on the left relative to the robot motion. Therefore our objective is to compute the value of possible left wall, which is expressed by *beliefW*(*Left*) or possible right wall, which is expressed by *beliefW*(*Right*). We are also interested in obtaining the distance to the wall. On the other hand, as the values of big distances can be more affected by noise then a threshold of distance is considered so that a measure is not taken into account if this threshold is reached. Actually this threshold is a new fuzzy set that we call WD (Wall Distance). This fuzzy set, shown in Fig. 4, allow us to smooth the threshold of maximum distance at which a wall can be sensed.

The linguistic description of a right or left wall could be as follows. A left (or right) wall exists if the sensors situated on the left (or right) are detecting an approximately perpendicular wall to the direction of those sensors and also the distance of detected wall is lower than certain threshold. This description is translated to the fuzzy expression *beliefW*(*Left*) which additionally is fusing the information provided by various sensors.

Here we suppose that the robot is following the left wall so that the value of *beliefW*(*Left*) is computed by:

$$beliefW(Left) = \max_i\{\min\{B(s_i), \mu_{\mathrm{WD}}(s_i)\}\} \quad i = 2, 3, 4.$$

That is, the belief of the possible left wall is computed using the value of possible perpendicular wall detected in some of the positions sensed by the sensors number 2, 3 or 4, taking the maximum value of *beliefW* among the three sensors and provided that the distance to the possible wall was under the threshold. Let s_{iMax} be the sensor in which the belief is maximum and being the distance to the wall within the established threshold on the distance, then the value of distance to the left wall is given by Eq. (5).

$$distanceW(Left) = distanceW(s_{iMax}). \tag{5}$$

In this process through the one which, the value of *beliefW*(*Left*) is computed, the sensors 5 and 6 are not taken into account, in spite of the fact that these sensors are also located in the left side of the robot. It is because up to now we are only interested for modeling the sensed environment according to the robot motion to forward since when the robot navigates then it mainly will focus its attention on this kind of information.

Another factor that must be considered to establish the beginning of a wall is the time factor, that is, in order to establish that a left wall is sensed, the value of *beliefW*(*Left*) must be greater or equal than certain threshold continuously during some time.

That is, let $\{beliefW_1(Left), beliefW_2(Left), \ldots, beliefW_n(Left)\}$ be n consecutive measures of *beliefW*(*Left*) and let $\delta \in [0, 1]$ be the considered threshold, then the beginning of the wall is setup if:

$$\forall\, i\{beliefW_i(Left) \geq \delta\} \quad i = 1 \ldots n.$$

Thus we use redundant information to avoid the negative influence of noise sources. Finally, it is necessary to say that the values of parameters n and δ have been determined experimentally once that numerous tests have been carried out.

When the beginning of the wall is detected, the level of belief in that wall is continuously observed and updated so that if this value becomes lower than the threshold during some consecutive times then that wall is given by finished. Other possibility that can provoke for finishing the detected wall is the perception of a new perceptual object in the data, as for example the presence of a corner or a corridor.

While the process of distinguished places detection is accomplished, setting the beginning and the end of every distinguished place, their positions are associated to the points that are sensed. The coordinates of the sensed points are stored and they can be used to build an environment segment map. Thus, we can know the segments that form part of a wall. An interesting consequence of establishing this relationship between the walls sensed from the point of view of the robot with the segments of a geometric map is that, in the topological map a wall has its own position regardless how it is perceived by the robot (as left or right wall). Thus if a wall is classified as a left wall and afterwards the same wall is classified as a right wall according to the side in which the robot has sensed it, both topological walls will be considered as one wall only. On the other hand, as the detection of the walls has been carried out from the point of view of the robot, then one can expect that when the robot navigates it would recognize properly the presence of the wall.

In a similar way, in which the *beliefW*(*Left*) has been obtained, can be determined the belief level of wall on the right, at front and at back using the expressions below described.

$$\textit{beliefW}(\textit{Right}) = \max_i\{\min\{B(s_i), \mu_{\mathrm{WD}}(s_i)\}\} \quad i = 12, 13, 14;$$

$$\textit{beliefW}(\textit{Front}) = \max_i\{\min\{B(s_i), \mu_{\mathrm{WD}}(s_i)\}\} \quad i = 1, 0, 15;$$

$$\textit{beliefW}(\textit{Back}) = \max_i\{\min\{B(s_i), \mu_{\mathrm{WD}}(s_i)\}\} \quad i = 7, 8, 9.$$

Again, the distance to each wall is computed following the same process as explained above.

