James L. CROWLEY
LIFIA (IMAG)

Navigation and World Modeling for a Mobile Robot: a Progress Report

Abstract

Factory robots function in highly engineered environments and thus can be made to function with very limited sensing and intelligence. The introduction of robots into such unconstrained environments such as a construction site or a mine requires a radically different approach. Such systems must sense and model their environment and base their actions on such a model. They must also apply intelligence so that they do not simply repeat actions, but accomplish goals.

This paper presents recent progress towards the development of mobile robots which intelligently accomplish goal oriented behavior. A primary contribution is the introduction of an architecture which permits a robotic system to apply knowledge to the coordination of action and perception. Hierarchical architectures are presented for both perception and navigation.

A discussion of recent developments in navigation and perception are presented. Among the most important points is the presentation of a general purpose vehicle-level control protocol. This protocol defines a virtual vehicle which can permit navigation systems to be easily transported between vehicles. The problems of estimating the uncertainty in a vehicle's estimated position, and recent advances in composite modeling are also discussed.

Résumé

Cet article présente les progrès récents dans le développement des robots mobiles capables de comportement intelligent dirigé par un but. Sa contribution essentielle est la description d'une architecture logicielle qui permet au système robotique de coordonner action et perception en utilisant la connaissance. Cette architecture se décompose en deux architectures hiérarchiques pour le contrôle de la perception et des actions.

L'article résume les dévelopements récents en navigation et en perception. Parmi les thèmes abordés, un point important est la présentation d'un protocole de contrôle pour le niveau véhicule. Ce protocole définit un véhicule virtuel qui facilite le transport des systèmes de navigation entre véhicules. L'article traite également les problèmes d'estimation de l'incertitude de la position estimée du véhicule ainsi que les avancés récentes en matière de modélisation composite.

1 Introduction

At the 1984 International Conference on Robotics in Construction the author presented a description [Crowley 84f] of research in progress towards a mobile surveillance robot, and described plans for the application of the resulting system to navigation in a mine environment and in an out-door environment. The project described at that time was the third of a series of systems for autonomous navigation and world modeling using ultrasonic ranging. At the 1985 conference a paper was presented [Whittaker et al 85] which described some aspects of the completed world modeling system as well as results of an experiment with navigation at a demonstration mine at Pittsburgh in March 1985. This paper continues that series with a report on the system architecture that has evolved, and a discussion of lessons learned in navigation and in composite modeling.

The purpose of this paper is to communicate the architectural framework that has evolved, as well as specific lessons in navigation and perception. Because of space limitations, technical details will be limited to a minimum necessary to communicate these points. Readers interested in technical details are referred to references [Crowley 85c] (In English) and [Crowley-Coutaz 85] (in french), either of which may be obtained from the author at the address listed above.

1.1 Background

The architecture presented in this paper has evolved over a series of projects in mobile robot navigation and vision that began with an effort to develop a low-cost Intelligent Mobile Platform (or IMP) for Commodore Business Machines. The IMP was based on a rotating ultrasonic ranging device [Crowley 85c], which provided 120 range measurements per rotation. The first version of the IMP system was developed using a mobile robot simulator, which proved to be an enormous benefit for experimenting with the system architecture and debugging specific software modules. The transition to IMP hardware demonstrated a number of weaknesses in specific modules, but at the same time validated the overall architecture. In particular, complete redesign of techniques within a module was easily accomplished with little or no change to the system architecture.

As the development of the IMP was nearing completion, work began on the development of a similar system for a mobile surveillance robot. A major innovation with the surveillance robot was the replacement of the rotating ultrasonic ranging device with a ring of 24 stationary sonar ranging devices. This large decrease in the density of range measurements dictated the development of a new approach to interpretation of the ultrasonic range data, and a heavier reliance on a pre-learned model of the limits of freespace. In the course of developing this system, new techniques were developed for automatically learning a model of the world, for avoiding obstacles, for wall following, for local navigation and for localisation. The best paper to date describing the system developed for the surveillance robot is [Crowley-Coutaz 85].

