Can Planning and Reactive Systems Realize an Autonomous Navigation*

Maki K. Habib

GMD-Japan Research Laboratory
AIM Building 8F
KokuraKita-Ku, Kitakyushu-City, JAPAN
maki.habib@gmd.gr.jp

Abstract:

The major challenges facing navigation of an autonomous mobile robot and need to be addressed are stem from, incomplete and uncertain knowledge of the environment, unpredictable aspects of real environment and surroundings, sensor limitation and uncertainty of sensory information (range, resolution limits, noise and occlusions), uncertainty in the effects of the robot own action (imperfect actuators), the need to respond quickly to environment demand in which the robot has to operate at a pace dictated by the interaction with its surrounding (limits on computation time due to restrictions imposed by the environment). These issues are fundamental to autonomous systems that have to function effectively while navigating and interacting with unknown, unstructured and dynamic environment. Several approaches have been developed to address this important issue at various levels in mobile robot's control architectures. This paper discusses the main approaches in the field of autonomous navigation. It focuses on the challenges, needs, fundamental issues along with the requirements that enable a mobile robot to move autonomously, purposefully, reliably and without human intervention through unstructured real world environments that have not specifically prepared for them at design time.

1. INTRODUCTION

Effective development of autonomous mobile robots has been one of the major research efforts of many universities and institutions due to their immediate applicability in a variety of tasks such as space missions, operations in hazardous environments, underwater, demining, civil security, scientific exploration, factory floors, disaster areas, and other services. Research on mobile robots focuses on the realisation of robotics structures that are able to move autonomously, purposefully, reliably and without human intervention through unstructured real world environments that have not specifically prepared for them at design time. Thus, autonomous navigation capability in a real environment represents the core functionality any mobile robot should be endowed with. At the same time it poses a series of problems that need to be addressed.

In general, robots used in manufacturing sector lack autonomy and flexibility, in the sense that while they can autonomously perform pre-specified tasks, they cannot autonomously reason about the task and answer important questions like: Which navigation task should be performed? What actions should be selected and performed to accomplish a given navigation task? How to know whether the selected actions were successfully and efficiently performed? While performing the selected actions, do these actions accomplish the given navigation task or do they need to be reconsidered and adjusted by selecting new actions?

Even if the navigation task of a robot is determined a-priori, the robot needs to be able to autonomously adapt to its practical environment in order to perform this task efficiently.

---

* Appeared at the International Symposium on Robotics ISR'99, October 1999 Tokyo, JAPAN. Pp. 683-690..
This paper reflects a private work done by the author.

Autonomous mobile robots must interact with complex environments that have not been engineered specifically for them. Interactions with such environments are extremely difficult to model because they are governed by an enormous number of independent variables [2].

Robot's flexibility is essential to achieve the target of a given task successfully and efficiently. Therefore, it is desirable to build systems that have facilities for exploring sensory feedback to allow adaptive interaction with the environment. Autonomy requires systems that are not only capable of controlling their motion in response to sensor inputs, but that are also capable to react to unexpected events and accordingly change course if necessary by efficiently deciding the next action to be performed while considering multiple conflicting objectives simultaneously. Moreover, it should be able to overcome errors in perceptions and actions.

Real environments contain elements of variability that limit the use of prior information. Objects such as furniture may change in position, and shape; other objects may move with unpredictable dynamic, and even topological properties may change too. In addition, information collected at run time by the sensors is affected by the specifications and performance of the sensors attached to the robot and by the inherent difficulty of the perceptual interpretation process.

The ability for an autonomous mobile robot to navigate intelligently in uncertain and dynamic environments requires consideration of multiple issues. The robot must be able to operate under conditions of imprecision and uncertainty and incomplete information about the environment in a timely manner. Perceptually acquired information is also typically noisy and incomplete. Furthermore, the execution of control commands is not completely reliable while the dynamics of real-world environments is complex and unpredictable. To cope with these difficulties, robot's controller must be able to respond to unforeseen events as soon as they are perceived [3-5,10-12].

Several approaches have been developed to address this important issue at various levels in mobile robot's control architecture. This paper discusses the main approaches in the field and focuses on the fundamental issues and the requirements that enable a mobile robot to move autonomously, purposefully, reliably and without human intervention through unstructured real world environments that have not specifically prepared for them at design time.