3.4. *Determining the Possible Corners Existence*

Another possible distinguished place to be considered is a corner, more exactly we consider a corner as the intersection between two approximately perpendicular walls. To detect a corner, the limitations of the ultrasound signal must be considered since the ultrasound signal can be quite affected by noise in this situation. The errors will affect especially to the beam that is pointing to the intersection between both walls, though at least it allows us to know whether the walls are actually joined in some point. Consequently our approach consists of using the information about "possible walls" which can form a corner. That is, the idea that defines the perceptual conditions of a corner is to establish its presence through the determination of two nearby walls that are intersecting and forming a corner. These walls are detected from the information of sensors perpendicular to them. Now we explain this process using an example which is shown in Fig. 5. In this example the robot is trying to detect the corner on the direction pointed by the sensor 2.

The values of "possible perpendicular wall" in the sensors 4 *y* 0 are used to establish the presence of the walls that are intersecting and forming the corner on

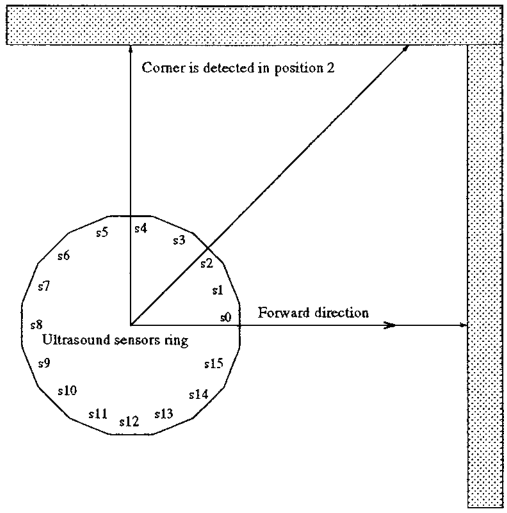

Figure 5. Detecting a corner.

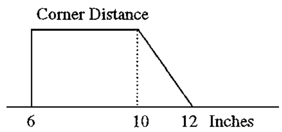

Figure 6. Corner distance.

the direction of sensor 2. Additionally the distance of these walls is bounded using a new fuzzy set in order to detect the corner in the exact moment in which the robot is reaching that corner. This new fuzzy set, that we call *Corner Distance* (CD), is shown in Fig. 6 and it is used to bound the distance to the walls so that they can be considered intersecting and forming a corner.

Then let *beliefC*(s_2) be the belief of existence of a corner in the direction of sensor 2, its value is computed by Eq. (6) which is expressing the possibility of existence of an approximately perpendicular wall to sensor 4 at certain distance and another approximately perpendicular wall to sensor 0 at certain distance too. The information is blended using the minimum operator.

$$\textit{beliefC}(s_2) = \min_i\{\min\{B(s_i), \mu_{\mathrm{CD}}(s_i)\}\} \quad i = 4, 0 \tag{6}$$

Generalizing this equation for the direction of any sensor s_j with $0 \leq j \leq 15$ then we obtain:

$$\textit{beliefC}(s_j) = \min_i\{\min\{B(s_i), \mu_{\mathrm{CD}}(s_i)\}\}$$

with $i = (j + 2)$ *mod* 16, $(j + 14)$ *mod* 16 and where *mod* is the mathematical operator module. Thus, a value of belief in the interval [0,1] is assigned to the possibility of existence of a corner in the direction of sensor s_j. The distance to this corner will be approximated using the minimum value between the distances to each wall.

Once a value of "possible corner" has been assigned to every position of the sensor ring, then these values are blended for computing the belief of "possible left or right corner" according to the direction of the robot motion. Using the maximum as an aggregation operator, the belief of left corner will be:

$$belief\,C(Left) = \max_i \{belief\,C(s_i)\} \quad i = 1, 2, 3, 4$$

similarly for a right corner, it will be:

$$belief\,C(Right) = \max_i \{belief\,C(s_i)\} \qquad i = 12, 13, 14, 15.$$

We notice that in both cases, a sensor located almost in the frontal part of the robot is used. In the case of left corner it is the sensor 1 and in the case of right corner it is the sensor 15. These sensors are used because the robot needs to detect the corners before arriving to them. The process of corner detection needs some redundancy in the measure of $belief\,C(s_i)$ so that in order to establish the presence of the corner, a high level of $belief\,C(s_i)$ is needed during some time using a process similar to the procedure described in 3.3. The end of the corner is determined when the measure of $belief\,C(s_i)$ is under certain limit during some time. The values of the parameters used in this process, as well as the support and membership function of the fuzzy sets have been obtained by experimental way. Regarding convex corners, they are not actually considered into the kind of the discussed corners here but they are considered as result of the end of a wall or a corridor. When it were necessary to associate them to a topological concept they can be modeled using the places that we will call *hallways* and *irregular contour* which are described in Section 3.6.