In March 1985, a sonar ring was placed on the Terragator Mobile robot, and the navigation and world modeling systems developed for the surveillance robot were demonstrated in a mine environment at the US Bureau of Mines demonstration facility at Pittsburgh [Whittaker et. al. 85]. More recently this software system has been re-implemented independently at IIRIAM, running with a simulator under VAX/VMS, and at LIFIA running as a set of 4 independent processes on a VAX with serial communications to a Denning Research Vehicle. At the same time work has begun at LIFIA on the development techniques to control action and perception with a production system, for the applications of a mobile military robot, and a mobile maintenance robot. All of the above projects have contributed to the ideas presented below.

1.2 Problem Statement

The central problem addressed in this paper is

> How can a system architecture for a mobile robot be designed such that the robot can automatically determine the appropriate actions to accomplish a specified goal, and base its execution of actions on a dynamically maintained model of the environment.

In addition we are interested in a number of corrolary issues such as:

1) The application of poorly structured knowledge for coordination of action and perception.

2) The combination of perception and control for such primitive actions as following a road or hallway.

3) The combination of (possibly poor) sensor data from multiple sources and multiple viewpoints to maintain a coherent model of the external environment.

4) The estimation of the uncertainty in localisation due to both dynamic effects (e.g. slippage) and perceptual errors.

Remarks concerning each of these problems appear below.

1.3 Contents

The remainder of this paper is organized into 4 sections. The first of these (section 2) presents an overall systems architecture as well as the structure of the systems for perception and for movement. Section 3 reviews the basic techniques which have evolved for navigation, and then describes a number of lesson which we have come to appreciate. Section 4 presents a similar review of the perception problem and lessons which we have learned. Section 5 concludes the paper with a conclusion and a brief prediction of future progress.

2 System Architecture:

This section describes a system architecture for an autonomous intelligent mobile robot. The architectures of the systems for motion tasks and perception are then presented and discussed.

