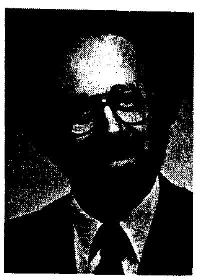
Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation

Ronald C. Arkin
School of Information and Computer Science, Georgia Institute of Technology, Atlanta, GA 30332-0280, USA

Reactive navigation based on task decomposition is an effective means for producing robust navigation in complex domains. By incorporating various forms of knowledge, this technique can be made considerably more flexible. Behavioral and perceptual strategies which are represented in a modular form and configured to meet the robot's mission and environment add considerable versatility. *A priori* world knowledge, when available, can be used to configure these strategies in an efficient form. Dynamically acquired world models can be used to circumvent certain pitfalls that representationless methods are subject to.

The Autonomous Robot Architecture (AuRA) is the framework within which experiments in the application of knowledge to reactive control are conducted. Actual robot experiments and simulation studies demonstrate the flexibility and feasibility of this approach over a wide range of navigational domains.

Keywords: Mobile robots; Reactive control; Knowledge-based systems; Artificial intelligence; Schemas.



Ronald C. Arkin was born in New York, N.Y., in 1949. He received the B.S. Degree from the University of Michigan in 1971, the M.S. degree from Stevens Institute of technology in 1977 and a Ph.D. in Computer Science from the University of Massachusetts, Amherst in 1987. From 1977 until 1985 he was a member of the faculty of Hawthorne College in Antrim, N.H., serving as the Computer Science Department Chair for the last three years. After receiving his Ph.D. in 1987, he assumed the position of Assistant Professor of Information and Computer Science at the Georgia Institute of Technology.

His research interests include reactive control and action-oriented perception for the navigation of mobile robots, dynamic replanning in hazardous environments, fault tolerant survivable control systems, navigation in three dimensional environments, and cognitive models of perception. He is included in several *Who's Who* publications including *Who's who in the World.*

North-Holland
Robotics and Autonomous Systems 6 (1990) 105–122

1. Introduction

Considerable success has been achieved of late in the design and implementation of working robotic systems that can cope with a dynamically changing world. Several of the companion papers in this volume describe such systems. They are characterized by robust navigational capabilities and rapid, real-time response to the environment.

It is equally important to design for flexibility and adaptability in mobile robot navigational systems. Our approach, embodied in the Autonomous Robot Architecture (AuRA), allows for such freedom. It affords the decided advantages of reactive navigation: modular and incremental design; the ability to cope with a changing world; and the production of intelligent emergent behavior. In AuRA, however, both the motor behaviors and perceptual strategies can be readily reconfigured based on current environmental conditions, available *a priori* knowledge, and the robot's intentions based on the current mission's needs.

This paper describes how knowledge can be used to allow a robot to exhibit different navigational abilities under different circumstances. The methodology of reactive navigation is described in Section 2. Section 3 outlines the forms of knowledge that are especially pertinent to this form of mobile robot control.

The Autonomous Robot Architecture is presented in Section 4. AuRA is designed as much as possible to be a generic architecture, suitable for use over a multiplicity of domains. These include navigation in buildings, in outdoor campus settings, in aerospace or undersea applications, over contoured landscapes, in manufacturing environments and other settings. Representative examples, both with actual robot experiments and simulation studies are presented in Section 5. A summary and conclusions regarding the role of knowledge in reactive control complete the paper.

2. Reactive Navigation

Reactive navigation is a form of robot control that is characterized by a stimulus-response type of relationship with the world, not unlike the viewpoint held by the behaviorist psychologists, epitomized by Skinner [30]. Mentalistic (representational) structures are eschewed and the organism (in our case, robot) reacts to the immediacy of sensory information in what we would term a very low-level non-cognitive manner.

Complex behaviors emerge as a combination of simple low-level responses to the rich variety of stimuli the world affords. Typically this involves decomposition of tasks into a collection of distributed parallel sub-tasks. Further, sensor data is normally channeled directly to the individual sub-tasks, reducing significantly the computational demand typically found in navigational regimes requiring world model building.

There are many representative examples of this form of navigation, a few of which will be described here. Brooks' subsumption architecture [15] has demonstrated robust navigation for mobile vehicles in dynamically changing domains. It is a layered architecture, well-adapted for hardware implementation [18]. It has been used in a wide range of robots, including legged ones [17]. Much of Brooks' work has been motivated by the desire to produce artificial insects. There is a deliberate avoidance of world modeling which is captured by the statement that *the world is it's own best model* [16].

Payton has described a collection of motor responses that are termed "reflexive behaviors" [27]. These behaviors react directly to sensory information yielding intelligent emergent behavior. Payton, Brooks, and several other proponents of reactive control incorporate the concept of *arbitration*. Multiple behaviors compete for control of the vehicle with a winner-take-all mechanism deciding the result. Only one behavior dominates the vehicle at any time, although the dominant behavior can change frequently in rapid response to environmental sensing. Earlier work by Kadonoff [22] also employs an arbitration scheme.

Kaelbling has developed a reactive architecture [23] that is an extension of Brooks' work. The emphasis is on embedded systems for real-time control. A hierarchical competency level for behaviors is established which is mediated by a high-level controller. The switching mechanism results in a decision as to which behavior is in control of the robot at any given time.

Firby has developed a different form of reactive control by utilizing modules called RAPs (Reactive action packages) which encapsulate tasks for a robot [19]. Situation-driven execution via goal satisfaction is the predominant mode of operation. Unsatisfied tasks are selected along with methods for their achievement that are consistent with the current world state. RAPs provide a hierarchical building block mechanism consistent with the task decomposition methodologies ubiquitous in reactive navigation.

Agre and Chapman in their PENGI system [1] have used reactive control in the domain of game playing. Several behaviors are active at any time, controlling the strategies used by a video game penguin and its relationship with other objects and entities in the world. An arbitration strategy is in evidence here as well.

Reactive navigation in AuRA [9] addresses reactive control in a manner that is significantly different than the approaches described above. Arbitration is not used for coordinating the multiple active agents; potential field formulations are employed to describe the reactions of the robot to the world; and explicit representational knowledge is used to select and configure both the motor and perceptual strategies used for reactive control.

Just as many psychologists moved away from behaviorism as an acceptable description of human information processing to cognitive psychology, our research has expanded to include many of the concepts forwarded by this relatively new discipline. Within AuRA, we do not abandon the unquestioned advantages of reactive control in the behaviorist sense as described above. It is our premise, however, that by encapsulating these stimulus-response behaviors in a form that is more flexible, adaptable, and controllable at a cognitive level above the reactive regime, robots can be created that are intrinsically more useful for a variety of missions over a wide range of task domains. The following section describes the structures that are used to represent these behavioral and perceptual techniques for navigation within AuRA and their motivation by modern psychology and neuroscience.

3. Applicable Knowledge

Despite the assumptions of early work in reactive control, representational knowledge *is important* for robot navigation. The fundamental problem lies in representing what is appropriate for the task. Amarel's classic paper [2] shows the importance of appropriate knowledge representation for problem solving using artificial intelligence.

The question is first what needs to be represented for successful general-purpose mobile robot navigation and then second how it is to be represented. Our answer to the first question is threefold: motor behaviors that are used to describe the set of interactions the robot can have with the world; perceptual strategies that provide the required sensory information to the motor behaviors; and world knowledge (both *a priori* and acquired) that is used to select (and reconfigure when necessary) the motor behaviors and perceptual strategies that are needed to accomplish the robot's goals. The remainder of this section answers the question as to how to represent this knowledge.

3.1. Schemas

Although it is not our goal to create robots that function internally in an identical or even similar manner as humans or animals do, it is our belief that tremendous insights can be drawn from these biological systems which already successfully achieve the tasks we would like our robots to perform. By studying psychological, ethological (behavioral), and neuroscientific theories, some of the models developed to explain these behavioral systems can serve well to motivate our approaches to robot navigation.

3.1.1. Motivation

Schema theory has been developed within cognitive psychology as a means for the codification and coordination of motor action and perceptual activity [14,21,28]. In particular, Neisser [25] describes the role of schemas within the context of the action-perception cycle. Perceptions are derived from the environment which in turn modify a cognitive map of the world resulting in motor actions which alter both the state of the world and the perceptions which arise from it. Norman and Shallice [26] have also used schema theory as a means for differentiating between willed and automatic behavior. Their studies provide motivation for the co-existence of both hierarchical and reactive control systems as is found in AuRA.