2. ROBOT CONTROL ARCHITECTURES

To construct flexible autonomous robot systems that can perform reliably in unstructured environments different architectures have been proposed and investigated. These architectures range from symbolic artificial intelligence planners to the increasingly popular behavior-based approaches. These include planning based architectures, that partition robot's functionality into perceive-plan-control cycle; reactive control architecture that directly maps perceptions of robot's world into actions; and hybrid control architectures, that combine a layered organization with a behavior-based decomposition of the execution layer.

2.1 Planning Control Systems

Most of the early work concerning navigation of mobile robots uses internal geometrical representations of robot's environment to plan and perform navigational tasks. Planers assume complete knowledge of the environment and the robot's sensing and actuating capabilities. Robot systems have traditionally been built using "perceive-plan-control" cycle [13-15]. In this cycle all the actions needed are handled in a co-ordinated manner. Deliberative planning is necessary for purposeful behavior in response to run time goals. Planning based robot control architecture would often create elaborate models of the environment (knowledge) and reason about how to accomplish their goals. The model of the environment may either be acquired off-line or it may be built on-line based on sensory information. In a compromise, a coarse model may be provided a priori and the model may then by elaborated on-line at the location where interaction is requested. Through the use of such models the robot control system may perform planning of missions as a sequence of actions that enable a robot to move from one place to the other. The general approach was to sense the world, build a world model, synthesise a plan and generate a sequence of actions with respect to goals, and then execute the plans through motor control system. These plans when executed correctly, it will in principle ensure goal achievement. The most powerful aspect of these systems is they lend themselves to theoretical analysis and verification of properties such as plan correctness, optimality, etc. But, the limitations of this approach quickly became obvious in which planning control systems have higher probability for reaching unanticipated situations. Planning based systems were often incapable of operating outside of controlled environments. This is because, such systems require an accurate world model to reason properly about their plan and actions at the time it can not perform reasonably in the face of uncertainty, i.e., environmental or sensory noise often made them unreliable. This leads to a conclusion that the resolution of any planning activity is limited by the resolution of the knowledge available at the planning time. Therefore, as a solution all robot's decisions and actions must take place at the time scale of its environment. Also, it is important to note that no amount of domain knowledge will be sufficient if a robot is expected to learn to cope with novel situations not covered in its domain knowledge. To make a robot generally "intelligent", it should be provided with general learning abilities.

2.2 Reactive Control Systems

In response to the limitations and problems associated with the planning based approach, attempts were made to abandon the planning approach and new efforts start to focus on developing an intelligent autonomous robot capable of interacting with a dynamic environment and better meet the demand of real time performance [16-20]. Accordingly, reactive based robot control architecture was introduced to demonstrate navigation capabilities that are quicker and better than those of planning systems. As the name already suggests, reactive controller responses to unexpected change (events) in robot's environment through coupling of perception and action. Reactive architectures have shown improved reliability when compared with classical sense-plan-act architectures. Rather than attempting to model the world, these systems had multiple task modules that react directly to sensory information. Reactive control systems use rules to combine sensor inputs with state information to produce motor control outputs and state changes in a common and successful approach. The basic idea underlying reactive control systems is the idea of behaviors. In contrast to the planning systems where the control architecture was split into functional tasks (sense, world model, plan, execute), reactive systems are built as multiple independent tasks which operate in parallel. Each behavior is responsible for a particular task. A behavior processes its own sensory information and issues its own motor commands. In order to co-ordinate the final motor commands, each behavior can disable the motor commands of the other behaviors that are in conflict with itself.

A particular form of reactive control using a hierarchy of rules is Brooks's subsumption architecture [16]. In a subsumption architecture, rules are assigned priority with high priority rules capable of overriding low-priority rules that would otherwise be enabled. This allows the programmer to develop several levels of competence for the robot. The lower priority rules associated with lower levels of competence, define behaviors such as line following, while the higher priority rules, associated with higher levels of competence, define more complex behaviors (more complex goal directed activities), such as mapping or solution of subgoals.

Intelligent behavior is achieved through coordination of a set of purposive perception-action units, called behaviors. Based on carefully selected sensory information, each behavior produces commands to control the robot with respect to a well-defined (and usually narrow) aspect of the overall task. However, it is not given how to select a set of appropriate behaviors for a given task. This will be denoted the problem of specification and design of behaviors. Another major problem is to choose the most appropriate action or to coordinate the behaviors to produce a rational next action. This problem is denoted the action selection problem.