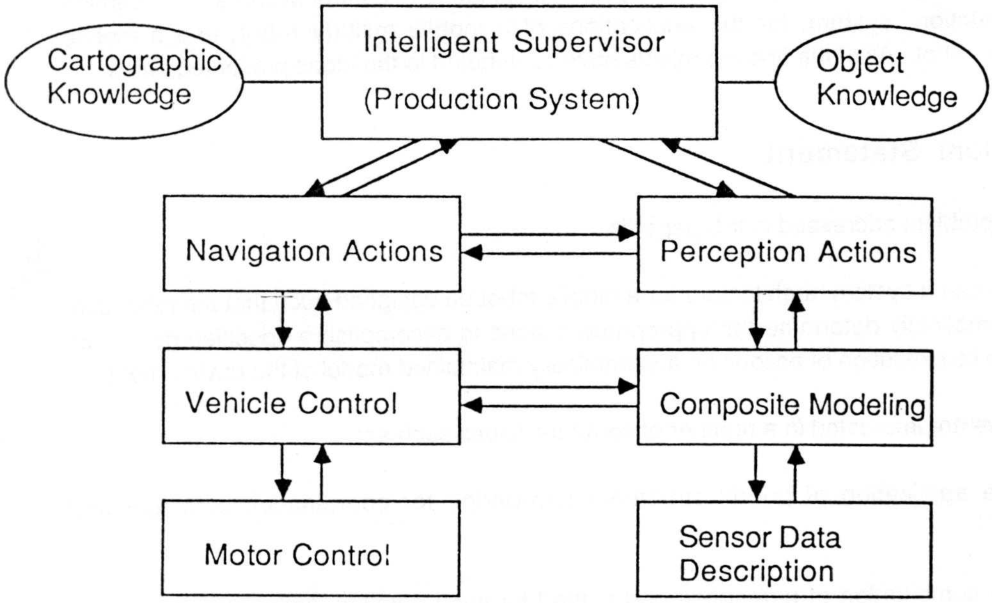

Figure 1. The Hierarchical Control of Action and Perception

2.1 Intelligent Control of Action and Perception

The development of systems for navigation and for perception has gradually led us to the conclusion that both processes are best handled by a hierarchical organization of processes. The power of hierarchy is based on the principle of abstraction, in which succeeding levels are capable of reasoning with greater scope by manipulating symbols which represent groupings of more complex information. This idea is in fact well known and imbedded in many aspects of most human cultures and social organizations. A much more surprising lesson has been the fact that both action and perception have a natural organization into parallel hierarchies in which corresponding levels have a need to communicate. In particular, both the hierarchies for action and perception exhibit a boundary between the information which can or can not be directly perceived or executed. Both processes also exhibit a boundary between information expressed in instrument coordinates (motor control signals and raw sensor data) and information expressed in vehicle coordinates (vehicle motion commands, and the composite model). Parallel hierarchies for vision and perception are shown below in figure 1.

At the top of the system is a supervisor charged with intelligently coordinating action and perception. In the systems developed for the IMP and for the surveillance robot, this top level was an algorithmic process which was hand crafted to accomplish the desired modes of behavior. Intelligent coordination requires the application of large amounts of domain knowledge. Thus we have recently begun development of an intelligent supervisor organised as a production system, using the OPS-5 language [Brownston 84]. Advance planning of actions is accomplished by the supervisor level using information in a cartographic data base organized as a "network of places". The network of places is encoded explicitly as a network and not in the rule base of the production system so that the system may easily modify its knowledge of places.

Actions are transmitted from the supervisor to a number of "black box" algorithmic processes for tasks such as following a wall, following a road, traveling around an object or traveling in a straight line to a given point. In a very similar manner, the supervisor makes requests of a number of perceptual "black boxes" which perform algorithmic functions on a composite model such as verifying that a free path exists, finding a given wall, or finding instances of an object.

The basis for perceptual "black-box" functions is the composite surface model, which describes the immediate external environment in terms of geometric primitives expressed in world coordinates. The composite model does not interpret sensor data as objects; any such interpretation is carried out by the higher level supervisor with the aid of the perceptual black-box processes. Similarly, the action "black-box" processes generate commands for vehicle movements such as move, turn or stop. A vehicle level protocol involving independent asynchronous commands for move and turn is described below.

Interestingly, both the control of vehicle movements, and composite modeling of the environments are based on access to an estimate of the current position and orientation of the vehicle. The controller for vehicle movements integrates differential signals from wheel encoders, and if possible gyroscopic and other sensors, to maintain an instantaneous estimate of the vehicles position and orientation. This instantaneous estimate of position and orientation is used as the reference signal for controlling the specified vehicle movements.

The composite modeling system uses the current estimate of the vehicle's position to project sensor data from sensor coordinates to a world coordinate system. This permits information from different viewing positions to be combined in the composite model. The process of matching new sensor data to the composite model can be used to measure errors in the vehicle's estimated position. These errors can then be fed back to the vehicle controller as a correction vector.

3 Planning and Execution of Motion Tasks

The fundamental task of a mobile robot is to successfully navigate to an assigned location. If the world is known to be empty between the robot's starting location and the final goal, then the robot may proceed by a straight line path, and the only perceptual problem is that of maintaining a good estimate of the current position and orientation. If the world is not known to be empty, but the goal is within the range of current perception and easily attainable, then it is feasible for the robot to attempt a straight line path and dynamically react to perceived obstacles which intervene between the robot and the goal. In the general case, neither condition is true; generally the goal is too distant from the robot to be immediately perceived, and obstacles are known to exist between the robot and the goal. Thus the robot must first plan a path to the goal using an abstract description of the world, and then execute that plan, modifying the plan to account for unexpected differences in the world.

