# A multi-agent system based on active vision and ultrasounds applied to fuzzy behavior based navigation *

Eugenio Aguirre, Miguel García-Silvente, Moisés Gómez, Rafael Muñoz, Carmen Ruiz
Department of Computer Science and Artificial Intelligence
E.T.S. Ingeniería Informática. University of Granada (Spain).
18071 - Granada. Phone: +34.958.240807 Fax: +34.958.243317
e-mail: {eaguirre,mgs}@decsai.ugr.es, moi_gomez@hotmail.com, salinas@decsai.ugr.es

**Abstract**

In this paper we present a multi-agent system that uses both visual and range sensors information to achieve a safe and efficient behavior-based navigation. The system uses a topological map of an indoor office-like environment and it is based on fuzzy behaviors, providing to the robot the ability to find doors in rooms. The system is formed by distributed agents that can establish communications among themselves. We use both reactive and deliberative agents and we have carried out a modular design of the system to facilitate its posterior expansion by adding new skills or new agents. Also, the use of a multi-agent system allows us to achieve a more robust performance of the robot. Regarding the behaviors, they have been designed using fuzzy rules to set the appropriate relationship between the input data and the control values to apply to the robot actuators. The system has been implemented in a Nomad 200 mobile robot and has been validated in numerous tests in a real office-like environment.

## 1 Introduction

In the mobile robot navigation area, the approaches that use topological models of the environment [6], are very popular because they have the advantage of manipulating information on a higher level of abstraction than those based on geometric representations [4]. Relating to behavior based navigation [3], in the approaches that use topological representations, the meaning associated to nodes and arcs of the topological map is not always the same. In some cases [7] the nodes represent distinguished places associated to the behaviors, while in other works [2] the arcs also indicate the behavior needed to navigate between nodes. The navigation between two points is seen as the execution of a plan composed by a set of nodes joined by arcs or behaviors to execute, where the nodes are sub-goals to achieve the desired plan. The abilities of the robot, like *follow door*, *follow corridor*, *cross door* or *avoid obstacle*, are essential to achieve the generated plan since the robot has to trust in these abilities to reach each sub-goal established in the plan.

In this work we present a multi-agent system that incorporates visual information to a navigation system based on fuzzy behaviors, adding the new ability to find doors in the rooms and leading the robot to them, either to enter or leave the room. This new ability arises from the interaction of several behaviors that get the input information from ultrasound sensors with others that only use information from the visual system. All of them have been designed using fuzzy rules to manipulate the imprecision and vagueness of the sensor data. The visual information is extracted from images captured by a camera placed on a pan-tilt unit (PTU) that allows to move the camera independently of the robot movement. For this reason, vision has an active role in this system. To ease the door detection, an artificial mark with numerical digits inside it has been used. It is recognized by the robot after a learning process using neural networks. Once the robot detects the mark in the door, it must keep watching the mark indicating the correct direction. The robot can move towards the door avoiding obstacles, reach the door, cross it and continue its way in the environment. We

*This work has been supported by the MCYT under Project TIC2003-04900

assume that all doors are open because our robot cannot open them and that a topological map of the environment is supplied to the robot.

The next section deals with an overall description of the multi-agent system architecture. Afterwards we focus our attention in the Fixation agent. This agent carries out the visual processing and camera control. In Section 4 some experimental results in a real office-like environment are shown. Finally we conclude with some conclusions and future work.

## 2 Multi-agent system architecture

The goal of the multi-agent system is to perform the navigation in an office-like environment that is modelled by means of a topological map (as shown in Figure 1), in which there are a set of rooms connected by doors and corridors. In this version of the work the map is built by an human operator but in further versions the robot will be able to build the topological map carrying out an exploration of the environment. Also, we have placed an artificial mark besides the doors to ease the detection by the vision system. The mark can be placed in the right-side or in the left-side of the door, always related to the direction of advance to cross it. This information also appears in the figure 1 as *Right* (right-side) and *Left* (left-side) while the doors are represented as $D_i$.