This approach aimed at obtaining the desired action without any use of plans. While plans are not represented explicitly in such systems, they are in some sense implicitly designed in to the system through the pre-established interactions between behaviors. One of the problems, which have received the most attention, is the control in the "survival" layer. This is probably due to the fact that it is manageable to build and demonstrate methods at this layer while higher level layers require much more information and they are correspondingly much harder to demonstrate. In addition, this approach has created new set of problems, such as co-ordinating behaviors into complex system. This is because, reactive systems consists of a number of distributed sensor-motor processes, thus mechanism should be devised to deal with scheduling, management, co-ordination of and communication between these modules so that coherent behavior can be achieved. Also, such systems are not able to learn and adapt their behaviors to become skilful in their action.

With behavior based control, much effort should be made to solve aspects like [6], formulation of behaviors, and the efficient consideration of conflicts and competitions among multiple behaviors.

While this approach allows generation of timely responses to runtime contingencies no formal guarantees can be made about goal achievement or global optimality of actions. In addition, formal analysis is sacrificed since this approach does not synthesis plans ahead of time.

Finally, there is ongoing debate regarding the complexity of behavior that can be achieved using solely reactive actions

3.3 Hybrid Control Systems

Hybrid method is the hierarchical approach to the navigation problem. According to this approach the task is solved at different levels of abstraction, in effect creating several subtasks at the different levels. The logical architecture of most hybrid systems is layered according to decreasing levels of deliberation from top to bottom. The level of abstraction also decreases from top to bottom with highly level planning components operating on symbolic description and the low-level reactive behaviors operating at the signal level.

Due to the limitations associated with planning and reactive approaches, researchers have new focus by exploiting the benefits of each approach and develop hybrid architecture that integrate the planning and the reactive components to complement the shortcomings of each other. While behavior based systems are adaptive to unknown and dynamic environments they cannot guarantee efficient goal achievement. Also, their ability to demonstrate more sophisticated behaviors seemed limited. Planning systems on the other hand can generate efficient plans while they lack the ability to efficiently handle changes in the environment and system complexity. Also, the robot should be able to accept new tasks and information about the environment at any time. Deliberative planning and reactive control are equally important for mobile robot navigation, each complement the other and compensates for the other's deficiencies. Thus, by the integration of planning systems and behavior-based systems researchers hope to construct systems which are both adaptive to changes in the environment and ensure efficient goal achievement. In order to achieve this synergetic effect of integrating planning with reactive behavior-based systems it is required to deal with a number of problems. Planning systems usually use abstract symbolic representation of physical objects whereas behavior-based systems usually operate on raw sensory information with limited processing. How to close the gap between planning and behavior-based system components is thus an important issue, which has received a considerable amount of attention in the literature [4, 21-25]. Some of the work in hybrid systems deals with integrating planning and reactive behavior-based system components. Others, do not exactly integrate the two components, rather they devise new approaches which combine the best of the planning and the behavior-based schools. Another direction considers one of the presented approaches, either planning or reactive, and try to push its limits towards the other. The problems of action selection and uncertainty handling are the main focus of this work.

3. THE CHALLENGES AND THE NEED FOR SOLUTION

While the use hybrid architectures is gaining increasing consensus as new control paradigm that supports autonomous navigation, a number of technological gaps remain. Among these, the following can be emphasized:

- How to design simple behaviors that guarantee robust operation in spite of the limited knowledge available at design time; e.g., designing an obstacle avoidance behavior that is effective in face of unknown obstacle configurations.
- How to coordinate the activity of several, possibly competing behaviors in order to perform a complex task; e.g., coordinating goal-achieving and obstacle avoidance behaviors to reach a target position while avoiding unforeseen obstacles.
- How to ensure coherence between representations used at different layers; e.g., registering perceptual information with map information. In general an autonomous navigation requires a number of heterogeneous capabilities, including the ability to,
- Execute elementary goal-achieving actions. This has to be done by favoring actions that contribute to one or several goals.
- Provide fast and timely response concerning real time and unexpected events, like the sudden appearance of an obstacle;
- Build, use and maintain a map of the environment; determine robot's position with respect to this map;
- Form plans that look ahead to pursue specific goals or to avoid undesired situations;
- Adapt to change in the environment.