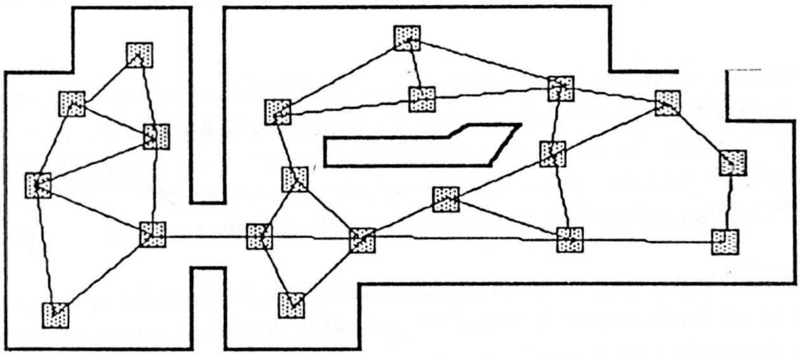

Figure 2. A Network of Places

3.1 Planning a Path

Almost all path planning techniques can be described as a graph search through a network of discrete "decision points" [Crowley 86a]. Such algorithms may differ widely in the manner in which the decision points are chosen. In developing the planning system for the surveillance robot we observed that the actual placement of "decision points" was not critically important, provided that they were placed at a reasonable density, and connected whenever a free path existed between adjacent nearby points. For the case where the robot perceptual system has a maximum range of D, a reasonable average density would be D/2. Of course, points should be neither too far or too close to walls and other limits to free space.

The network of decision points used by the surveillance robot were chosen by the robot itself during a learning phase in which the robot explored the environment by following walls

[Crowley-Coutaz 85]. The learning phase served both to acquire the network of decision points and to acquire a reliable model of the surfaces seen by the sonar sensors. To guarantee a reliable and complete world model, the robot would stop every 60 cm to update his world model. At each such point, the system would verify that a free path of less than 3 meters existed between the last learned "decision point" and the next stopping point. If no such freepath was found, a "place" or decision point was encoded for the current stopping point. This place was linked to all other learned places within 3 meters for which a free path existed. Because the robot followed walls at a distance that was optimum for position estimation, places were created at locations that were well suited to correcting the robots estimated position.

3.2 Algorithms For Vehicle Movement

Vehicle movements such as wall following, road following, or straight line travel are algorithmic processes which require a tight link with perception. Such process can be implemented as independent procedures which access the information in the composite model by a set of interface functions, as described in [Crowley 85c] and [Crowley-Coutaz 85]. In both [Crowley 85c] and [Crowley-Coutaz 85] straight line control is presented under the name "local navigation". Algorithms for wall following and road following are described in [Crowley 86a]. All of these process generate vehicle level commands such as move and turn and stop.

3.3 Vehicle Level Control

Two principles of vehicle control have been found to be particularly important. First, the vehicle controller should be asynchronous. That is, the controller should be capable of responding to a new command at any instant, even if the previously issued command has not finished execution. Secondly, the vehicle controller should make available, at any instant, the estimated position of the vehicle. This estimated position is essential to the perception processes as well as to the movement processes.

A general purpose vehicle-level protocol was developed for the Terragator robot. This protocol consists of the asynchronous interpretation of the following commands:

Move [V, [D, [A]]]	Move for a maximum distance of D at velocity V. All accelerations should be at A.
Turn [V, [D, [A]]]	Turn for a maximum distance of D degrees at velocity V degrees/second. All angular accelerations should be at A.
Stop [A]	Stop with acceleration A.
Emergency-Stop	Stop as fast as physically possible.

Get-Estimated-Position Return the current estimated position and orientation.

Correct-Estimated-Position ($\Delta x, \Delta y, \Delta \theta$) Correct the estimated position by adding the parameter vector.

All parameters in square brackets "[]" are optional. If these values are ommited, the last given values for that parameter are used. The commands for move and turn are independent; that is, a command of turn has no effect on the execution of a move command, and vice versa.