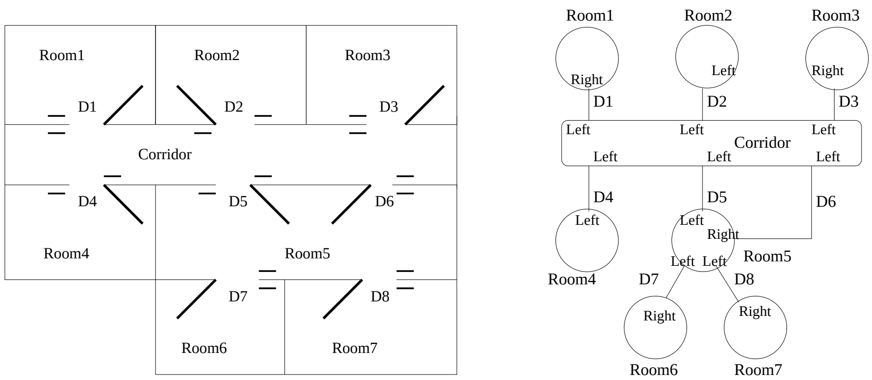


Figure 1: Office-like environment and its topological map

At this point we define an *agent* as a software process aimed to get or keep a goal implicit in its own design. We have designed a distributed system based on agents with both reactive and deliberative agents. The system as been designed to be modular and to facilitate its posterior expansion, either adding new abilities to an agent or adding new agents. The set of agents is the following:

- Fixation: it is in charge of processing the images to detect the desired mark in the environment, and to move the PTU to fixate it, allowing the robot to move towards the door.

- Navigation: this agent has several fuzzy behaviors that allows the robot to navigate in the environment safely, avoid obstacles, approach to a mark, cross a door and to navigate in a corridor among others abilities.

- Planner: Its aim is to find a way to an end position from the current position. It uses a topological map and a path finder algorithm based on minimal paths using the weights associated to the arcs of the graph. It is also in charge of the execution and monitoring of the plan, activating the rest of the agents to find the mark in the desired door, approaching to it, crossing it and navigating in the environment.

- Nomad200: This agent establishes a communication via TCP/IP with a daemon running in the robot. This daemon is in charge of getting external petitions and is supplied by the manufacturer of the Nomad 200 robot. Any agent that desires to send a command to the robot has to do it through this agent that will redirect the command in an appropriate format to the daemon. This agent allows a concurrent use of the robot queuing all petitions. The commands that can be sent allow to configure the speed and acceleration of the robot, to obtain the data from the ultrasound and infrared sensors and to move it.

- PTU: This agent allows the use of the PTU to the rest of the agent community. The main advantage of using this agent is the possibility of queuing the PTU movements in a remote way, besides controlling concurrent use of PTU. The agent gives the service of moving the PTU and to inform about the exact position of both axes (pan and tilt) by message-passing.

In figure 2 is shown the multi agent system. The lines that join agents indicate that there is some kind of communication to accomplish some task.

In this paper we only describe the Fixation agent and show experimental results of the performance of the multi-agent system. Details on the design of the fuzzy behaviors can be seen in [1].

## 3 Fixation Agent

*Fixation* agent is in charge of detecting and fixating the desired mark. The agent can develop the following tasks: a) to detect the total number of marks in the image captured in the current robot position; b) to inform if a certain mark is present in the image captured in the current robot position; c) to fixate a mark if it is in the image captured in the current robot position; d) to inform about the distance between the camera and the mark. With these capabilities the agent can detect the set of marks in a captured image and inform if a certain mark is in it. When the desired mark needed to accomplish the current part of the plan is identified the agent can fixate it using the PTU to make the robot go towards it. Also, the agent can measure the distance between the mark and the camera to decide if the robot is close enough to the door to start the appropriate behavior to cross it.

The work of the agent can be structured in two parts: mark detection and mark fixation.

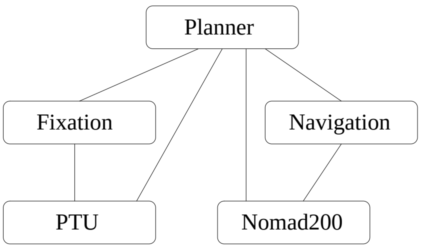


Figure 2: Multi-agent system architecture.

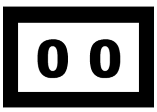


Figure 3: Artificial mark.

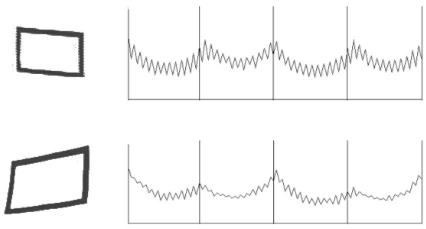

Figure 4: Examples of the double signature applied to a rectangular border.

### 3.1 Mark detection

The mark (see Fig.3) is composed by some numbers with an external black rectangle, in order to facilitate the detection task. The numbers are used to identify the door. It will help the decision system to know which door leads to each room in the topological map. The marks have been designed using a simple text editor and it fits in an A3 sized paper.