The main challenges facing robotics rise from the following issues [25]:

- Incomplete knowledge of the environment. The robot usually cannot assume complete and consistent environmental knowledge, which is necessary to plan detailed courses of actions ahead of time. Often planning and execution should be interleaved.
- Unpredictable environment and surroundings. In real world and especially dynamic environments, the robot is for most part unable to predict how events will unfold in the future. Thus again rendering it infeasible to plan ahead of time. To perceive changes in the environment, the robot has to sense repeatedly and often. (having sensing modules that work autonomously).

- Imperfect sensors. Sensors are inaccurate, noisy, faulty and with limited field of view, thus “what the sensor see might not what is”, which means that decisions might be based on wrong information.
- Imperfect actuators. A robot cannot assume correct and perfect execution of its actions due to actuators imperfections and uncertainties in the environment. Thus, even if the robot had complete knowledge so that it could plan ahead of time, due to the imperfection of actuators etc. it would have to do some re-planning in order to take correction action.
- Limited time. The time available to decide what to do is limited, because the robot has to operate at a pace dictated by its surroundings.

There are several ways to reduce the impact these problems, such as, utilizing better and more sophisticated technologies. But, using better equipment comes at significant increase in cost. Therefore, there is a need to consider the current resources at hand to face the problems outlined above and devise systems and techniques that can cope with and handle them.

Planning and reactivity are not mutually exclusive characteristics of robot’s control architecture. But, they serve complementary purposes while they are implemented in different ways. Hybrid control architecture rise the hope that by having a synergetic integration between the two approaches will enable the robot to deal efficiently with dynamic environments. Interaction between the components of planning system and that of reactive system is an important architecture issue and should be done asynchronously. This allows the fast reactive components to react in real-time whereas the slow planning components can run at slower paste without affecting the system fast response to run time contingencies.

To develop persistent robots that can achieve complex tasks in dynamic and uncertain environments. A robot of this type requires a number of capabilities,

1. The robot must be taskable in that it can accept requests for, or specification of, activities and to integrate these tasks into its goal structure at run time.
2. The robot must be able to synthesis new plans at run time as necessary in order to achieve its goals. Planning is one of the most important capabilities of an intelligent robot to have. The ability to plan is closely linked to the robot’s representation of its environment. It is also necessary to modify or completely rebuild plans in response to changes in the environment.
3. The dynamic nature of the environment necessitates that robot be able to deal with unexpected change. Thus, robots must be able to react to unanticipated events by taking appropriate actions in a timely manner, while continuing activities that support current goal.
4. Unpredicted environment may lead to failure of the generated plans for individual tasks. Hence, robots must be robust by having the ability to recover from failures by adapting their activities to the new situation. Such ability may include the ability to modify the plan while continuing its execution is critical in domains where it is infeasible to halt all execution activities while re-planning.
5. The robot should have the ability to modify its behavior based on experience, i.e., to capability to learn.
6. The robot should have a form of focusing mechanism that enable it to determine what sort of information it needs to use in solving a problem out of the overwhelming amount of information available from the real world.
7. The robot control architecture should be enhanced with impasse-driven processing, i.e., when a robot is unable to generate a response at a certain situation due to lack of knowledge, the impasse-driven function registers this need as a subgoal to be resolved and try to find the relevant knowledge.
8. The robot should be able to perform all of the above operations even in the face of uncertainty about the environmental (world) state.

4. ISSUES OF CONSIDERATION

The design of robot’s control architecture is greatly influenced by the capabilities, properties that should be included, and the types of environments should be anticipated.

In addition to the requirements stated in section 3, There is a need to focus on specific issues that are of concern to autonomous navigation in dynamic environments.