3.5. *Determining the Existence of Possible Corridors*

To define the corridor we use a linguistic description of the concept of corridor and then the perceptual features are expressed using fuzzy logic and the concept of wall that has already explained. The idea is to try to determine the existence of the corridor detecting "two parallel walls and that are separated to a certain distance". This distance is considered as the width of the corridor. The information of the existence of the walls is taken from $belief\,W(Left)$, $belief\,W(Right)$, $belief\,W(Front)$, $belief\,W(Back)$. Thus, the robot can compute the possibility of existence of two kinds of corridor depending on the direction of the robot motion and the relative position of the walls with respect to the robot. We call $beliefCo(Ahead)$ to the level of belief on a corridor composed by walls that are detected at both sides of the robot and $beliefCo(Front)$ if the walls are detected at front and back of the robot. In any case the process deals with many more intermediate possibilities relative to the position of the robot and the walls since the sensor information is manipulated through fuzzy sets. The width of the corridor *WideCo* will be established by the following equation

$$WideCo(Ahead) = distance\,W(Left) + distance\,W(Right) + 2 \cdot R \qquad (7)$$

where R is the radius of the robot turret.

This width must be within certain limits to be considered as a valid possible width of a corridor. These limits are represented by means of a new fuzzy set that we call *Corridor Width* (CW) which is shown in Fig. 7.

Besides considering the width of the corridor, the level of belief of the parallel walls must be considered. The level of belief of "possible ahead corridor" (Eq. (8)) is defined blending the belief of wall on the left, wall on the right and the belief of these walls being separated within the considered limits. The minimum is the operator used since the three conditions are necessary to determine the existence of the corridor.

$$belief\,Co(Ahead) = \min\{belief\,W(Left), belief\,W(Right), \mu_{CW}(wideCo(Ahead))\}. \qquad (8)$$

This kind of corridor is the most frequently detected when the robot is navigating. However when the robot has to find out its position in the environment, it is

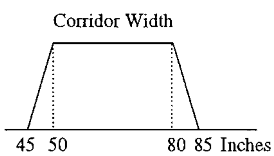

Figure 7. Corridor width.

also useful the detection of the other kind of corridor, that is the "possible frontal corridor" which is formed by a frontal wall and another wall situated at the back of the robot. In this case the width of the corridor is determined by Eq. (9) while the belief of this kind of corridor is defined by Eq. (10).

$$\begin{aligned} &\mathit{wideCo}(\mathit{Front}) \\ &\quad = \mathit{distanceW}(\mathit{Front}) + \mathit{distanceW}(\mathit{Back}) + 2 \cdot R \end{aligned} \tag{9}$$

$$\begin{aligned} &\mathit{belief\,Co}(\mathit{Front}) \\ &\quad = \min\{\mathit{beliefW}(\mathit{Front}), \mathit{beliefW}(\mathit{Back}), \\ &\qquad \mu_{\mathrm{CW}}(\mathit{wideCo}(\mathit{Front}))\}. \end{aligned} \tag{10}$$

Anyway, the detection of the end of the corridor is determined when the robot detects a frontal wall that interrupts the corridor or when it arrives at a hallway situated at the end of the corridor.

3.6. Determining the Existence of Others Distinguished Places

It is possible that there were places with contours not sufficiently regular to be considered as walls and that can not be classified in any of the previous categories. In this case, a special kind of object that we call *irregular contour* is used for modeling these places. For example a convex corner at the end of a wall will be classified as this kind of object.

To finish the environment classification in perceptual objects or distinguished places, we also need to comment on the places that connect some objects with others. These places are the *doors* and the *hallways*. Actually, in the current map building process, we have not developed yet totally autonomous procedures to determine the localization of these places into the environment, therefore they must be located by a human operator. However, once the map has been fully built, the robot will be able to detect the presence of these places using the perceptual features, which are going to be briefly commented, and matching them with the expectation according to the place that it was waiting to find out.

To detect the doors, the information provided by the ultrasound and infrared systems is not enough, thus some previous information has to be supplied to the system. This information is the location of the door within the map of the room. When the robot navigates in the room, it compares the information of the location of the doors with the information from its sensor system in order to determine the existence of the door. The doors are considered as opened doors so that the robot only has to detect the hole at the supposed location. If the hole is not detected in the supposed place then it thinks that the door is closed. The detection of the hole of the door considers the relative position of the robot with respect to the door. That is, if the robot navigates following the contour of a wall and it is expecting to detect a door, for example at the left, then the robot looks for an open space at the left.

On the other hand, if the door to detect is at the front of the robot then the detection process is different. In this last case, the robot tries to determine the location of the door frame using the sensors of the left and right sides that are sensing a minimum distance within a range of certain values. Taking into account the measures of these sensors and the position of the robot, the position of both sides of the door frame is expressed in local coordinate and the width of the door is computed as the distance between both sides of the door frame. The computed width is bounded by a fuzzy set that we call *Door Width*, which is displayed in Fig. 8.