The work of detecting the mark can be divided into two different parts: first detecting the mark in the image and second determining what number is inside the mark. To detect the mark the agent captures a grey scale image to process. The first step is to binarize the image. To do this, we have used a thresholding algorithm proposed by Otsu [8] based on the histogram. This method selects a thresholding value based on the illumination that minimizes the entropy in the result image and so, it makes more robust the detecting process.

The following step is detecting the external border of the mark. The first attempt was to use the *signature* [5]. The signature of a object is a function $f(\phi)$ that indicates the distance from the central point to the most external point of the object that is in a line traced with a certain angle $\phi$. Being the angle $\phi$ in the range $[0, 2\pi]$.

However, signature does not allow to detect holes in objects and is not good enough to represent the mark from different points of view. Instead of the signature, we have developed a variant of the method that has been called *double signature*. It consists in taking the distance to the more external, as well as to the nearest point to the centre in a certain angle $\phi$ in the range $[0, 2\pi]$. Figure 4 shows the double signature applied to the rectangular black border from different points of view. This method brings two main advantages. First it has been proved to be more robust in the detection of the marks using several classification methods, and second it allows the description of holes in the objects that it represents, which is impossible using the *signature* descriptor.

Those distances are stored in an array to represent the object that appears in the image after thresholding. We have used a neural network with one hidden layer to learn the double signature of the rectangular black border using the *backpropagation* algorithm with momentum.

Once the mark has been found it is necessary to determinate the number that is inside. To do this we make an image with every digit and extract the following features:

1. *Horizontal and vertical laplacian*. The method consists in creating a function with the number of pixels occupied by the digit in every row and column, then obtaining the laplacian of this function and escalating it to the range $[-1, 1]$.

2. *Block Sum*. Due to the previous method has not give us good results to detect a digit from different points of view, it has been necessary to create another method for extracting features. We have made 2x2 blocks and have summed the number of pixels in each block. Finally we translated the sum to the range $[0, 1]$.

Using both methods we obtain a set of features that have been learned using a neural network. Joining the digits inside the mark, we calculate the number.

### 3.2 Mark Fixation

Once the mark has been located, it is necessary to fixate it. The objective is to have the mark always into the next image and to indicate to the robot the appropriate direction to the door. We have used the PTU to centre the mark in the image. The only information needed to centre the mark is the pixel distance of the mark's centre to the centre of the image. Then it is necessary to establish the correspondence that leads the PTU to center the mark in the image.

To determinate the PTU angle increment needed to centre the mark we have made two regression lines for both pan and tilt axis based on the distance explained above. The lines that form our model are:

$$pan\_degrees = -25.99 \cdot \frac{2IncPixX}{Width} + 0.05; \quad tilt\_degrees = 17.63 \cdot \frac{2IncPixY}{Height} + 0.29$$

Where $IncPixX$, $IncPixY$ are the pixels distance from the center of the mark to the image center in both axis $X, Y$. $Width$, $Height$ are the width and height of the image respectively.

## 4 Experimental Results

This multi-agent system has been implemented and tested using a mobile robot Nomad 200 updated with a vision system and placed in a real office-like environment represented in Figure 5. We will try to explain the concurrent running of the whole system with a real example. The robot is initially situated in *room 1* and has to get *room 3*, travelling through a corridor. With that information the Planner agent calculates the following path according to the topological map: Room 1 → D00 → Corridor → D02 → Room 3. These steps represent sub-goals to accomplish in order to achieve the main goal (reach room 3). We want to emphasize at this point that the plan is a high level of abstraction plan where no trajectory is specified to the robot, but rather a set of points to reach. The plan is executed in the stages explained below.

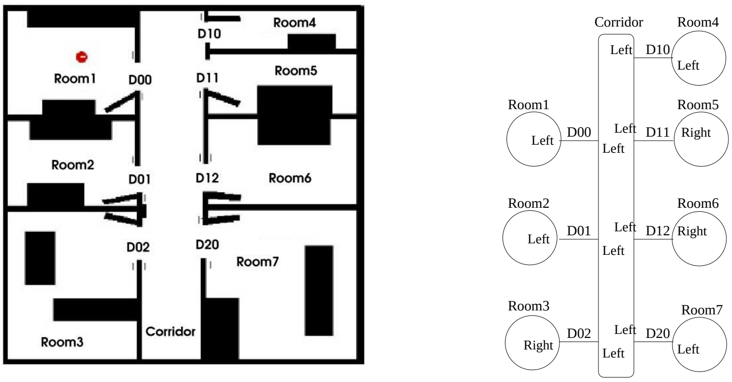


Figure 5: Rooms map and topological map of the real environment

1. *Search the door 00 and approach to it*: The robot searches for the mark that identifies the door 00 by means of a static search. When the robot finds the mark 00, it will approach it. To get to the mark, the robot tries to orientate the wheels and the turret in the direction that indicates the PTU lined up with the camera. It is shown in Figure 6, in which are also shown several images taken by the camera during the time the robot approaches the mark.