4.1 Formulating Action Selection Mechanism

Autonomous systems must be able to plan their actions in a flexible way to make good use of their resources. In reactive systems, the control of a robot is shared between multiple behaviors with different objectives. Each behavior is responsible for controlling the robot to achieve or maintain a particular objective such as following a target or avoiding obstacles. The objective of one behavior might be in conflict with the objectives of others. In most cases when deciding which next action to take, multiple conflicting objectives should therefore be considered simultaneously. This is known as the action selection problem. The problem of run-time choice between multiple parallel, competing, conflicting and overlapped goals, that must respond to unpredictable and passing opportunities in the world where the robot is moving. This will tell how does a mobile robot decide which goals to pursue at a given moment, when to interrupt, and when to opportunistically divert, in response to events within its environment. Thus a major issue in the design of systems for control of autonomous mobile robots is the formulation of effective mechanisms for coordination of the behaviors activities into strategies for rational and coherent behavior. However, due to several constraints such as environmental complexity and unpredictability and due to robot's limited resources, action selection cannot be completely rational or optimal. Thus, it is appropriate to consider selecting good enough actions that satisfy the objectives.

Numerous action selection mechanisms can be found in the literature [27]. Among the available mechanisms two groups can be classified; state based (arbitration) and continuous (command fusion). The first are applied when a relevant subset of robot's behavior needs to be activated in a given state. The later group is used to coordinate activities of the set of behaviors that are active simultaneously. In most systems the use of both mechanisms is necessary because they are not competitive but complementary.

Two possible cases can be considered for the action selection. The first case covers behaviors with common objectives, i.e., homogeneous, those agree on what action or set of actions needs to be selected. The second case covers behaviors that have distinct objectives, i.e., heterogeneous behaviors. Homogeneous behaviors imply redundancy, and redundancy can be exploited to improve the reliability of the system and to enable uncertainty handling. Action selection for heterogeneous behaviors is more complicated due to the possible conflict between the objectives of the behaviors. Thus, there is a need to find out a suitable approach for selecting a proper behavior that is the best to satisfy the situation.

4.2 Representation of Knowledge and the Use of Map/Model

Robot navigation in large-scale environment requires an adequate representation of the working space. This representation should be abstract enough to facilitate higher level reasoning tasks like strategic planning or situation assessment, and still be detailed enough to allow the robot perform lower levels tasks like trajectory generation of self-localisation.

In the field of mobile robotics there have been various approaches to mapping of the environment. Some systems use a geometric representation, whilst others take the topological approach, whereby the environment is modeled as a graph containing nodes representing distinct locations, and pathways between locations are denoted by arcs. However, most methods assume static environments, which can be problem in real world application where some allowance must be made for change within the robot's environment [28].

An autonomous mobile robot has a fundamental task of exploring its environment, learning its structure, building a proper representation of the world, having efficient tools to abstract, search, modify and manipulate that representation

A common belief in the robotics field is that robots need to represent and reason about information at different levels of abstraction at the same time [4,5]. There are several reasons for this. First: different tasks call for different types of representation. For example, global navigation strategies are more easily planned using a topological map, where we can decide the sequence of rooms and corridors to be traversed; but fine motion control needs geometric information to precisely control navigation among features and obstacles. Second: geometric information is difficult to collect and expensive to handle, and we cannot pay the price to maintain a detailed geometric representation of the entire environment where the robot can operate. The final reason is ontological adequacy: fine grained information is difficult to obtain a priori and is likely to change with time; coarse maps are easier to estimate and more prone to remain valid over time.

Symbolic world models are required by the control architectures that attempt to understand their environment and predict the effects of their actions. Symbolic models typically take the form of global databases that maintain a list of objects, their properties and locations.

4.3 Learning

One of the fundamental capabilities of an intelligent robot to posse is the ability to modify its behavior based on experience, to learn. This means, any architecture attempts to behave intelligently should have some learning component. The nature of learning by control architecture varies widely. Learning is widely accepted as useful and necessary. The question of what to learn is much more open to debate. Learning can increase the range of problems that can be accomplished as well as increase the efficiency in which a robot is able to perform tasks. But, indiscriminant learning will harm the efficiency of the robot if the learned knowledge is of low usefulness relative to the cost The robot is said to have deliberative learning capability if it is done based on cost evaluation and a learning decision has to be made accordingly. Also, the robot may have reflexive learning capability. This type of learning is done automatically, i.e., the robot does not consider the possible cost of learning a particular piece of knowledge. Humans in general, cannot help but learn from their experiences, and we certainly are not able to explicitly unlearn something if we decide that it is not worth retaining.

Learning and planning generally work together to a large extent, with planning generating opportunities for learning and learning helping to generate more efficient plans. An important point for planning from experience is the ability to represent the world in symbolic form.

Different types of learning has been observed and identified. Some of these learning types are, learning by abstraction, learning by instruction, learning by analogy, learning by experimentation, and case based learning.