Moreover, the free space between both sides of the door frame and the matching of the door position against the expected position must be checked. If these conditions are fulfilled then a model of the frame door is built in order to generate a free trajectory that crosses the door. The robot has to follow the trajectory to cross the door. This model of the door frame is totally temporary so that if the robot comes too near of the door-frame then the model is again built using new sensor information and a new safe of collision trajectory is generated. The described process in a simulated and in a real environment is shown in Fig. 9. To detect that the door has been finally crossed, the robot monitors continuously the distance to the door frame so that when certain distance has been achieved, it considers that the door has been passed.

Regarding the places *hallways*, the robot detects them using a similar process for detecting a door when this one is located at a left or right side of the direction

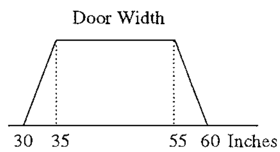

Figure 8. Door width.

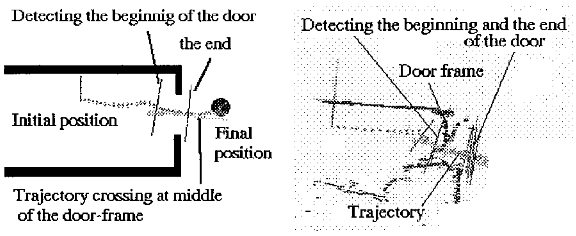

Figure 9. Detecting and crossing a door in a simulated (left) and real (right) environment.

of robot motion. That is, if the robot is following a corridor and it arrives near to an opened door which links the corridor and a room, then it thinks that the corridor is interrupted by the presence of the opened door and classifies that place as a hallway, since it is a place that communicates the corridor with a room. Moreover if the corridor ends on a clear space then the robot will detect the open space in both sides and it will classify this place as a hallway too.

Figure 10 shows both situations that are describing the concept of the hallways for the robot. Besides, this figure shows some convex corners and how these places can be included into the hallway places.

Additionally, we notice that in the case of detection of doors and hallways, the fuzzy logic has also been used in order to establish a level of belief on such distinguished places following a process like the previous ones but specifying other different perceptual features.

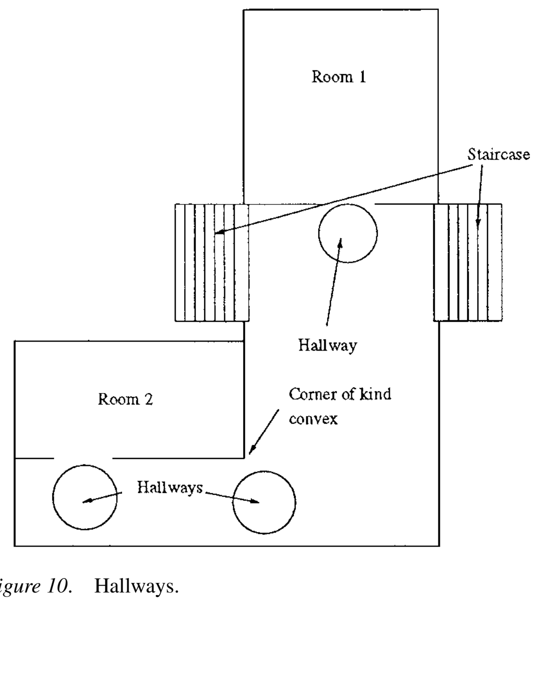

Figure 10. Hallways.

4. Experimental Results

As we said, the objects or distinguished places that have been explained can be used for both building a qualitative map of the environment and navigating using the set of suitable behaviors. Thus, in our architecture one important advantage of using a topological map is the assistance to the navigation since the distinguished places considered show features that can be used by the robot to navigate. For example, if a wall is detected then the robot can activate the *Follow wall* behavior, if a corridor is sensed then the suitable behavior will be *Follow corridor* and so on. This way of generating behaviors is quite robust because the robot can navigate according to the objects that it is sensing. Moreover, the qualitative descriptions of the environment are more suitable to adapt the behavior of the robot to the features of the world. For example, when a door is detected then a specific behavior called *Cross door* will be used. Therefore, as a consequence of the environment description using these distinguished places, a high level plan can be generated and then this plan give us a sequence of places to visit so that the robot can reach the final goal.

With regard to the concrete experiments, we first show an example of sensor data classification in perceptual objects in a real environment using the proposed perceptual model. Next, we show a navigation task and its resolution using a topological map that has been built through the perceptual objects previously explained. First, the navigation task is executed in a simulated environment and then another navigation task is shown but in the real world using a Nomad 200 mobile robot in an office-like environment. Both navigation tasks are accomplished using our own hybrid deliberative-reactive architecture and the proposed perceptual model. It allows the robot to model the environment and navigate according to the context perceived.

4.1. Perceptual Objects Detection in Real World

First experimental example shows the detection of several perceptual objects in a room of an office-like environment of real world. The perceptual objects are each region of room between two lines or flags in Fig. 11. These flags has been marked out by the robot while it was following various kinds of boundaries in the environment. In this room there are chairs, tables and other objects.