Psychologists have defined schema in a variety of ways. For our purposes, they are the primitives that serve as the basic building blocks of perceptual and motor activity. Arbib [3] was the first to apply schema theory to the robotics domain. This involved studies in the application of schema theory to dextrous hand control [4]. Other related research delves into neuroscientific models for schema operation within the brain itself.

The approach used in AuRA for developing navigational techniques in a new problem domain is as follows. First, the motor behaviors (motor schemas) required for the particular robotic application and domain are determined and then tested in simulation. Next, the perceptual strategies required to provide the information necessary for those motor schemas are designed and then implemented on the actual robotic vehicle. The techniques involving motor and perceptual schema design are discussed next.

3.1.2. Motor Schemas

Motor schemas, as used within AuRA, comprise a collection of individual motor behaviors each of which reacts to sensory information gleaned from the environment. The output of each individual motor schema is a velocity vector representing the direction and speed at which the robot is to move given current environmental conditions. A partial listing of some of the available motor schemas for our robot include (see also *Fig. 1*):

- **Move-ahead** – Move in a general compass direction.
- **Move-to-goal** – Move towards a discernable goal.
- **Avoid-static-obstacle** – Move away from a detected barrier.
- **Stay-on-path** – Move towards the center of a path.

- **Dock** – Combine aspects of ballistic and controlled motion to achieve a safe trajectory for mating with a docking workstation.
- **Noise** – A random process used for handling problems with local minima.
- **Move-up** – Move uphill on undulating terrain.
- **Move-down** – Move downhill on undulating terrain.
- **Maintain-altitude** – Follow isocontours on undulating terrain.

The first six schemas listed above have also been formulated for three-dimensional navigation as might be found in aerospace or undersea robotic applications (Sec. 5.2).

Each of these schemas is instantiated as separate asynchronous computing agents with parameters reflecting current world knowledge. The computations for each schema are very simple, usually involving a couple of additions or subtractions and at most one or two multiplications or divisions (with the exception of docking which includes a transcendental function). It should be noted that the entire potential field is never computed by the robot (although it is depicted in the figures to aid the reader's comprehension). Only the point where the robot is currently located needs to be computed. So each process is performing relatively simple mathematical operations, outputting a single vector expressing the robot's desired motion for that behavior.

The output of each primitive motor schema is combined using vector summation and normalization (keeping the resultant vector within the constraints of the actual robot's capabilities). This simple process can result in quite complex trajectories and behaviors as illustrated in the simulations shown in *Fig. 2*.

3.1.3. Perceptual Schemas

Action-oriented perception is the founding tenet for perceptual schema construction and usage. Each individual perceptual strategy is created to produce *only* the information that is necessary for the particular task at hand. Perceptual schemas are embedded within motor schemas providing the information that is required for them to compute their reaction to the world. In contrast to other non-reactive navigational approaches, no abstract model of the world is built using sensory data which is then reasoned over by perceptual processes during plan execution.

With this technique it becomes possible to exploit expectation-based mechanisms by using *a priori* knowledge of objects, constraints on positional uncertainty obtainable from a spatial uncertainty map [7], and adaptive models based on processing performed by previous sensing. By matching perceptual techniques to motor requirements, computational demand is greatly reduced.

Current sensors for our mobile vehicle include 24 ultrasonic sensors, a monocular CCD video camera, and shaft encoders. Simulation studies have used inclinometer data [12]. The example below using vision for docking operations describes the coordination between multiple perceptual strategies.

Docking with a manufacturing workstation is a complex operation. It requires a wide range of perceptual skills and knowledge (See [11] for a more detailed description of the perceptual processing). When the robot is located at a large distance from the workstation, it is impossible to discern the dock's structure. A salient feature however is the presence of motion (activity) due to the workstation's normal operation. A temporal activity detection algorithm has been designed to provide information regarding the location of the workstation relative to the robot given constraints on the location of the robot relative to a world map. As the robot gets closer to the workstation, it becomes necessary to positively identify it. A more computationally expensive algorithm, exploiting a spatially constrained version of the Hough transform is used to get a positive location. For final positioning, adaptive tracking is performed which abandons the *a priori* model of the workstation after it has been located and uses feedback from previous images to finally position the robot relative to the dock. An adaptive version of a fast region segmentation routine followed by texture-based positioning completes the sequence of perceptual techniques for docking in a complex manufacturing setting.

The above sequence can be likened to the strategies used by someone who is given instructions to turn right at the second flashing traffic light. At first the person walks along looking for some long-distance perceptual event. After detection of a candidate event (something flashing), models of what traffic lights look like (as opposed to car's turn signals) are brought to bear for a positive identification of the flashing object as a

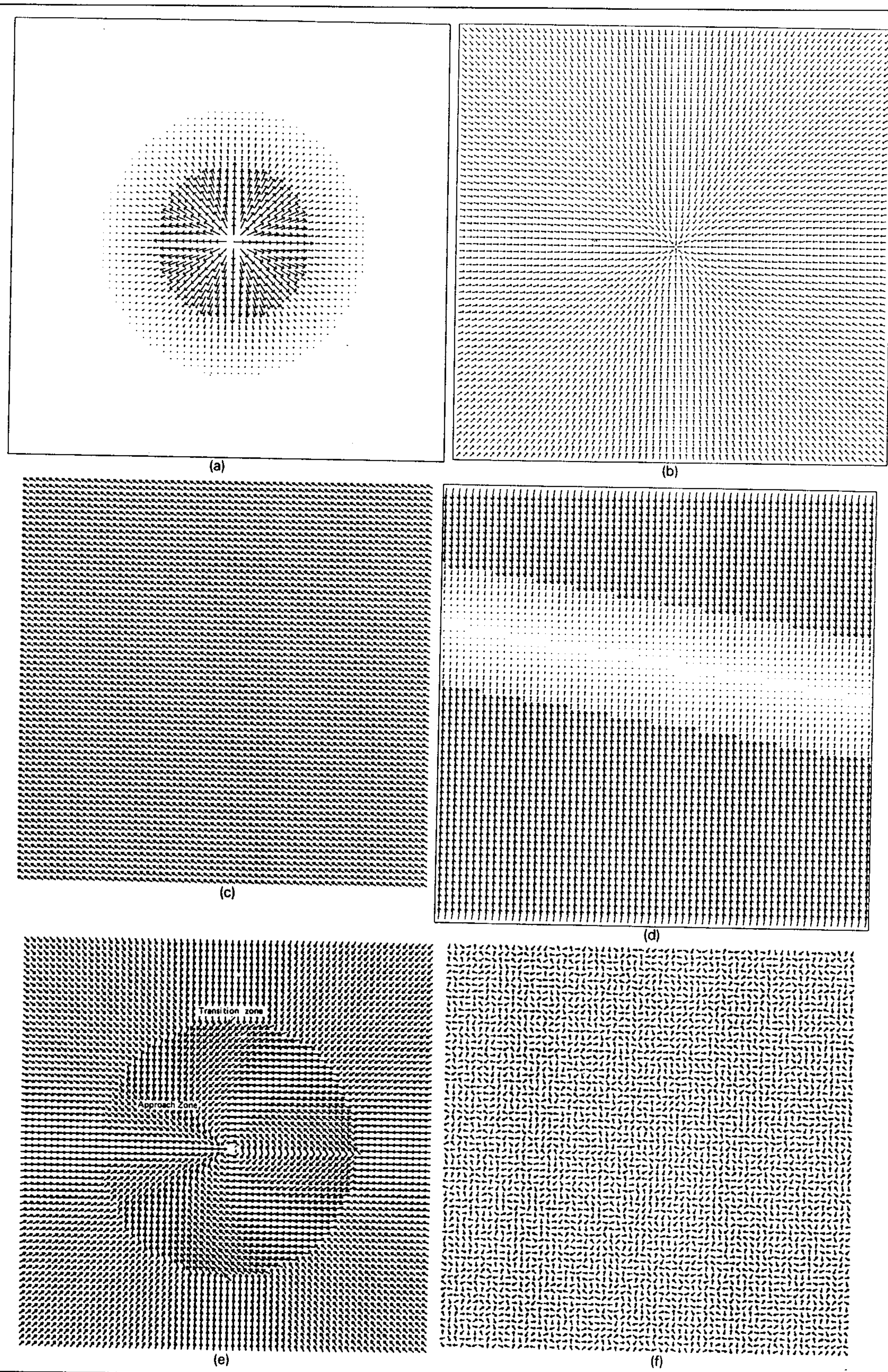


Fig. 1. Motor Schemas. a) **Avoid-static-obstacle**; b) **Move-to-goal**; c) **Move-ahead**; d) **Stay-on-path**; e) **Dock**; f) **Noise**.

