Chapter 7

Conclusions and Future Work

7.1 Revisiting the Objectives

The need for autonomous robots has been rapidly increasing in the last years. There are many areas in which these robots are used, ranging from "service robots", such as museum guides or transportation robots in factories, to robots used for tasks to be performed in inaccessible environments, such as planetary exploration, hazardous material handling and rescue missions.

Usually, service robots operate in indoor structured environments. The problem of navigating through indoor environments has been the focus of robotics research during many years, and many successful results have been achieved. Usually, the map of the environment is given a priori (either a detailed metric map or a topological one, showing the spatial relationship among different places of the environment), or, when it is not given, there is an initial phase for learning the map. Once it is learned, the robot repeatedly performs the task in this environment. Examples of such robots are those performing delivery tasks in office environments or guiding tours in museums [67, 9].

On the other hand, inaccessible environments are usually unknown and unstructured (as is the case in most outdoor environments), which pose a more difficult problem. The lack of structure of such environments makes the map building very difficult. Moreover, the large scale of these environments also adds to the difficulty of mapping and navigation tasks. These characteristics make it impossible to apply the approaches used in indoor structured environments. Although there has been also a lot of research on navigation in unstructured environments, it is still an open problem.

This PhD thesis has focused on this latter problem, that is, on ***navigating in unknown unstructured environments***. The research was part of a robotics project whose goal is to have a completely autonomous robot capable of navigating in outdoor unknown environments. A human operator selects a target using the visual information received from the robot's camera, and the robot has to reach it without any further intervention of the operator. Navigating to a target is a fundamental task of any mobile robot, whatever its mission is (be it grasping objects, analyzing them, looking for something, etc.) The task to be performed once the target has been reached is outside the scope of the project and this thesis.

A first milestone of the project was to develop a navigation system for indoor unknown unstructured environments. The reason for starting with indoor environments was that the development of robust vision systems for outdoor environments is still an open and very difficult problem in the field of computer vision. Therefore, since the vision system was not the focus of our research, we decided to start experimenting indoors, for which vision systems are much easier to develop. Moreover, we designed the landmarks so that we could easily change their location, thus, permitting us to configure scenarios of different complexity.

This thesis has reported the research carried out in order to accomplish this first milestone. For achieving it, we have combined *landmark-based navigation*, *fuzzy* distance and angle representations and *multiagent coordination* based on a *bidding mechanism*. The objective of our research was to have a ***robust navigation system with orientation sense for unknown unstructured environments using visual information***.

7.2 Contributions

The research has been focused on two main threads: the ***control architecture*** and the ***mapping and navigation method***. The contributions of the thesis on these two areas are presented next.

Regarding the ***control architecture***, we have proposed a general coordination architecture based on a *bidding* mechanism. In this architecture there are two types of systems: *executive systems* and *deliberative systems*. Executive systems have access to the sensors and actuators of the robot. These systems offer services for using the actuators to the rest of the systems (either executive or deliberative) and also provide information gathered from the sensors. On the other hand, deliberative systems take higher-level decisions and require the services offered by the executive systems in order to carry out the task assigned to the robot. Although we differentiate between these two types of systems, the architecture is not hierarchical, and coordination is made at a single level involving all the systems. This coordination is based on a simple mechanism: *bidding*. Deliberative systems always bid for the services offered by executive systems, since this is the only way to have their decisions executed. Executive systems that only offer services do not bid. However, those executive systems that require services from any executive system (including themselves) must also bid for them. The systems bid according to the internal expected utility associated to the provisioning of the services. A coordinator receives these bids and decides which service each of the executive systems has to perform.

The bidding mechanism assures that the action actually being executed by the robot is the most valued one at each point in time, and thus, if the systems bid rationally, the dynamics of the bids lead the robot to execute the necessary actions in order to reach a given target. An advantage of using such mechanism is that there is no need to create a hierarchy, such as in the subsumption architecture, but it is dynamically changing depending on the specific situation of the robot and the characteristics of the environment. A second advantage is that its modular view conforms an extensible architecture. To extend this architecture with a new capability we would just have to plug in a new system. Moreover, the coordination mechanism can be applied at different levels of the architecture, be it at the overall architecture level, or within each one of the systems.

For our specific navigation problem, we have instantiated this architecture with three systems: the Pilot, Vision and Navigation systems. The first two being executive systems, and the latter one being deliberative. The Navigation system has been designed as a multiagent system using the same bidding coordination mechanism used in the overall architecture. The high-level task of navigating to a given target has been decomposed into a set of simpler tasks, and we have designed one agent competent in each of these tasks. These agents compete, since they may request the execution of conflicting actions. As in the overall architecture, each agent bids for the services offered by the executive systems, and there is a coordinator agent that decides which is the most urgent request. This request is then sent as the request of the Navigation system, which will have to compete with the requests of the Pilot system.