The flags are indicating the beginning and the end of each perceptual object which has been labeled

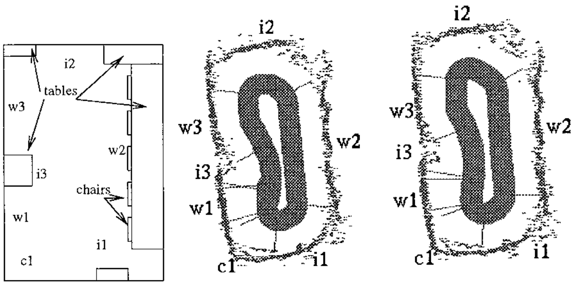

Figure 11. Two trials of perceptual objects detection in the same real world room.

according to the type of object that it is meaning. That is, walls are represented by labels w_i, corners by c_i and irregular contours by i_i. The robot begins its navigation near at wall $w1$ and it follows this wall which is situated on the right of the robot. So that the next object sensed is the corner $c1$, that determines the end of $w1$. The level of belief of corner is high only while the robot is follows the walls that forms that corner and when it moves away $c1$ the belief becomes lower than the threshold and the end of the corner is marked out. The next place is not clearly classified as a concrete perceptual object so that it is defined as an irregular contour called $i1$. In this case the belief in wall has not been large enough to be considered a new wall. The reason is the presence of various objects and the turn of the robot to avoid the collision with that objects. However the concept of wall that we have managed allows the robot to interpreter as wall the row of chairs which is sensed at next moment. This wall is labeled as $w2$. When the belief on wall is lower than the threshold then the end of the $w2$ is determined and a new region begins. The process continues until the robot arrives at original position.

Actually Fig. 11 depicts two different trials in the same room in order to test that the perceptual objects are marked out in approximately the same places. The results show that the locations of each perceptual object are very similar since the flags differ only in some centimeters so that the robot can use the perceptual model for maintain a qualitative representation of the world.

4.2. A Navigation Task that Uses Topological Information

Second experimental example shows a navigation task in a simulated environment. This navigation is accomplished using our own architecture and the perceptual model proposed in this work. Both simulated environment and the result of navigation task are shown in Fig. 12. Before the navigation, a world topological map like the one displayed in Fig. 12 has been built using the explained perceptual objects. In this figure new labels appear since co_i means corridor, d_i means door, h_i means hallway and ec_i is representing the end part of a corridor. The rest of labels upon the arcs only represent links between nodes.

The navigation task consists to achieve the corner labeled as $c15$ from a initial position at corner $c5$. The plan to resolve this navigation task is computed by the planning level using a minimum-cost path search algorithm such as Dijkstra's shortest path algorithm or A*

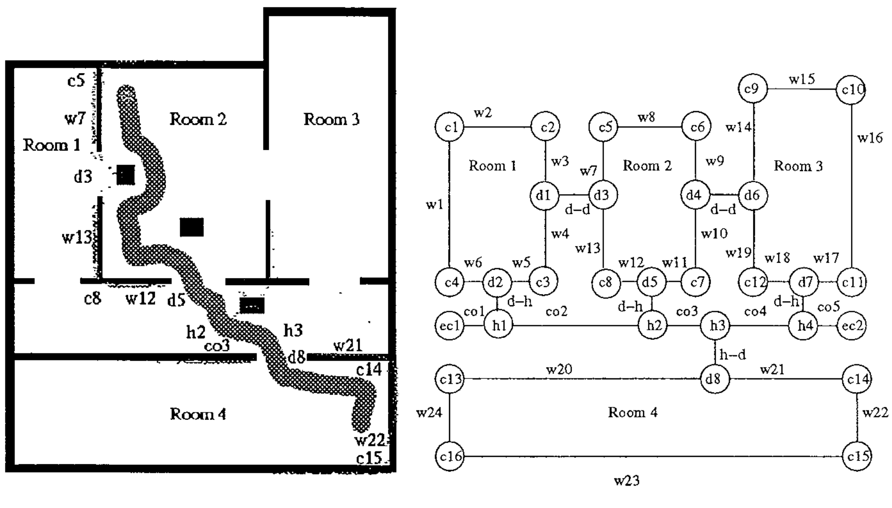

Figure 12. From left to right: A navigation task based on topological information and environment topological map.

algorithm and taking into account an estimated length of each arc of the topological map. The plan that links the initial and final position is composed of the following steps:

$$c5 \to w7 \to d3 \to w13 \to c8 \to w12 \to d5 \to d\text{–}h \to h2 \to co3 \to h3 \to h\text{–}d \to d8 \to w21 \to c14 \to w22 \to c15.$$