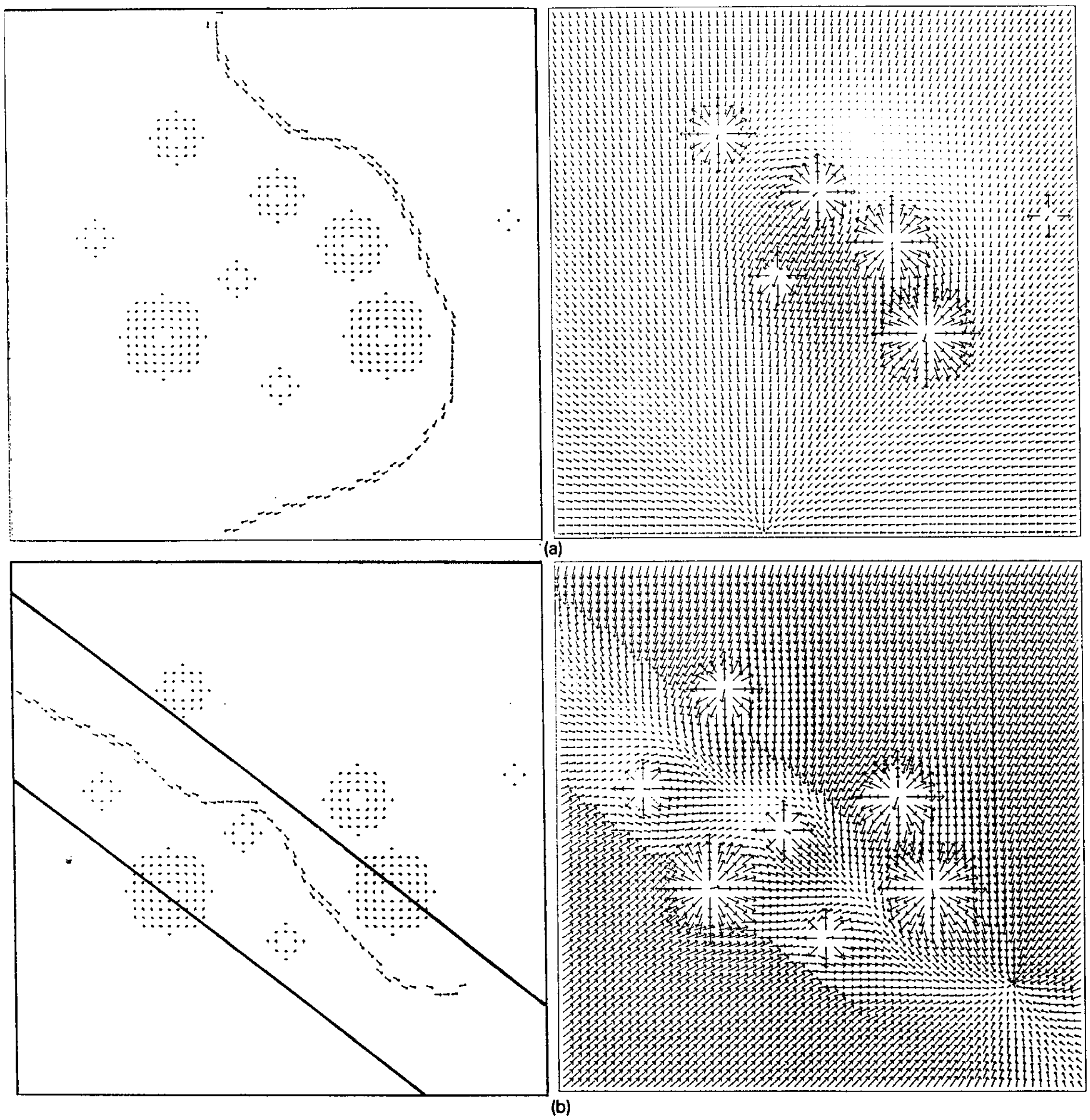


Fig. 2. Simulated robot trajectories through complex worlds. Figures to the left show the path of the robot through the world towards the goal. Figures to the right indicate a typical field present at some point during the robot's travel. a) **Move-to-goal** and 9 obstacles. Only 6 **avoid-static-obstacle** schemas are active in the field to the right as the other obstacles are out of range. b) **Move-to-goal**, 8 obstacles, and **stay-on-path** schemas. Only 7 **avoid-static-obstacle** schemas are active in the right-hand figure.

traffic light. Once this is identified, the preconceived notion of an *a priori* model of a traffic light is abandoned, allowing the user to continuously monitor the perceived object in an adaptive manner, updating their understanding (acquired model) of what that object is only in terms of what is necessary to perform the motor action of turning at the correct point.

This strategy is loosely consistent with the concept of affordances as espoused by Gibson [20]. Only the information that is required to perform a motor action is extracted from the environment and little or nothing more. This computational economy underscores the principle that perception is meaningless without the context of motor action.

Some of the perceptual schemas that we have developed in addition to the four described above include: ultrasonic sensing used for obstacle avoidance; fast region segmentation using computer vision for road/path following; fast line finding for landmark identification and road following; depth-from-motion algorithm for obstacle avoidance; inclinometers for moving on non-planar outdoor terrain; and the use of shaft encoders for direction and approximate goal location.

3.2. World Knowledge

Most reactive systems are unconcerned with the use of world knowledge. It is our contention that world knowledge plays a vital role in a robot's interaction with the world. It is not a prerequisite for navigation, but it is a prerequisite for *efficient, flexible, and generalizable* intelligent navigational techniques.

Two types of world knowledge can be utilized. *A priori* information about the robot's environment that can be considered relatively static for the duration of the mission is termed *persistent* knowledge. This data typically arises from object models of things the robot might expect to see within it's world, models of the free space within which it navigates, and an ego-model of the robot itself. The knowledge base within which this information resides is termed long-term memory (LTM), indicative of the persistence of this data.

Transitory knowledge is dynamically acquired by the robot as it moves through the world. It is remembered within the context of short term memory (STM). World models constructed from sensory data fall into this category. Although this data is not used for reactive navigational control, it is brought to bear when difficulties are encountered with the reflexive/reactive techniques used in the absence of this form of knowledge. Much work has been undertaken in dynamic world model acquisition in mobile robotics (e.g., [13,24]). It should be noted that dynamically acquired world models should only be used when needed, as indicated by the failure of reactive control to cope with difficult situations. Even then, that data should only be used to reconfigure the reactive control regime and not to supplant it. Transitory knowledge is forgotten (fades) as the robot moves away from the locale within which that information was gathered. Learning mechanisms could be constructed to migrate knowledge from STM to LTM (assuming that semantics could be developed to distinguish persistent environmental objects from transitory ones) but this currently is not one of our research thrusts.

For both persistent and transitory knowledge, the choice of representational structure and format is less important than merely the availability of the knowledge itself for use within a navigational system using reactive control for plan execution. Persistent knowledge allows for the use of pre-conceived ideas of the relationship of the robot to the world, enabling more efficient use of its resources than would be accomplished otherwise. Transitory knowledge on the other hand, if misused, could interfere, with the simplicity and efficiency of reactive control. Nonetheless, when difficulties with a reflexive control regime arise, it is important to have a bigger picture available to help resolve them. This can result in solutions to problems such as the *fly-at-the-window* situation in reactive control when an insect, striving to go towards the light of the sun entering from the outside of a window, but rebuffed by the glassy barrier, expends all of its energy trying to solve the problem with its fixed set of behaviors and dies. If transitory models of the environment are constructed under these conditions, a robot could use this information to circumnavigate the barrier.

4. The Autonomous Robot Architecture

The Autonomous Robot Architecture (AuRA) was designed to provide general purpose navigational capabilities over a wide-range of problem domains. It has been used as the framework to conduct navigational experiments in the interior of buildings [9], outdoor campus settings [5], and manufacturing environments [6]. Simulation studies in three dimensional domains [8,29] and rough outdoor terrain [12], have also been performed.