Regarding the ***mapping and navigation method***, we have addressed two problems: the problem of providing the robot with orientation sense and the problem of building a map of the environment and using it for navigational purposes. Concerning the orientation sense, we have built upon previous work presented by Prescott [55], which describes a model for storing spatial relationships among landmarks in the environment. We have extended Prescott's model so that it can be used with fuzzy information about the locations of landmarks. This is of great importance when working with real robots, as it is impossible to avoid dealing with the imprecision of real world environments. As far as we know, this is the first application of Prescott's model on a real robotic system. As part of this extension, we have also developed methods for building a topological map of the environment, which is used for computing diverting targets, needed by the robot when it finds that the path to the target is blocked.

Although the robotic system proposed in this thesis has been presented as a whole system, including both the control architecture and the mapping method, they are two solutions for two completely independent problems. Thus, we could substitute Prescott's mapping method by any other mapping method (be it another topological approach, a metric approach, etc.). Obviously, the particularities of each system depend on the mapping method (e.g. it would make no sense having a Vision system if the map uses sonar readings), but the overall architecture and its coordination mechanism would not be affected at all by the choice of this mapping method. Similarly, our mapping method could be used in a robotic system controlled by any other architecture (be it hybrid, centralized, etc.).

We have obtained successful results, both on simulation and on real experimentation, showing that the mapping method is capable of building a map of an unknown environment and using this information to move the robot from a starting point to a given target. The experimentation also showed that the bidding mechanism we designed for controlling the robot produces the overall behavior of executing the proper action at each moment in order to reach the target. Thus, we consider that we have satisfactorily achieved the objective of developing a navigation system with orientation sense for unknown unstructured environments.

In parallel with the experimentation with the real robot, we have also used simulation to apply Machine Learning techniques. More concretely, we have used Reinforcement Learning for having the system learn how to use the camera more appropriately, that is, to use it only when needed. We have also used a Genetic Algorithm approach, in order to tune some of the parameters that define the behavior of the agents in the Navigation system. Successful results have been obtained with both techniques, though there is still much work to do. Actually, they could easily be the subject of several PhD theses, especially the work on Reinforcement Learning.

7.3 Future Work

Although, as we have just said, we consider that the goal of the thesis has been accomplished, there are plenty of improvements that could be done in order to achieve better results. In the following sections we present, for each of the aspects of the research carried out in this thesis, some of the open issues that deserve further research (some of which we are already working on). Note that it is basically a compilation of the Future Work sections of each of the previous chapters.

7.3.1 Mapping and Navigation

The extension of Prescott's method, together with the algorithms to compute diverting targets, has been shown to successfully encode the environment into a map that permits navigating from a starting point to the target. However, we would like to explore other mapping methods, so that the combination of the different methods adds robustness to the Navigation system. With the current mapping method, the robot needs to see at least three landmarks in order to be able to use the information stored in the map. We would like to develop some other mapping methods to cope with the situations in which the robot has very little information (i.e. less than three landmarks). These methods would be even more qualitative than our fuzzy extension of Prescott's method. We could, for example, look at the field of Spatial Cognition, which works with spatial relationships such as "landmark X is at the left hand side of the line connecting landmark Y and landmark Z".

7.3.2 Robot Architecture and Multiagent Navigation System

One of the first things to explore in our coordination architecture is the use of a more economic view of the bidding mechanism. With this approach, each system (or agent) would be assigned a limited credit, and they would only be allowed to bid if they had enough credit. There should also be a way to reward the systems (agents). If not, they would run out of credit after some time and no one would be able to bid. The difficulty of the reward mechanism is how to decide when to give a reward and who deserves to receive it. This problem, known as the credit assignment problem, is very common in multiagent learning systems, especially in Reinforcement Learning, and there is not a general solution for it; each system uses an ad hoc solution for the task being learned.

An alternative to the economic view would be to have a mechanism to evaluate the bidding of each system (agent), assigning them succeeding or failing bids, or some measure of trust, in order to take or not take into account their opinions. However, we would face again the credit assignment problem.

Regarding the specific set of agents we have designed for solving the navigation problem, we could introduce some improvements on some of them, and even add new agents to the Navigation system. Some of these improvements could go in the following lines:

- *Target Tracker*: this agent could do some more intelligent tilt angle selection, being a function of the distance to the target, thus, increasing the chances of having it in the view field of the camera.