4.4 Dealing with Uncertainty

The designers of highly specialized behaviors exploit very specific knowledge about the task and the environment to come up with efficient and economic implementations. Such specialized modules or behaviors operate reliably as long as their assumptions are valid. If they are placed in environments in which they are not programmed to handle, they may fail completely due to invalidation of their assumptions. Thus the performance of behaviors highly depends on various assumptions, imposed by the sensors and algorithms used by the behaviors. In order to perform reliably in face of uncertainties other sources of information are usually used. Thus an important issue is to figure out how to combine the evidence provided by multiple sources of information in a way that would allow system components to cope with uncertainties.

An intelligent robot has to be able to reason and make decisions on the basis of less than perfect information about its environment. A robot may be uncertain about its exact location, its future location after performing some action, which objects are (or will be) in its proximity, etc.

Uncertainty plays an important role in robot navigation at many levels, such as sensor interpretation, sensor fusion, map making, path planning, self-localization, and control. Dealing with uncertainty also constitutes the focus of a large research effort in artificial intelligence, which has led to the development of a number of new theories and new techniques. These techniques use probabilistic representations or fuzzy set theory to model uncertainty in sensor information and the outcome of actions taken by the robot. Example of these techniques includes occupancy grids, and intelligent sensor fusion. Recent successes of mobile robots in practical areas showed that these techniques have reached a level of robustness, which allows robots to operate even in crowded environments.

There are two sources of problems in getting a reliable model of the robot and its environment. The first one is the environment. Knowledge of the environments is necessarily incomplete, uncertain, and approximate: maps typically omit some details and temporary features, spatial relations between objects may have changed since the map was built, and the metric information may be imprecise and inaccurate. Moreover, the environment dynamics is typically complex and unpredictable: objects can move, other agents can modify the environment, and relatively stable features may change slowly with time. The second problem is that the interaction between the robot and the environment may be difficult to model. In addition, the results of robot’s actions are influenced by a number of environmental conditions that are hard to be accounted for and may lead to imprecise data; and errors in the measurement interpretation process may lead to incorrect beliefs.

To obtain accurate and reliable information from its sensors, the robot should fuse sensor data from multiple redundant sensors preferably from several different positions. We can thus directly conclude that Sensor Data Fusion is an essential prerequisite to robot autonomy.

5. CONCLUSIONS

The main approaches of robot control architecture have been presented with focus on the challenges, needs, fundamental issues and the requirements that enable a mobile robot to navigate autonomously, purposefully, reliably through unstructured real world environments.

Autonomous navigation represents a fundamental milestone towards intelligently behaving mobile robots. The paper show that deliberative planning and reactive control are equally important for mobile robot navigation, each complement the other and compensates for the other’s deficiencies, and there is a need for synergetic integration between them with focus on issues that need to be included within such synergy.

REFERENCES

[1] L. Wang, and W. H. Tsai, “Collision Avoidance by a Modified Least-Mean-Square-Error Classification

Scheme for Indoor Autonomous Land Vehicle Navigation", Journal of Robotic Systems, Vol. 8, No. 5, pp. 677-698, 19991.

[2] E. Gat, "Towards Principled Experimental Study of Autonomous Mobile Robot", In proceedings of the International Symposium of Experimental Robotics, Stanford CA, 1995.

[3] U. Nehmzow, T. Matsui, and H. Asoh, " Virtual Coordinates: Perception-Based Localisation and Spatial Reasoning in Mobile Robots", Proceeding of the International Conference on Intelligent Autonomous Systems 5, Sapporo, 1998.

[4] J. Yen, and N. Peluger, "A Fuzzy Logic Based Extension to Payton and Rosenblat's Command Fusion Method for Mobile Robot Navigation", IEEE Trans. on Systems, Man and Cybernetics, Vol. 25, No. 6, pp. 971-978, 1995.

[5] A. G. Skarmeta, and H. Martínez Barberá, "Fuzzy Logic Based Intelligent Agents for Reactive Navigation in Autonomous Systems", Fifth International Conference on Fuzzy Theory and Technology (FT&T'97), Durham, USA, pp. 168-171, Mar 1997.