AuRA (*Fig. 3*) is comprised of 5 basic subsystems:

1. *Perception:* The gateway for all sensory data into the system. Sensory information is shunted in two primary directions: to the motor schema

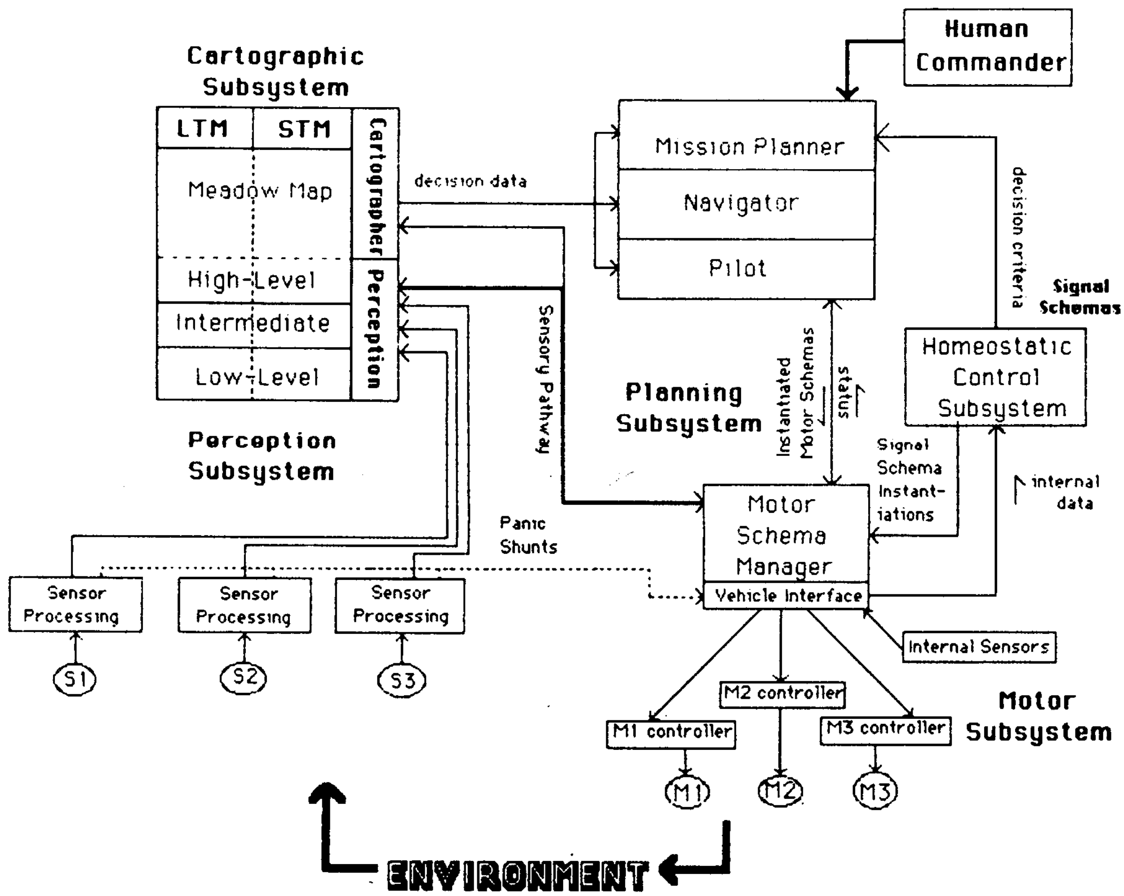


Fig. 3. The Autonomous Robot Architecture.

manager for processing by the perceptual schemas providing data for the reactive motor behaviors; and to the cartographer for construction of dynamically acquired world models within short-term memory. Specific sensor preprocessing and information flow control occurs within the perception subsystem.

2. *Cartographic*: World knowledge, both *a priori* knowledge stored in long-term memory and perceived world models stored in short-term memory, is constructed and maintained within this subsystem. It is available for use by the planning subsystem. Spatial uncertainty management is also maintained within the cartographic subsystem, providing data for the generation and control of expectations used for perceptual processing.
3. *Planning:* The planning subsystem consists of both a hierarchical planner and a distributed reactive plan execution subsystem. The hierarchical planner determines, via a series of actions involving changes in planning scope, first a global path through the modeled world (*Fig. 4a*) that consists of a series of piecewise linear path segments (*Fig. 4b*) that fit the constraints of the overall mission. Each segment is translated sequentially into a collection of motor and perceptual schemas that will accomplish the subtask (*Fig. 4c*). These schemas are then instantiated concurrently for plan execution, driving the robot to successful completion of each leg (*Fig. 4d*).
4. *Motor:* The motor subsystem is the interface to the actual robot. It is intended to be the only

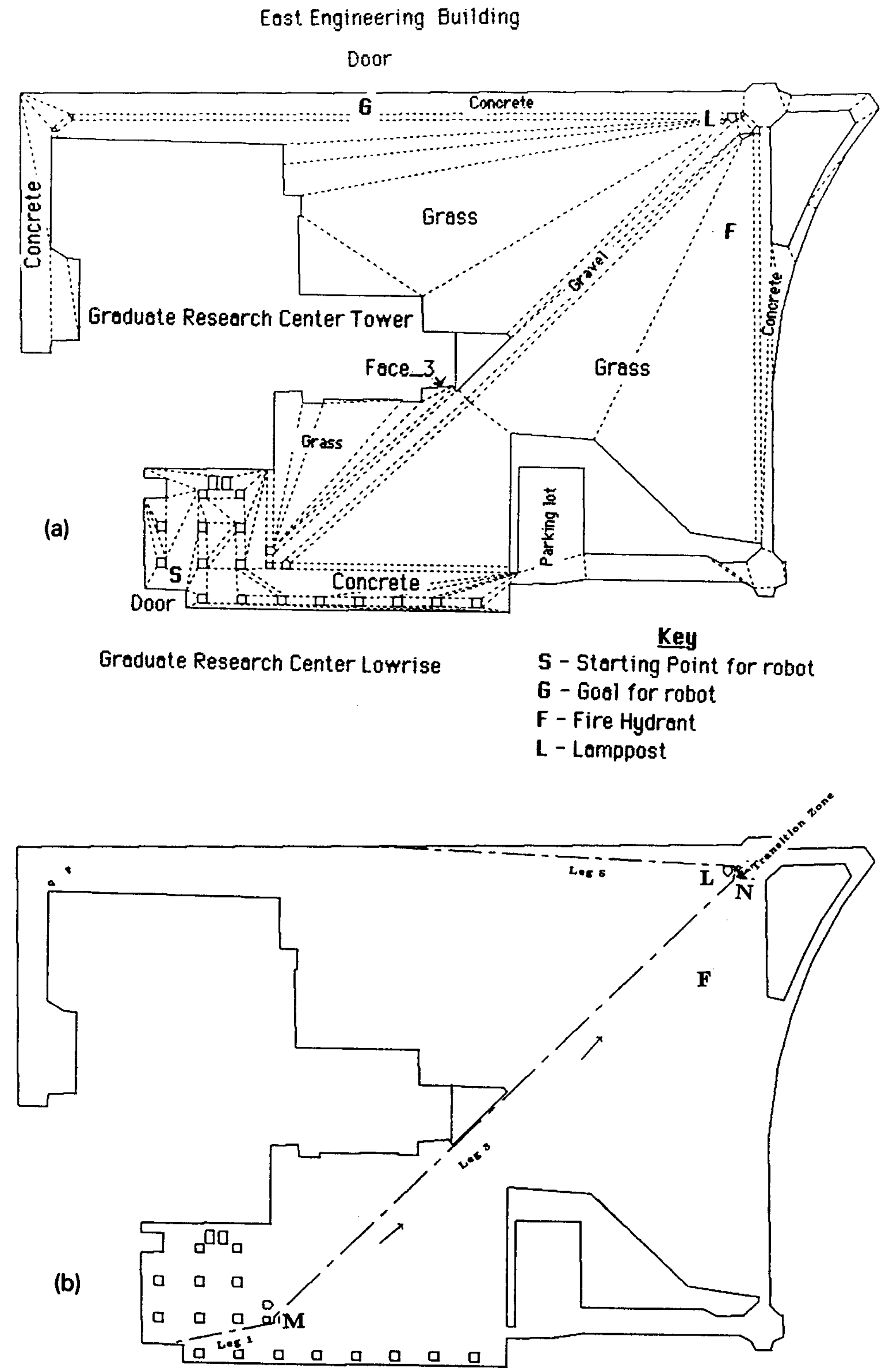


Fig. 4. Planning in AuRA. (a) (*top*) World model of a campus setting. (b) (*bottom*) Navigational path computed through world model.