In this kind of plan, it is very important the skill of the robot to detect the current context so that the robot can recognize the part of the plan in which it is. Thus, this plan is given to the executive level that uses the perceptual model of the system and information about its localization to set the state of the robot and determine the current perceptual context. According to this context several metarules are activated and therefore the corresponding behaviors are going to control the motion allowing robot to follow the walls, cross the doors, follow the corridor, etc. In fact, the robot achieves successively every goal until it arrives to the desired final goal. In detail, first the robot is situated near to corner $c5$ and wall $w7$, while it is sensing such a objects it has to follow the wall on the right until door $d3$ begins to be detected. At this moment an unexpected obstacle is perceived so that the *Avoid obstacle* behavior is activated and the robot follows the contour of the object until again the beginning of a new object wall is determined. In this case is $w13$ which is ended by the beginning of perception of corner $c8$. After this, the wall $w12$ is determined and ended by the perception of door $d5$. The robot crosses $d5$ using the *Cross door* behavior and then it detects hallway $h2$, where corridor $co3$ begins. In this moment *Follow corridor* behavior is activated and another obstacle is sensed and properly avoided. The navigation successively continues until it finally arrives at corner $c15$. Notice the robot adapts its observable behavior to the perceived context in each case which can be determined using the proposed perceptual model.

4.3. A Navigation Task in the Real World

The results of navigation in simulated environment have been validated through trials in the real world. In this case the navigation task is accomplished in a real environment cluttered with chairs, tables and other objects.

Figure 13 depicts from left to right the real world environment, its representation in a topological map and the result of this navigation task. In this case, the task is to go into the *room*2 from a intial position near the corner c_1 of *room*1. The path computed by the planning level is formed by the next steps:

$$c1 \to w1 \to i1 \to w2 \to d1 \to d\text{–}h \to h1 \to w3 \to h2 \to h\text{–}d \to d2.$$

Again, the plan is supplied to executive level and this level generates the corresponding metarules to activate the behaviors according to the particular context and the defined goals. In this example, the determined perceptual objects have been marked out by flags in the trace of the navigation task of Fig. 13. Thus, the robot which is near to corner $c1$ detects the beginning of wall $w1$ so that it follows the left wall until the belief of $w1$ decreases and the end of this wall is determined. The next object is defined as irregular contour $i1$ until a new object of type wall is detected. It is wall $w2$, the robot

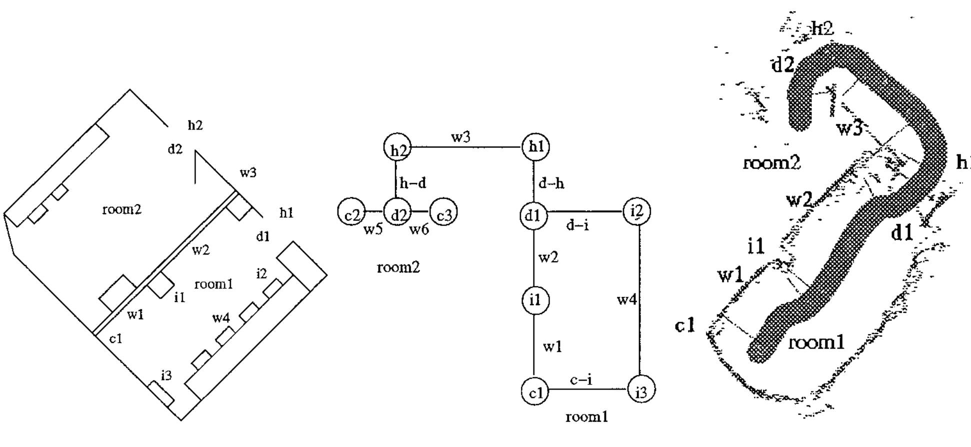

Figure 13. A navigation task in the real world. From left to right: Environment, topological map and trace of the robot motion.

follows this wall until the belief on wall decreases and a new object is expected. The perceptual system determines the presence of an open door that is crossed using the *Cross door* behavior. When the robot goes out the *room*1, the hallway $h1$ is perceived and the robot turns on the left. After this, the wall $w3$ begins to be detected and again the *Follow wall* behavior is activated until the end of the wall is determined by other open door. This open door is defining a new object hallway $h2$ that connects to the door $d2$. The robot turns in order to face the door and then it crosses $d2$ so that finally it goes into *room*2.

5. Conclusions and Future Work

In this work a new *perceptual model* based on fuzzy logic has been proposed to be used in the resolution of two basic requirements for intelligent mobile robot navigation: perception and reasoning. The first one is related to the sensory system that gathers information about the environment. Our perceptual model addresses a process to diminish the negative influence of noise in ultrasound sensor data and a *sensor model* to deal with the uncertainty and vagueness of perceptual information so that the information from different sensors is blended to determine the level of belief about the possible existence of a straight contour around the robot and situated approximately perpendicular to that sensor. Upon this sensor model and following an incremental process, different perceptual objects are defined as *wall*, *corridor* and *corner*, so that various levels of interpretation in the data are managed. To determine the different perceptual objects, a linguistic description is given and then this description is represented using *fuzzy expressions* that define certain perceptual features for generating a belief level in the existence of the object. The determination of other perceptual objects or distinguished places as *doors* or *hallways* are also commented on but in a lesser detail level.