3.4 Uncertainty in Estimated Position

We have recently come to appreciate the importance of estimating the uncertainty in the estimated position of the vehicle. The uncertainty in the estimated position determines the tolerance region which must be kept clear around the robot. Knowledge of the uncertainty can permit the robot to dynamically adjust its freespace tolerances and its sensing strategies. Perhaps even more important, position uncertainty is a major factor complicating correspondence matching during perception (along with uncertainty in the sensor behavior). Having available an estimate of the position uncertainty can simplify the correspondence process.

In the vehicles with which we have had experience, the major sources of error in the position estimated from odometry have been

- Wheel slippage caused by acceleration,
- Wheel slippage in traveling due to uneven surfaces,
- Off-axes rotation during turning, due to uneven weight distribution, and
- Position error arising from orientation error.

We have recently begin experimenting with modeling vehicle position uncertainty with a covariance matrix. Uncertainties are built-up incrementally during travel, and are reduced by constraints provided by perception.

4 Composite Modeling

One of the most important scientific lessons of this sequence of projects has been the importance of combining information from different sources and different viewing points. This problem is sometimes referred to as the "sensor data fusion" problem. Our approach to this problem is a body of techniques which we refer to a composite modeling.

4.1 Computational Framework

The architecture for a composite modeling system is presented in figure 3. As new data arrives from sensors it is immediately projected into position invariant scene coordinates and converted to an abstract representation. The new data is then matched to the existing composite model. The result of matching is used to determine errors in the estimated position and orientation, as well as to provide new constraints and new structures for the composite model.

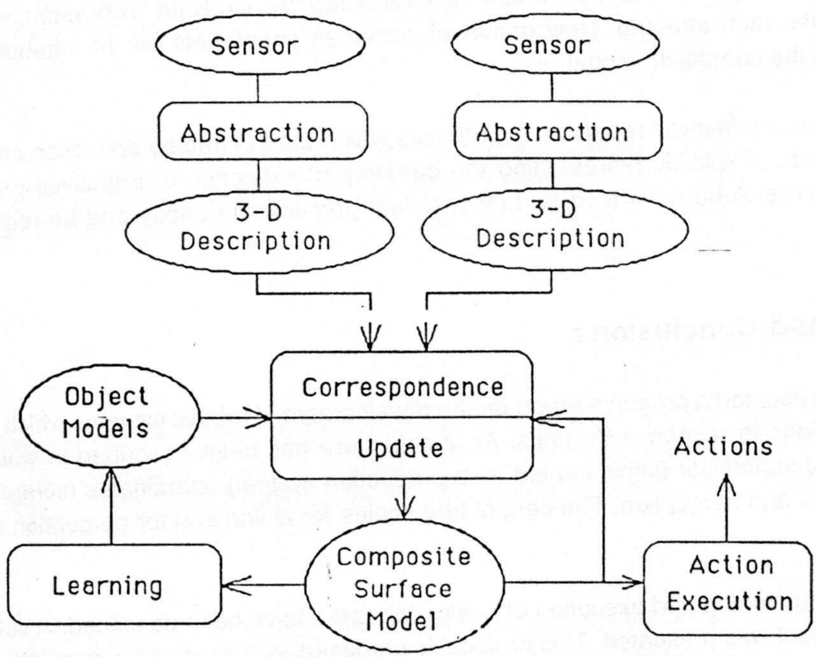

Figure 3. The Computational Framework for Composite Modeling

A prelearned model of the world can serve as an important source of hypotheses for constructing the composite model. In the case of a ring of ultrasonic range sensors on the surveillance robot, in normal operation, the quality of composite model was dramatically improved when a prelearned segments were introduced into the composite model hypothesis with a low confidence.

4.2 Representing the Composite Model

For interpreting ultrasonic range data, we have developed a representation based on 2-D line segments which represent the limits to freespace. In the context of a 3-D vision system we have recently developed a vocabulary primitives including planar patches, contours, cones and ellipsoids [Crowley 86c]. These primitives are based on the following set of principles:

1) Composite model primitives are geometric entities which represent surface or volumes expressed in a position invariant scene coordinate system. Interpretation as known objects may be performed, if necessary, by later processing.