A. **Stay-on-path**(**find-path**(gravel))
B. **Move-ahead** (NNE — 30 degrees)
C. **Move-to-goal**(right(**find-landmark**(LAMPPOST-107),3))
D. **Move-to-goal**(**find-transition-zone**(gravel,concrete))
E. **Find-landmark**(HYDRANT-2)
F. **Find-landmark**(GRC-TOWER(face-3))
G. **Avoid-obstacles**

(c)

(d)

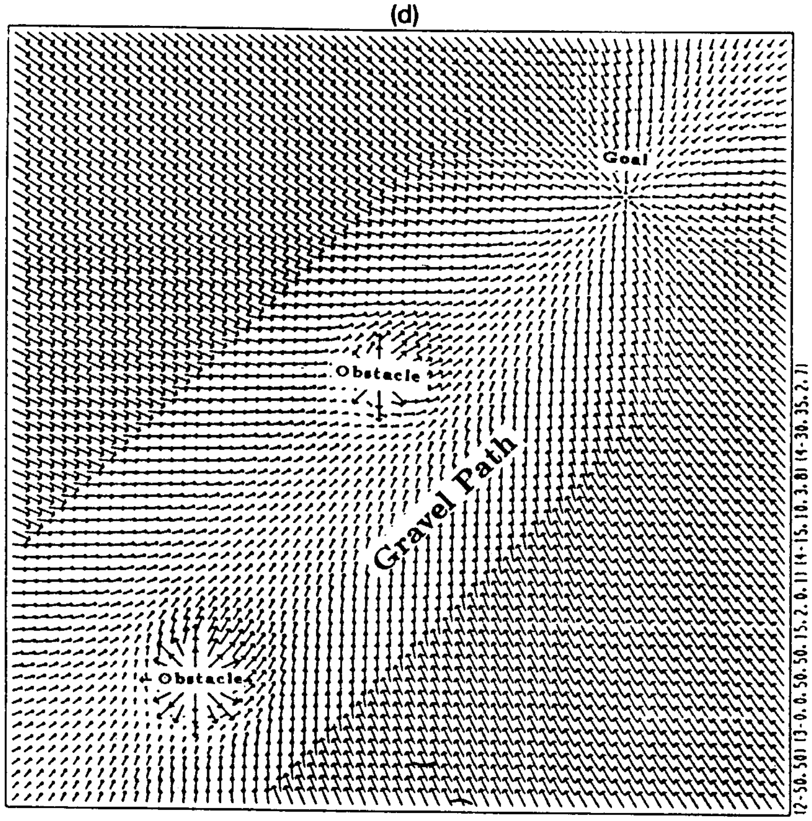


Fig. 4 (continued). (c) (*top*) Collection of behaviors to traverse leg 3 of Fig. 4b. (d) (*bottom*) Example of a potential field generated during path traversal where the robot is approaching the goal in the presence of unmodeled obstacles.

vehicle dependent component of the architecture. The velocity signal provided by the planning subsystem is translated into a correct set of steering and drive commands for the particular robot, in our case, George – a Denning mobile robot (*Fig. 5*).

5. *Homeostatic Control:* This subsystem monitors internal conditions of the robot (e.g., available fuel, internal temperature) and feeds the data to both the higher level planning mechanisms and the motor schemas themselves. Knowledge of the robot's current condition can thus affect high-level planning and also modify reactive behavior in a manner that can minimize the internal stress on the system. Schema-based techniques are employed that exploit an analog of the mammalian control system to allow for dynamic replanning in hazardous environments. See [10] for additional information.

AuRA in its current state is only partially completed. Most of the modules exist in operational form, but are yet to be fully integrated. Section 5 presents examples of the navigational successes already achieved within the existing framework.

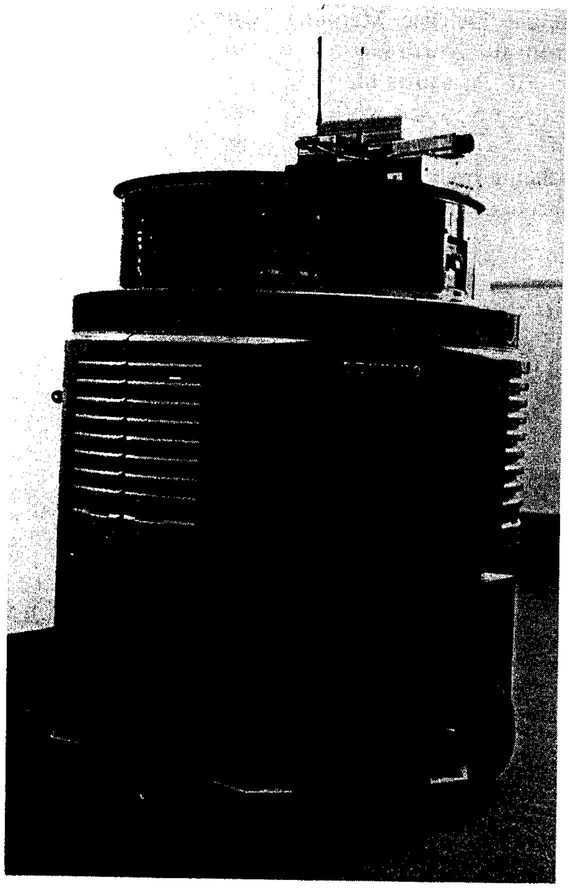


Fig. 5. George – a Denning mobile robot.

5. Reactive Navigational Examples

Section 3.1 has already presented simulation results for two-dimensional terrestrial schema-based navigation. In this section, results using our actual mobile robot are presented in addition to simulation studies in more complex worlds.

5.1. Mobile Robot Experiments

The concepts in this paper have been developed and tested on our mobile robot. The first sequence (*Fig. 6*) illustrates door entry behavior (using an earlier Denning robot – HARV) by a combination of **move-ahead** and **avoid-static-obstacle** schemas. 24 Ultrasonic sensors provide the perceptual data for the robot. The **move-ahead** schema is pointing obliquely into the wall, while the **avoid-static-obstacle** is concurrently repulsing the robot from the wall. The net emergent effect has been termed the *drunken sailor behavior*, where the robot follows the contour of the wall (as someone drunken might lean against the wall while progressing forward) and when the robot (or the sailor) encounters a sufficiently large opening, falls through. World knowledge is exploited to indicate when a door might be present without relying on the shaft encoders or other correlation mechanisms to find its position accurately. The result is successful passage through a doorway.

Docking in a complex, cluttered environment is shown in *Fig 7*. Here the robot is under the influence of **docking**, **avoid-static-obstacle**, and **noise** schemas. The robot never computes a global path, but continuously reformulates its reactions to the world (using ultrasonic and shaft encoder data), to maneuver around obstacles and satisfy its final positioning requirements. We have also demonstrated the robot's ability to cope with moving obstacles using the same schema configuration for this domain.

5.2. Three-dimensional Simulation

Schema-based navigation has been extended to three-dimensional domains [8] that afford six degrees of freedom for the robot (3 translational, 3 rotational). The schemas used for 2D navigation have been reformulated in a straightforward manner to produce three dimensional vectors instead of two. As the simulation studies show, the navigational technique generalizes very well.

The first example (*Fig. 8a*) shows the robot's path while maintaining its position in a channel, moving towards a goal and avoiding obstacles along the way. The second example (*Fig. 8b*) illustrates docking in the presence of obstacles. Remember that a global path is never computed and the robot reacts from its current position to its perceived world using combinations of schemas that have been selected by higher level planning mechanisms. The simulation studies incorporate uncertainty in perception to reflect more realistically actual conditions.

5.3. Reactive Navigation over Contoured Terrain

Rough terrain can also provide an interesting test domain for schema-based navigation. Using inclinometer data that provides pitch and roll information for a mobile robot, the vehicle can react directly to the topography of the land. This can result in literal artificial intelligence "hill-climbing" behavior. A discussion of the issues of sensor design and the role of noise in dealing with local maxima and minima appears in [12]. The simulation studies presented here are based on actual Defense Mapping Agency data obtained from the North Georgia region.