2. *Cross the door number 00*: When the robot is placed near the door, it orientates itself towards the door hole using the information obtained from the sonar about the door frames. Once the robot is placed properly, the behavior *Cross the door* that is in charge of such function is activated. The path of the robot and some images taken while the robot crosses the door is shown in Figure 7.

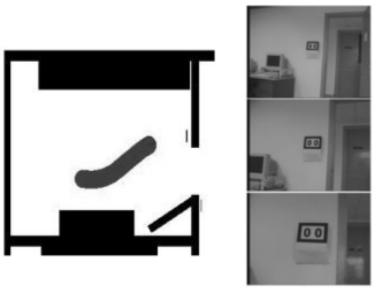


Figure 6: Approaching to mark 00.

Figure 7: Crossing door 00.

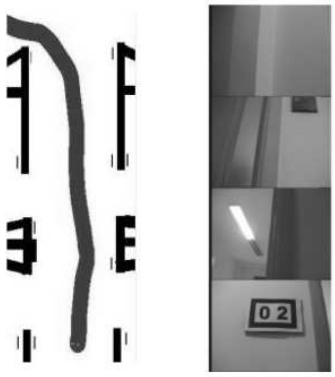


Figure 8: Going along corridor.

3. *Go across the corridor looking for the destination room*: When the door is crossed the robot is in the corridor and must go along lined up in the centre to find the destination room. Figure

8 shows how the robot search and finds the mark of the door 02 after navigating along the corridor. When the robot detects the mark 02, it consults the map to know the relative position of the door to the mark. Helped by this information and by ultrasound sensors, it centres itself in relation to the door hole and cross it.

## 5 Conclusions and future work

In this work we have presented a multi-agent system that employs both visual information and information provided by range sensors to incorporate a set of abilities: a) detecting the doors in rooms by means of an artificial mark, b) lead the robot towards a door and c) crossing a door. The environment is represented using a topological map, where nodes represent either rooms or corridors and arcs represent doors. In each door is situated an artificial mark formed by a rectangular border and interior digits. The mark is learned using a neuronal net that has been trained using several images from different points of views. The mark is identified and used to help the robot to approach the door specified in the plan. The Fixation agent is capable of maintaining the camera fixated on the mark using the PTU, whereas the navigation agent counts with fuzzy behaviors that allow it to guide the robot towards the mark of the door, to avoid the obstacles, to follow the walls, to cross doors and move the robot along a corridor. These behaviors have been designed using fuzzy rules that allow us to manage the underlying vagueness and uncertainty in the data of sensors in an appropriate way. Finally, the Planner agent generates the plan and gives orders to the rest of the agents to execute the plan monitoring the operation. The experimental results carried out in an real office-like environment show that the robot is able to identify and fixate the indicated mark in real time in spite of the movement of the robot; it is able to approach it avoiding possible obstacles detected using ultrasound sensors and makes new searches of the mark when it is lost. In the future we intend the robot to be able to detect the doors without the help of the artificial mark and also we want to include in the topological map the content of rooms in detail using other perception models developed previously by the team.

## References


[1] E. Aguirre and A. González. Fuzzy behaviors for mobile robot navigation: Design, coordination and fusion. *International Journal of Approximate Reasoning*, 25:255–289, 2000.

[2] E. Aguirre and A. González. Integrating fuzzy topological maps and fuzzy geometric maps for behavior-based robots. *International Journal of Intelligent Systems*, 17(3):333–368, 2002.

[3] R.C. Arkin. *Behavior-Based Robotics*. The MIT Press, 1998.

[4] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer Magazine, Special Issue on Autonomous Intelligent Machines*, 22(6):46–57, 1989.

[5] R. González. *Tratamiento Digital de Imágenes*. Addison-Wesley Iberoamericana, SA, 1996.

[6] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proc. of the Twelfth National Conf. on AI (AAAI-94)*, pages 979–984, Menlo Park, Calif., 1994.

[7] M. Mataric. Integration of representation into goal-driven behavior-based robots. *IEEE Transactions on Robotics and Automation*, 8(3):304–312, 1992.

[8] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions On-Systems, Man, and Cybernetics*, SMC-9(1):66–69, 1979.