- *Risk Manager*: this agent could also bid not only for looking ahead or around, but also to specific areas with fewer landmarks, or even selecting a random direction to look to. Right now, if there are very few landmarks ahead, this agent sticks bidding for looking ahead, and never bids for looking around, thus, ignoring a large part of the environment. An alternative to modifying the *Risk Manager* would be to add a new agent with this behavior.

Some improvements could also be done on the Pilot and Vision systems. Regarding the Pilot, we could use a better obstacle avoidance algorithm. With the current algorithm, only the closest obstacle is considered for computing the avoidance path. We could improve the robot's performance if the Pilot took into account all the obstacles and landmarks stored in the Visual Memory, thus, producing better avoidance paths We are also planning to equip the robot with a laser scanner. This laser would be continuously scanning a 180 degree area in front of the robot to accurately detect obstacles that are several meters away. With this new sensor, the Pilot could avoid the obstacles before bumping into them, thus, generating better paths. Regarding the Vision system, we plan several improvements. The first one is to finish the stereo algorithm, so we can use the two available cameras. Another very important improvement is to make the Vision system more robust, so that it does not need to check the recognized landmarks against the Visual Memory. Actually, we should use the robust Vision system to adjust the imprecisions of the Visual Memory. We also plan to convert the Vision system into a Multiagent Vision system. In this system, several agents would process the camera images with different algorithms, and the agents should agree on what could be a good landmark (salient enough, robust, static, etc.). A final improvement of the Vision system would be to let it bid for services by other systems (either the Pilot system or itself). With the bidding capability, it could request the Pilot to approach a landmark to better recognize it, or even "request itself" to slightly move the camera so that a partially seen landmark enters completely the view field.

7.3.3 Reinforcement Learning

Although the results obtained through Reinforcement Learning showed that the system learned to select actions in order to solve the complex camera tradeoff, we still need to integrate it into the overall multi-agent system, to see if the performance of the whole system is also improved. Even though the *Learning Agent* knows which actions it has to bid for (following the learned policy), it is not clear what its bidding function should be; it could be a constant bidding value, or a bidding depending on the values of $V(s)$.

Some more further work will be focused on the design of the state and feature representation and the set of available actions. Asada et al. [5] proposed a solution for coping with the "state-action deviation problem", in which actions operate at a finer grain than the features can represent, having the effect that most actions appear to leave the state unchanged, and learning becomes impossible. We plan to evaluate the suitability of this approach in our experiments. Regarding the action set design, we found that the set of available actions was maybe too small and some more actions may be needed. We are working on an "action refinement" method [20] that exploits prior knowledge information about the similarity of actions to speed up the learning process. In this approach, the set of available actions is larger, but in order to not slow down the learning process, the actions are grouped into subsets of similar actions. Early in the learning process, the Reinforcement Learning algorithm treats each subset of similar actions as a single "abstract" action, estimating $P(s'|s, a)$ not only from the execution of action a, but also from the execution of its similar actions. This action abstraction is later on stopped, and then each action is treated on its own, thus, refining the values of $P(s'|s, a)$ learned with abstraction.

7.3.4 Genetic Algorithm

We should analyze the generality, in terms of different environments and starting points, of the parameters obtained by the genetic algorithm. Further work should also focus on designing an agent capable of identifying the complexity of the task being performed, so that the parameters can be switched from one set to another. We will explore the use of Case Base Reasoning techniques on this "situation identifier" agent.

7.3.5 Real experimentation

The results obtained through real experimentation confirmed that, as already seen through simulation, the bidding coordination mechanism and the mapping and navigation methods work appropriately. Nonetheless, the scenarios used in the real experiments were not very complex, and some more experimentation on more complex scenarios should be performed. These new scenarios should include some more obstacles, eventually having some cul-de-sacs, so that the robot would need to undo the path already done.

However, the big next step on our research is to move the experimentation to outdoor environments. The main difficulty of doing so is the availability of a vision system for outdoors, which we do not have at this moment. However, we think that the successful results obtained on indoor unstructured environments could be quite easily obtained outdoors, since neither the navigation method nor the control architecture are dramatically affected by the differences of indoor/outdoor environments.

7.3.6 Case Based Reasoning

Besides the use of CBR described in the Genetic Algorithm approach, we also plan to add a CBR agent that would bid for actions. This agent would use the information of past experiences in different trials (stored in form of {situation,action,result} tuples) to recognize similar situations, and would then bid for executing the actions (or similar actions) that best suited those situations. The difficulty of this approach is to find the proper way to characterize the situations and how to compare two situations in order to find out how similar they are. In this approach we also face the credit assignment problem, since we cannot evaluate a situation-action experience until the robot either successfully reaches the target or fails in its mission.