Fig. 9a shows the results for a robot striving to move up in the world and using noise to rock it off of local maxima. The figure to the right of the schema path shows the robot's altitude as it move through the world. *Fig. 9b* shows a combination

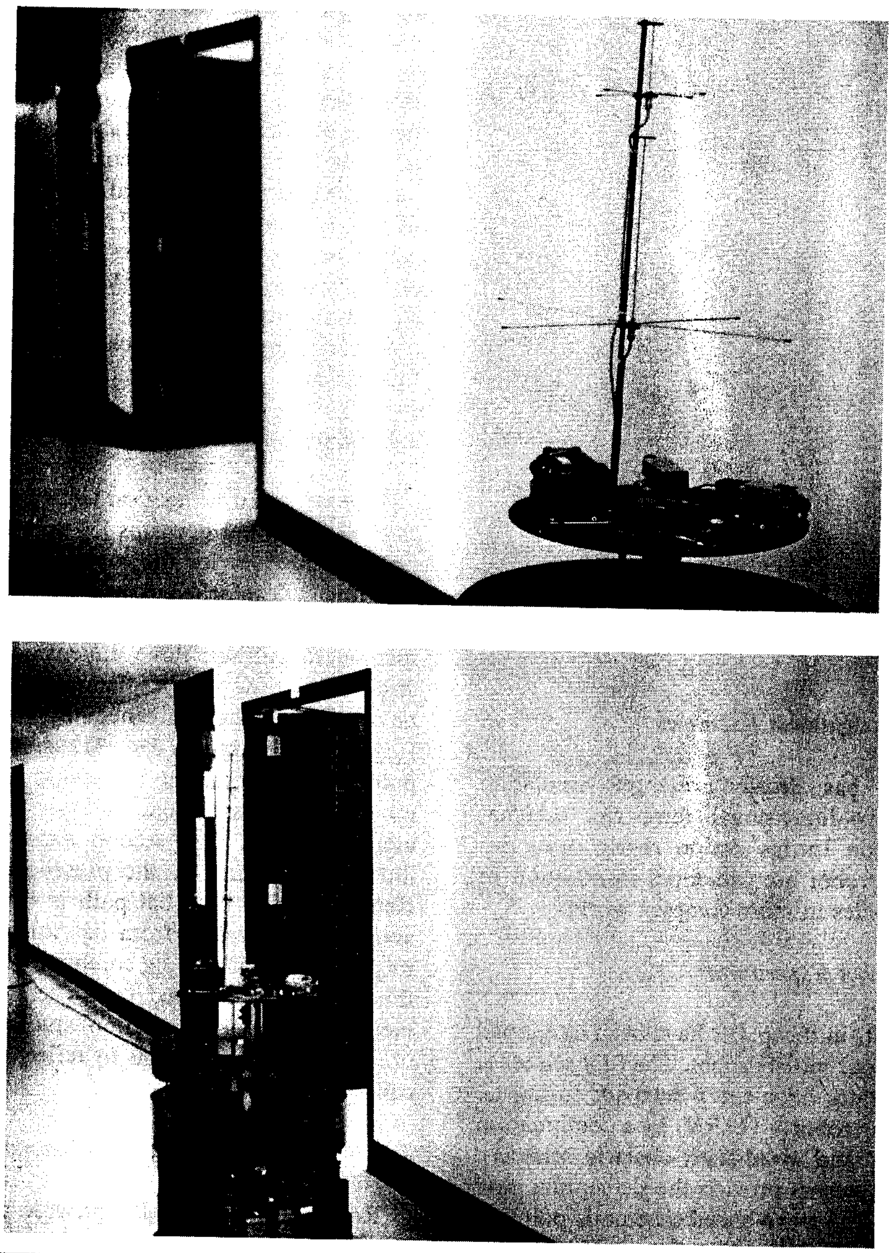


Fig. 6. Door entry via a combination of **avoid-static obstacle schema** and **move-ahead** schemas.

of **move-down** and **move-to-goal** schemas successfully completing the robot's mission. Space prevents illustration of other interesting combinations of inclinometer behaviors (including **maintain-altitude** with other schemas such as **avoid-static-obstacle**, **move-ahead**, and **stay-on-path** (see [12]).

6. Conclusions

Reactive navigation is an important general purpose navigational technique that can be applied to a wide variety of problem domains. It is characterized by its independence of world models

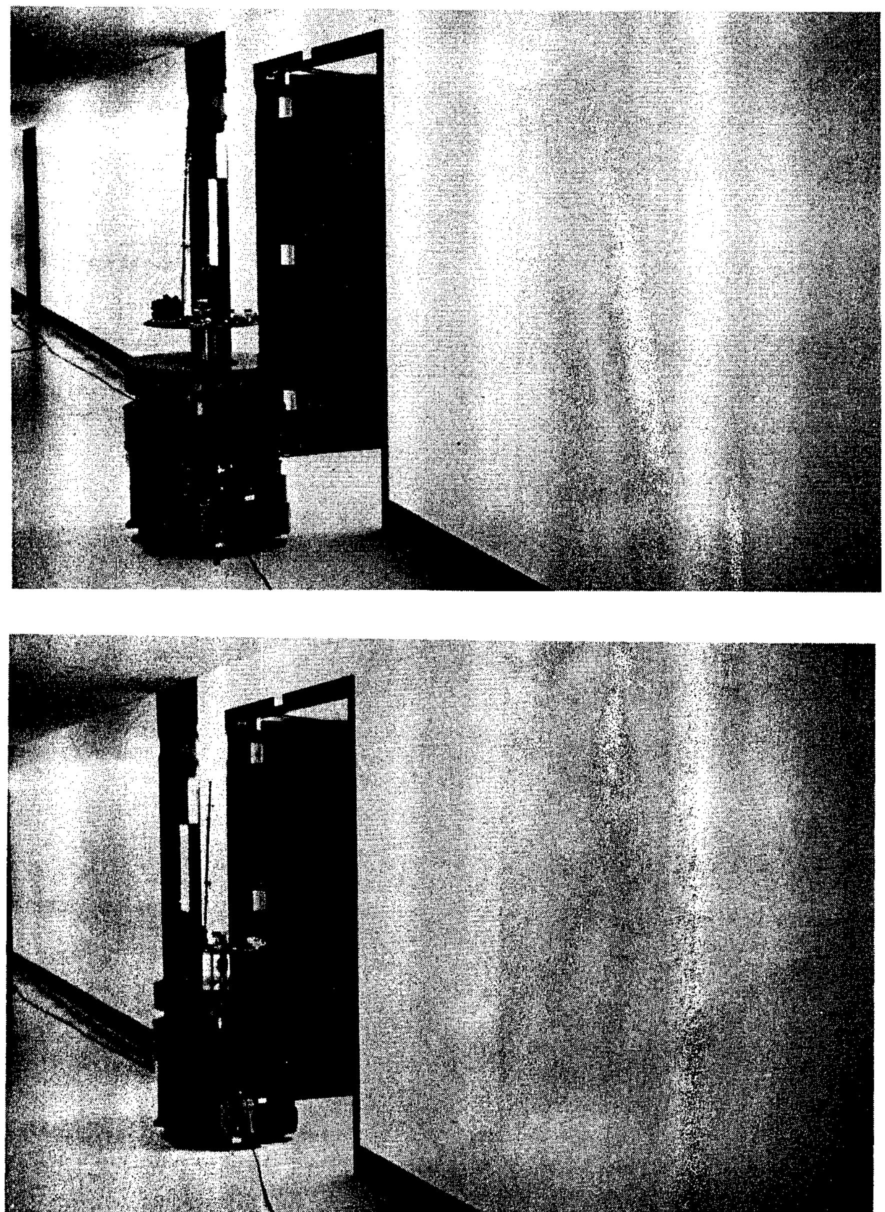


Fig. 6 (continued).

during plan execution. Knowledge of various forms, however, plays a crucial role for flexible and generalizable reactive navigation. Flexibility can be readily incorporated by modularizing behavioral patterns and perceptual strategies (schemas within AuRA). In this manner, behaviors drawn from a library of motor skills and sensory algorithms can be configured to meet the needs of a particular high level mission and any known environmental constraints.