2) Composite model primitive represent hypotheses which account for a continuum of possible geometric interpretations. This continuum is expressed by explicitly representing the uncertainty within each attribute. New primitives serve as constraints on the attributes of primitives within the composite model.

3) Composite model primitives represent hypotheses which are inferred by abduction and are thus never certain. Explicitly representing the certainty of existence of primitives permits consistent primitives to be reinforced and inconsistent primitives to decay and be removed from the model.

5 Summary and Conclusions

This paper has presented a progress report on the development of robotic systems which plan and execute actions in response to goals. An architecture has been described in which a knowledge-based supervisor (implemented as a production system) coordinates hierarchical systems for action and perception. The control hierarchies for action and for perception have been described.

Recent lessons in planning and execution of navigation tasks have been described. A vehicle level control protocol was presented. This protocol is proposed as a general purpose interface which separates the details of controlling a particular vehicle from the problems of navigation and perception. Such a protocol may be thought of as a "virtual vehicle". Adoption of this protocol would permit navigation software to be easily transported among widely different vehicles.

The problem of estimated the uncertainty in the robots position was described. This uncertainty is useful to both a navigation task controller and perceptual processes. A computational framework for composite modeling was presented and primitives and principles were discussed.

This paper is not intended to serve as a self contained description of technique for navigation and world modeling, but is written to present lessons and principles in navigation and perception that have been recently developed. Readers interested in technical details are referred to [Crowley 85c], [Crowley-Coutaz 85], and [Crowley 86a].

Bibliography:

[Brownston 85] Brownston L. , R. Farrell, E. Kant, N. Martin, Programming Expert Systems in OPS5, Addison Wesley, Reading Mass., 1985.

[Crowley 84c] Crowley, J. L., "A Computational Paradigm for Three Dimensional Scene Analysis", IEEE Workshop on Computer Vision, Annapolis Md, March 1984. (Also available as CMU Tech Rep. CMU-RI-TR-84-11.)

[Crowley 84f] Crowley, J. L., "Navigation and World Modeling for a Mobile Robot using a Sonar Collar", 1st International Conference on Robotics in Construction, C-MU, June, 1984.

[Crowley 85b] Crowley, J. L., "Navigation and World Modeling for a Mobile Robot Using a Rotating Ultrasonic Ranging Device", 2nd IEEE Conference on Robotics and Automation, St. Louis, March, 1985.

[Crowley 85c] Crowley, J. L. "Navigation for an Intelligent Mobile Robot", IEEE Journal on Robotics and Automation, 1 (1), March 1985.

[Crowley-Coutaz 85] Crowley, J. L. et J. C. Coutaz, "Navigation et Modèlisation pour un Robot Mobile", IMAG Rapport de Recherche 545 (LIFIA 28), Universités de Grenoble, Juin 1985. (in French) (to appear in TSI).

[Crowley 86a] Crowley, J. L., "Path Planning and Obstacle Avoidance: A Survey", IMAG Rapport de Recherche 565 (LIFIA 35), Universités de Grenoble, Janvier 1986. (in English) (to appear in The Encyclopedia of Artificial Intelligence).

[Crowley 86c] Crowley, J. L., "Construction and Maintenance of a Composite Surface Model", Third IEEE Conference on Robotics and Automation, San Francisco, March 1986.

[Whittaker et. al. 85] W. L. Whittaker, J. L. Crowley, I. J. Oppenheim, J. Bares, T. A. Wood, S. Berman, and K. Lee, "Mine Mapping by a Robot with Acoustic Sensors", 2nd Int. conference on Robotics in Construction, C-MU, June 1985.

James L. Crowley*

LIFIA (IMAG)
BP 68
38402 St. Martin d'Heres
France

*(On leave from the C-MU Robotics Institute)

Phone: (33) 76.51.48.74

Computer Mail:
uucpnet: ...!mcvax!inria!imag!jlc
arpanet, csnet: !mcvax!inria!imag!jlc@seismo.arpa