With regard to reasoning, our perceptual model is a valid basis upon which an environment *topological map* can be built so that this map can be used in the deliberative layer to compute a plan that links the initial and the desired final positions. This plan is a high level abstraction plan since it is only addressing the successive objects that the robot must reach while it is navigating and the motion control is under the responsibility of the control layer that can use different behaviors depending on the perceived context. The perceptual model is also used to state the robot's beliefs about the perceptual objects that are present in the environment so that the context of applicability of the appropriate behaviors can be defined and evaluated.

Fuzzy logic is used in different parts of the perceptual model. First, in the sensor model it deals with the uncertainty and vagueness of the ultrasound sensor data and it is used for blending the information from various sensors. After this, to determine the perceptual objects, the linguistic description of them is given by fuzzy expressions that generate the level of belief in the existence of the object. Regarding the navigation architecture, the behaviors are written and combined using techniques based on fuzzy logic too.

Comparing our perceptual model and the world modeling which can be built upon our perceptual model, with others approaches we found interesting advantages. Regard to grid-based approaches [22], such us approaches require a trade off between the level of detail of the grid map and the computational complexity and also the stored information can be dramatically affected by changes in the environment so that the occupancy of a cell could be inaccurate or untimely in dynamic environments. However, using a qualitative model upon the proposed perceptual objects, the world modeling is carried out in a more compact way since this modeling is actually expressing the spatial relationship among the different perceptual objects which form the environment. Thus the number of nodes is usually much lesser than the number of cells of grid-based maps and therefore the planning process will be faster [36]. Regard to other geometrical approaches such as [20], the qualitative models integrated into a deliberative-reactive architecture are more robust to changes in the environment because generally the behaviors are able to achieve their corresponding goals in spite of uncertainty and vagueness in the sensor data and in the representation model [31]. Besides, the proposed perceptual model is based on fuzzy logic and this fact allows the definition in a natural way of the different perceptual objects and obtaining a great flexibility when the sensor data have to be interpreted as for instance *walls*, *corners* or *corridors* since within each one of these concepts is grouping different real situations with their owns particular features (surface, texture, smoothness).

Among the weaknesses of the qualitative models we notice that some places can exist which are not appropriate for a qualitative description or that cannot be described by perceptual features. An example of these situations are the typical wide places near the stairwells and staircases. In second place, in the purely qualitative maps, the robot can even have difficulty distinguishing among the different perceptual object locations, a problem known as sensor aliasing. This problem is resolved in this work associating to each perceptual object some metric information about its location. In third place, the topological paths are not always the shortest paths and sometimes, if a room is very cluttered, perhaps the reactive behaviors could not achieve the current goal without additional assistance because fine motion control is needed to navigate among obstacles and to achieve the goal. These limitations can be overcome using some kind of geometric information of the environment to supply the topological representation. In order to enhance our system we have developed a new version of our architecture to integrate topological and geometric modeling in a hierarchical map that can be viewed in [24].

With respect to perceptual model, experiments in simulation and in the real world have been carried out to test the validity of the model contained into the deliberative-reactive architecture. These results support the utility of the model to achieve a safe and intelligent navigation in office-like environments allowing world modeling, planning and connecting properly perception to action. Regarding future works related to the perception is important the integration of other kind of sensors into our perceptual model. For example the usage of visual information could help to achieve a higher level of autonomy in the process of localization of doors and hallways when the map is being built and also, during the robot navigation it would allow the robot to detect visual landmarks that cannot be perceived using the current perceptual model.