World models play an important role in configuring these behaviors. We have seen that world models are unnecessary for low-level actions, but in order to efficiently explore the environment, *a*

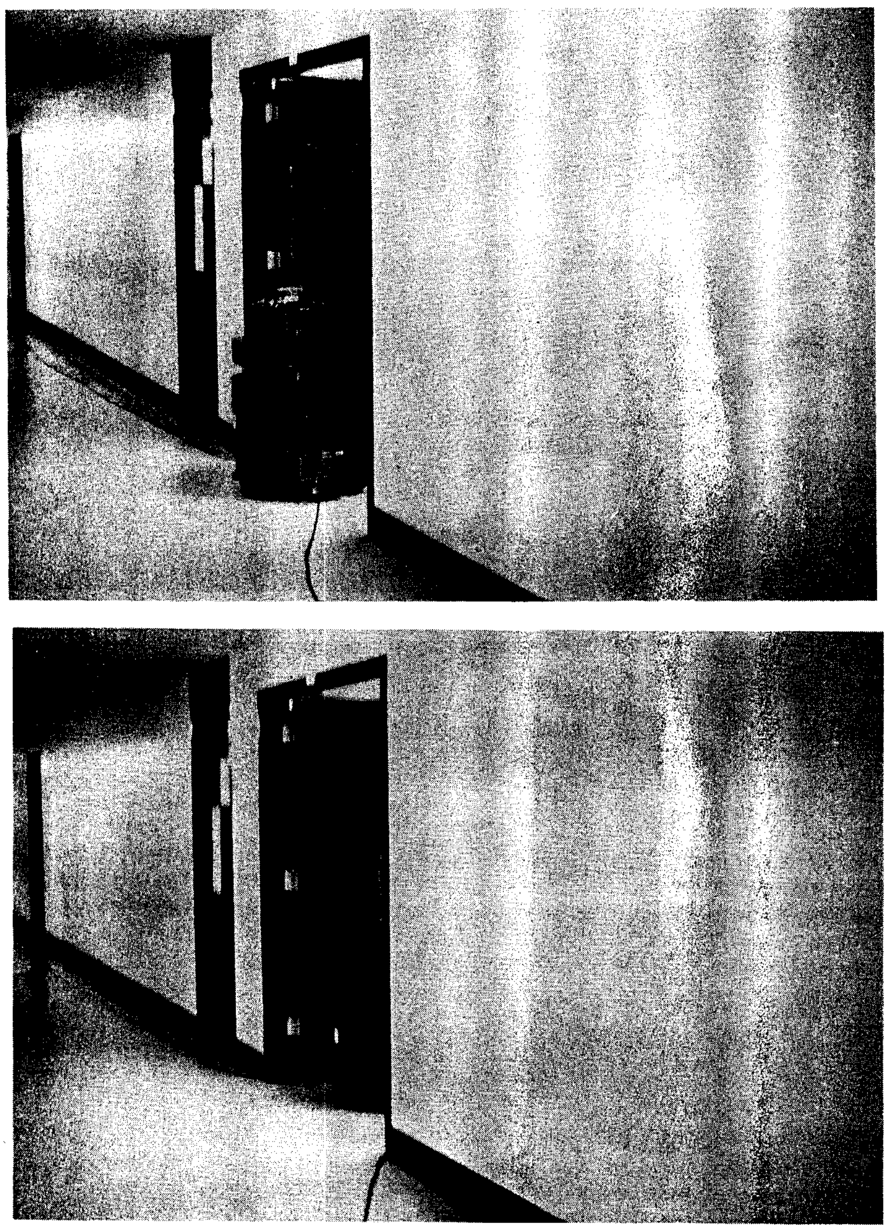


Fig. 6 (continued).

Fig. 7. Docking in a complex environment. The robot winds its way through a world cluttered with obstacles ultimately assuming the correct position and orientation relative to the docking workstation.

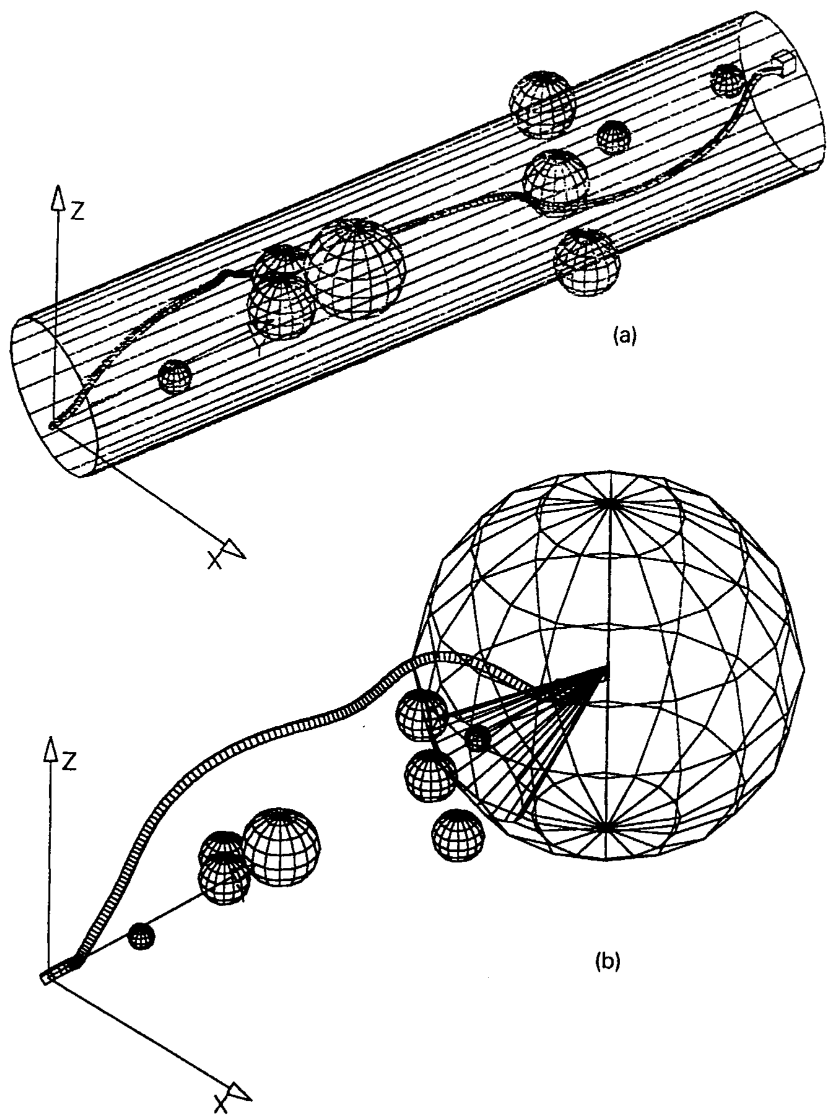


Fig. 8. Two examples of 3D schema-based navigation. a) **Move-to-goal**, **stay-in-channel**, and **avoid-static-obstacle** schemas produce the robot's path through the cluttered channel to the goal on the right. b) **Docking** and **avoid-static-obstacle** schemas produce a safe traversal through the cluttered environment to the dock in the center of the controlled motion zone. The outer sphere of the **docking** demarcates the transition from ballistic to controlled motion, while the inner cone distinguishes the approach zone from the coercive region.

priori knowledge of both objects to be perceived and the navigational free-space of the robot should be exploited when available. Dynamically acquired world models also play a role, but only when problems are encountered using reactive techniques. Routine navigation can be conducted in the absence of such short-term memory data.