[6] A. G. Skarmeta, H. M. Barberá, and M. Sánchez Alonso, " Learning Behavior Fusion in Autonomous Mobile Robots", IX ESTYLF'99, Mallorca, SPAIN, Sep 1999

[7] M. Frank, "Strategies for Intelligent Agent Exploration of Complex Environments." Term paper for Michael Genesereth, March 1990. http://medg.lcs.mit.edu/mpf/papers/Frank/Frank-90a.html.

[8] D. E. Wilkins, K. L. Myers, J. D. Lowrance, and L. P. Wesley, "Planning and Reacting in Uncertain and Dynamic Environments", Journal of Experimental and Theoretical AI, Vol. 6, pp. 197-227, 1994.

[9] A. Saffioti, "Handling Uncertainty in Control of Autonomous Robots", Technical Report TR/IRIDIA/97-8, University Libre de Bruxelles, 1997.

[10] R. Cattoni, "Behaviours as Bridges Between Symbolic Reasoning and Reactivity", In Proceedings of the Italian Workshop on Planning IPW'93, Rome-Italy, 1993.

[11] J. Van Dam, "Environment Modeling for Mobile Robots: Neural Learning for Sensor Fusion", Ph. D. Thesis, University of Amsterdam, 1998.

[12] S. Lacroix, S. Flury, H. Haddad, M. Khatib, F. Ingrand, G. Bauzil, M. Herrb, C. Lemaire and R. Chatila, "Reactive Navigation in Outdoor Environments", submitted to the International Journal of Robotics Research, 1998.

[13] H. P. Moravec, "The Stanford Cart and the CMU Rover", Autonomous Mobile Robots, pp. 407-419, 1982.

[14] A. Kosaka, and A. C. Kak, "Fast Vision Guided Mobile Robot Navigation Using Model Based Reasoning and Prediction of Uncertainties", CVGIP: Image Understanding, Vol. 56, No. 3, pp. 271-329, 1992.

[15] P. Pirjanian, and H. I. Christensen, "Modelling and Planning for Sensor Based Intelligent Robot Systems, World Scientific, Chapter Hierarchical Control for Navigation Using Heterogeneous Models, 1995.

[16] R. A. Brooks, "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, Vol. 2, No. 1, pp. 14-23, 1986.

[17] R. C. Arkin, "Motor Schema Based Navigation for a Mobile Robot: An Approach to Programming by Behavior", IEEE ICRA'87, pp. 264-271, 1987.

[18] P. Maes, "How to Do the Right Thing", Technical Report NE43-836, AI-Laboratory, MIT, 1989.

[19] J. K. Rosenblatt, and C. E. Thorpe, "Combining multiple goals in Behavior Based Architecture", Proc. of IROS'95, 1995

[20] M. Mataric, "Behavior-Based Control: Examples from Navigation, Learning and Group Behavior", Journal of Experimental and Theoretical Artificial Intelligence, Vol. 9, No. 2-3, pp. 323-336, 1997.

[21] D. W. Payton, J. K. Rosenblatt, and D. M. Keirsey, "Plan Guided Reaction", IEEE Trans. on Systems, Man, and Cybernetics, Vol. 20, No. 6, pp. 1370-1382, 1990.

[22] R. C. Arkin, " Towards the Unification of Navigational Planning and Reactive Control", AAAI'90, 1989.

[23] E. Gat, "Integrating Planning and Reacting in a Hetrogeneous Asynchronous Architecture for Controlling Real-World Robots", AAAI'92, pp. 809-815, 1992.

[24] A. Saffiotti, "Some Notes on the Integration of Planning and Reactivity in Autonomous Mobile Robots", AAAI'93, pp. 122-126, 1993.

[25] P. Pirjanian, H. Blaasvear, and H. I. Christensen, "AMOR: An Autonomous Mobile Robot Navigation System", Proc. IEEE Conf. on Systems, Man and Cybernetics, ol. 3, pp. 2266-2271, 1994.

[26] P. Pirjanian, "An Overview of System Architecture for Action Selection in Mobile Robotics", 1997.

[27] D. C. Makenzie, R. C. Arkin and J. M. Cameron, "Specification and Execution of Multiagent Missions", Autonomous Robots, Vol. 4, No. 1, 1997.

[28] C. Owen and U. Nehmzow, "Map Interpretation in Dynamic Environments", Proc. of the 5th International Workshop on Advanced Motion Control", 1998.