References

1. J. Latombe, *Robot Motion Planning*, Kluver Academic Publishers, 1991.
2. R. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol. RA-2, pp. 14–23, 1986.
3. R. Arkin, "Path planning for a vision-based autonomous robot," in *Proc. of the SPIE Conference on Mobile Robots*, 1986, pp. 240–249.
4. E. Gat, "Reliable goal-directed reactive control of autonomous mobile robots," Ph.D. Thesis, Virginia Polytechnic Institute, 1991.
5. A. Saffiotti, K. Konolige, and E. Ruspini, "A multivalued logic approach to integrating planning and control," *Artificial Intelligence*, vol. 76, pp. 481–526, 1995.
6. H. Hexmoor, D. Kortenkamp, R. Arkin, P. Bonasso, and D. Musliner, *Lessons Learned from Implemented Software Architectures for Physical Agents*, AAAI Spring Symposium Series, 1995.
7. R. Arkin, "Motor schema based navigation for a mobile robot," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1987, pp. 264–271.
8. R. Arkin, "Integrating behavioral, perceptual and world knwledge in reactive navigation," *Robotics and Autonomous Systems*, vol. 6, pp. 105–122, 1990.
9. E. Gat and G. Dorais, "Robot navigation by conditional sequencing," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1994, pp. 1293–1299.
10. L. Zadeh, "The concept of linguistic variable and its applications to approximate reasoning," *Part I Information Sciences*, vol. 8, pp. 199–249; *Part II Information Sciences*, vol. 8, pp. 301–357; *Part III Information Sciences*, vol. 9, pp. 43–80, 1975.
11. A. Bonarini and F. Basso, "Learning to compose fuzzy behaviors for autonomous agents," *International Journal on Approximate Reasoning*, vol. 17, no. 4, pp. 409–432, 1997.
12. S. Goodridge, M. Kay, and R. Luo, "Multi-layered fuzzy behavior fusion for reactive control of an autonomous mobile robot," in *Proc. IEEE Int. Conf. on Fuzzy Systems*, Barcelona, Spain, 1997, pp. 579–584.
13. W. Li, "Fuzzy-logic based reactive behaviour control of an autonomuous mobile system in unknown environments," *Engineering Applications of Artificial Intelligence*, vol. 7, no. 5, pp. 521–531, 1994.
14. F. Michaud, "Selecting behaviors using fuzzy logic," in *Proc. IEEE Int. Conf. on Fuzzy Systems*, Barcelona, Spain, 1997, pp. 585–592.
15. M. Sugeno and M. Nishida, "Fuzzy control of model car," *Fuzzy Sets and Systems*, vol. 16, pp. 103–113, 1985.
16. T. Takeuchi, Y. Nagai, and N. Enomoto, "Fuzzy control of a mobile robot for obstacle avoidance," *Information Sciences*, vol. 43, pp. 231–248, 1988.
17. A. Saffiotti, E. Ruspini, and K. Konolige, "Blending reactivity and goal-directedness in a fuzzy controller," in *Proc. of the IEEE Int. Conf. on Fuzzy Systems*, San Francisco, California, 1993, pp. 134–139.
18. S. Goodridge and R. Luo, "Fuzzy behavior fusion for reactive control of an autonomous mobile robot: MARGE," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, 1994, pp. 1622–1627.
19. W. Kim, J. Ko, and M. Chung, "Uncertain robot environment modeling using fuzzy numbers," *Fuzzy Sets and Systems*, vol. 61, pp. 53–62, 1994.
20. J. Gasós and A. Martín, "A fuzzy approach to build sonar maps for mobile robots," *Computers in Industry*, vol. 32, pp. 151–167, 1996.
21. H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1985, pp. 116–121.
22. G. Oriolo, G. Ulivi, and M. Vendittelli, "Real-time map building and navigation for autonomous robots in unknown environments," *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, vol. 28, no. 3, pp. 316–333, 1998.
23. R. Arkin, *Behavior-Based Robotics*, The MIT Press, 1998.
24. E. Aguirre and A. González, "Integrating fuzzy topological maps and fuzzy geometric maps for behavioral-based robots,"

International Journal of Intelligent Systems, vol. 17, no. 3, pp. 333–368, 2002.
25. R. Bauer and W. Rencken, "Sonar feature based exploration," in *Proc. of the IEEE/RSI Int. Conf. on Intelligent Robots and Systems*, 1995, pp. 148–153.
26. J. Crowley and P. Reignier, "World modeling and position estimation for a mobile robot using ultrasonic ranging," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1989, pp. 674–680.
27. W. Rencken, "Concurrent localization and map building for mobile robots using ultrasonic sensors," in *Proc. of the IEEE/RSI Int. Conf. on Intelligent Robotics and Systems*, 1993, pp. 2192–2197.
28. E. Aguirre, M. García-Alegre, and A. González, "A fuzzy safe follow wall behavior fusing simpler fuzzy behaviors," in *Proc. of the Third IFAC Symposium on Intelligent Autonomous Vehicles*, Madrid, Spain, 1998, pp. 607–612.
29. E. Aguirre and A. González, "Fuzzy behaviors for mobile robot navigation: Design, coordination and fusion," *International Journal of Approximate Reasoning*, vol. 25, pp. 255–289, 2000.
30. E. Aguirre, "A behavior-based architecture for mobile robot navigation," Ph.D. Thesis, Department of Computer Science and Artificial Intelligence, Granada, Spain, 2000. (In spanish).
31. M. Mataric, "Integration of representation into goal-driven behavior-based robots," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 304–312, 1992.
32. N. Technologies, "User's manual," 1995.
33. L. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets and Systems*, vol. 1, pp. 3–28, 1978.
34. D. Driankov, H. Hellerdoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*, Kluwer Academic Publishers, 1993.
35. R. Yager and D.P. Filev, *Essentials of Fuzzy Modeling and Control*, John Wiley & Sons, Inc., 1994.
36. S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.