The Autonomous Robot Architecture is one framework in which world knowledge is used to facilitate reactive navigation. Potential field meth-

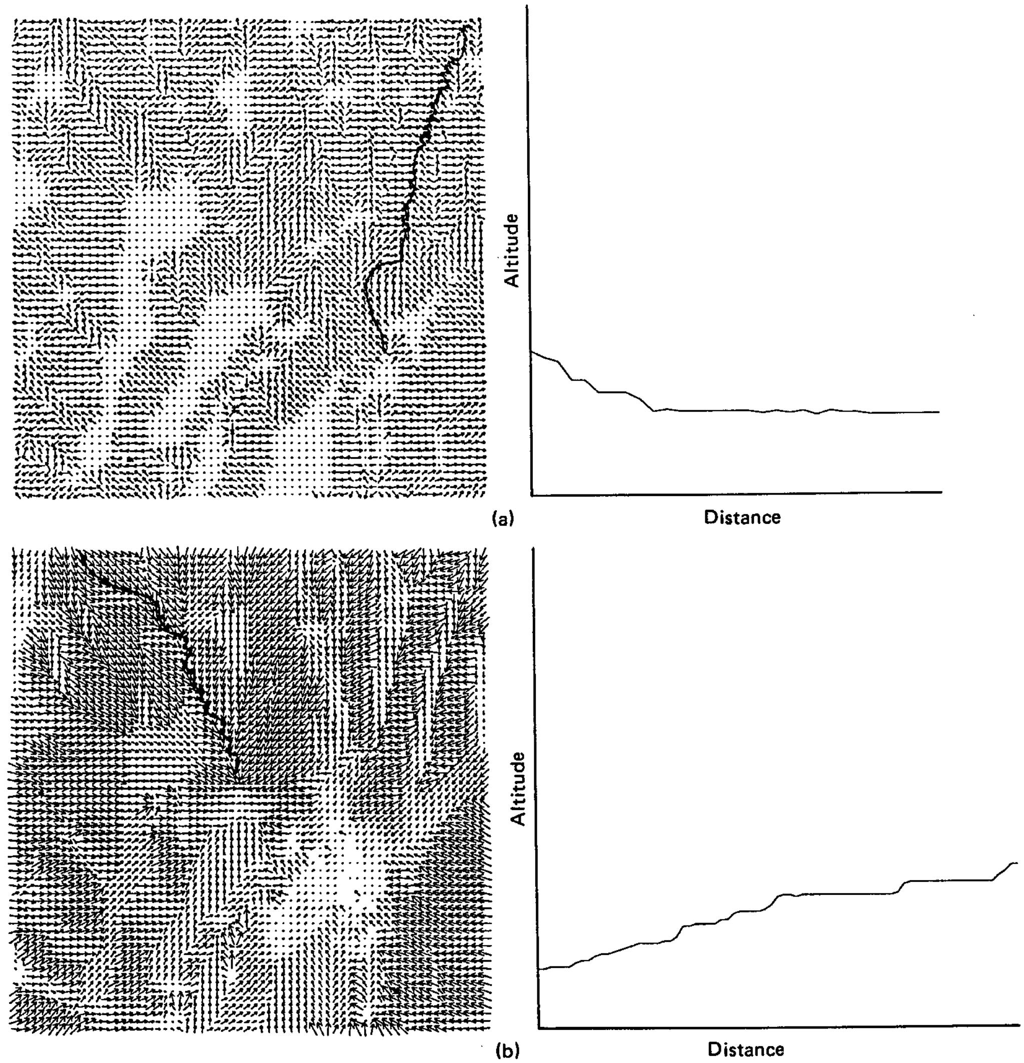


Fig. 9. Inclinometer schema-based navigation. a) **Move-up** and **noise** schemas produce the steady climb over rough terrain. b) **Move-down** and **move-to-goal** schemas produce the path shown. The robot overcomes local minima due to the pressure exerted on it by the goal.

ods afford an effective means for combining multiple active motor behaviors concurrently, without arbitrating between them. The simulations and experiments in this paper have demonstrated the viability of this methodology.

Acknowledgments

The author would like to thank Robin Murphy and Warren Gardner for their assistance with the experiments and simulations. This research is supported in part by the Georgia Tech Computer Integrated Manufacturing Systems Program and the Materials Handling Research Center.

References

[1] P. Agre and D. Chapman, Pengi: An implementation of a theory of activity, Proc. AAAI-87, 268-272.

[2] S. Amarel, On representations of problems of reasoning about actions, *Mach. Intelligence* 3 (1968) reprinted in *Readings in Artificial Intelligence*, Eds. Webber and Nilsson (Tioga, 1981) 2-22.

[3] M. Arbib, Perceptual structures and distributed motor control, *Handbook of Physiology – The Nervous System II*, Ed. Brooks (1981) 1449-1465.

[4] M. Arbib, T. Iberall and D. Lyons, Coordinated control programs for movements of the hand, COINS Technical Report 83-25 Dept. of Comp. and Info. Science, University of Massachusetts, Amherst, Ma., 1983 (also *Exp. Brain Res. Suppl.*, 10, (1985) 111-129).

[5] R.C. Arkin, E. Riseman and A. Hanson, AuRA: An architecture for vision-based robot navigation, *Proc. 1987 DARPA Image Understanding Workshop* (Los Angeles, CA, 1987) 417-431.

[6] R.C. Arkin, Intelligent mobile robots in the workplace: Leaving the guide behind, *Proc. First International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems* (1988) 553-561.

[7] R.C. Arkin, Spatial uncertainty management for a mobile robot and its role in expectation-based perception, *Robot Control 1988 (Syroco '88)*(Pergamon Press, Karlsruhe, W. Germany, Oct. 1988) 279-284.

[8] R.C. Arkin, Three dimensional motor schema based navigation, *Proc. NASA Conference on Space Telerobotics*, Pasadena, CA (Jan. 1989).

[9] R.C. Arkin, Motor schema-based mobile robot navigation, *Int. J. Robotics Res.* 8 (4) (August 1989) 92-112.

[10] R.C. Arkin, Dynamic replanning for a mobile robot based on internal sensing, Proc. 1989 IEEE International Conference on Robotics and Automation, Scottsdale, Arizona, (May 1989) 1416-1421.

[11] R.C. Arkin, R. Murphy, M. Pearson and D. Vaughn, Mobile robot docking operations in a manufacturing environment: Progress in visual perceptual strategies, *Proc. IEEE International Workshop on Intelligent Robots and Systems 89*, Tsukuba, Japan, 147-154.

[12] R.C. Arkin and W.F. Gardner, Reactive inclinometer-based mobile robot navigation, submitted to 1990 IEEE Conf. on Robotics and Automation, May 1990.

[13] N. Ayache and O. Faugeras, Building, registration, and fusing noisy visual maps, *Proc. Int. Conf. on Computer Vision* (1987).

[14] F.C. Bartlett, *Remembering: A Study in Experimental and Social Psychology* (London, Cambridge Univ. Press, 1932).

[15] R. Brooks, A robust layered control system for a mobile robot, *IEEE J. Robotics Automation* RA-2 (1) (1986) 14-23.

[16] R. Brooks, Intelligence without representation, Research paper, AI lab, Mass. Institute of Tech. (1988).

[17] R. Brooks, A robot that walks; Emergent behaviors from a carefully evolved network, *Proc. 1989 IEEE Conf. on Robotics and Automation*, Scottsdale, AZ. (May 1989) 692-694.

[18] R. Brooks and J. Connell, Asynchronous distributed control system for a mobile robot *Mobile Robots*. Eds. W. Wolfe and N. Marquina, *Proc. SPIE* Vol. 727 (Bellingham, Wash. SPIE) 77-84.

[19] R.J. Firby, Adaptive execution in complex dynamic worlds, *Ph.D. Dissertation*, YALEU/CSD/RR #672, Yale University, Jan. 1989.

[20] J.J. Gibson, The theory of affordances, *Perceiving, Acting, and Knowing*, Ed. R. Shaw and J. Bransford (Erlbaum, 1977).

[21] H. Head and G. Holmes, Sensory disturbances from cerebral lesions, *Brain* 34 (1911) 102-254.

[22] M. Kadonoff, F. Benayad-Cherif, A. Franklin, J. Maddox, L. Muller and H. Moravec, Arbitration of multiple control strategies for mobile robots, Mobile Robots - *SPIE Proc.* 727 (1986).

[23] L. Kaelbling, An architecture for intelligent reactive systems, *SRI Technical Note 400*, SRI International (Oct. 1986).

[24] H. Moravec, Sensor fusion in certainty grids for mobile robots, *AI Magazine* 9 (2) (1988).

[25] U. Neisser, *Cognition and Reality: Principles and Implications of Cognitive Psychology* (Freeman, 1976).

[26] D. Norman and T. Shallice, Attention to action: Willed and automatic control of behavior, from *Consciousness and Self-regulation: Advances in Research and Theory*, Eds. Davidson et al., 4 (Plenum, 1986) 1-18.

[27] D. Payton, An architecture for reflective Autonomous vehicle control, *IEEE Conf. Robotics and Automation* (1986) 1838-1845.

[28] J. Piaget, *Biology and Knowledge* (Univ. of Chicago Press, 1971).

[29] T.M. Rao and R.C. Arkin, 3D path planning for flying/crawling robots, to appear in *Proc SPIE Conference on Mobile Robots IV*, Advances in Intelligent Robotics Systems, Philadelphia, PA, November 1989.

[30] B.F. Skinner, *About Behaviorism* (Knopf (Random House), 1974).