Principles of Planning and Control Concepts for Autonomous Mobile Robots

Paul Levi

Forschungszentrum Informatik (FZI)
Haid- und Neu-Str. 10-14
D 7500 Karlsruhe, F.R.G.

ABSTRACT

This paper tries to solve three questions. The first question is which of the knowledge based planning techniques are to be used or must be developed further in order to provide autonomous robots with appropriate intelligence. The second question illustrates which planning modules has already been implemented under the light of new planning techniques. In the last theme the various concepts for control systems of autonomous vehicles are ***constructed.***

1. INTRODUCTION

Autonomous mobile robots should be able to move around in an unknown real world and it should be able to perform a dedicated task from a given class of tasks. It should be in a position to cope with uncertainties and to learn task adaptability.

Stationary and mobile autonomy is obtained by partitioning of the robot architecture into three basic parts consisting of a multiple sensor system, a world model, and a control module. Within this architecture the additional requirements for mobilities also exist. Examples for this are the motion control (very manoeuvrable), maintenance of balance (e.g. walking machines), self powering, on-board intelligence etc. Planning activities can be related to short term goals with obstacle avoidance and road following. As long term goals, planning activities are related to off-road-travel and object recognition in a rapidly changing environment. What has not been done uptil now, is the fulfillment of the conditions for coding capacity and the fast function of the knowledge-manipulation, the fusion of a variety of sensor data, the long range navigation and the mission planning.

The planning of robot actions was one of the first task area in the scope of AI-methods. This first use reduces the complexity of the world to the blockworld and allows the uncertainties to be overlooked. As control strategy, the linear planning is used. It is clear that this classical use can not lead towards true requirement for mobile autonomy. Due to this conclusion the question arises, as to which new use of planning exists for robots and to do that which AI-techniques should be used. In chapter 2 the AI-problems relevant to robots e.g., planning with uncertainty, achievement of multiple goals, constraint manipulation etc. are described. These uses are, however, not specially to be developed for robotics but, in principle, they exist and they concern todays applications in robotics.

Each individual knowledge based planning technique require a more conceptional basis. The robotics is surely a predestinated field in order to give the above mentioned impact. In chapter 3 therefore, it is investigated which new planning module has been investigated for mobile robots e.g., for the road-following and for the visual navigation.

All major industrial nations have initiated programs in order to meet the challenge to develop autonomous mobile systems. In chapter 4 the different concepts to structure the control modules of autonomous vehicles and their realization will be described.

2. Knowledge Based Planning Techniques

2.1 AI-Problem Domains of Planning

Since 20 years planning systems are known to be a well established research area within the scope of AI. Their activities e.g.: search strategies, reasoning, knowledge representations etc. are closely related to the principal concerns of AI. /Tate 85/. But, planning also introduces its own problems as well as it generates special techniques by itself. Typical current domains of interests are:

(1) Use of multidimensional abstraction space (problem decomposition) for situations and/or operation abstractions.
(2) Achievement of multiple goals.
(3) Activation of multiple agents with cooperation or interference.
(4) Perception of the real world.
(5) Uncertainties in the world model (dynamic environment).
(6) Uncertainties in the execution and monitoring of plans.
(7) Constraints manipulation for the generation of suitable solutions.
(8) Replanning actions as an answer to unexpected events.
(9) Efficient interleaving of the planning and execution phases of an autonomous system.

Planning has also some common feature with the development of expert systems. These are the need for the justification of solutions (explanation component), the Truth Maintenance Systems (TMS) /Doyle 79/, the qualitative reasoning /Forbus 86/ including the connection between temporal and causal reasoning and the knowledge elicitation.

The planning process is therefore considered to be not independent on other important activities of the AI. The necessity to deal with the planning problems as mentioned above, is also made by the use of expert systems as they are involved in many different application areas.

The single step plans have already been extended to multi-steps in the seventees. Basically there are two starting points to be achieved in multi-level concepts . One of them is the situation abstractions as in ABSTRIPS /Sacerdoti 75/. Other is an abstraction of the possible operations. The classical system which works with this idea is presented in NOAH /Sacerdoti 77/. Both these applications may be combined to a more powerful system, in which both of them may be applied simultaneaously. A planning system using this combination is described in MOLGEN /Stefik 81/.

The use of various abstraction levels are often made when the goal to be achieved are not one, but many. The necessity to achieve

CH2413-3/87/0000/0874$01.00 © 1987 IEEE

several goals simultaneously is usually coupled with the complexity of the problem and the efficiency of the corresponding solution algorithm. Typically, a scheduling problem originated from the ISIS system /Fox 83/, is exclusively taken as purely order oriented. The extension of this system to OPIS is carried out for the fact that simultaneously three goals eg., orders, resources and events are to be achieved /Ow 86/.

A typical application area in case of mobile robots is the mission planning, where several goals must be achieved simultaneously. An example for such application in industrial area is to collect materials from various store houses with the help of an autonomous platform and supply them to the individual workstations (machine tending) /Newman 85/.

The development of 3rd generation robots leads inevitably towards the dealing with uncertainties and the adjustment towards similar task classes (eg., flexible production) /Giralt 85/. The bridge that leads towards this objective is predominently fulfilled by the basic block AI /Brady 85/.

Planning arising from available uncertainties (world model, effect of an action) should not be single stepped. A multi-level planning allows the uncertainties to be divided into hierarchical classes, in order to bring them in a systematic way. Due to this, for autonomous route planning a global plan is first established, thereafter a navigation module generates a local plan which deals specially with local obstructions and finally the pilot distributes the locomotion instructions /Meystel 86/.

The first work of assembly task is to prepare the planning but not to carry out the supervision of uncertainties /Brooks 82/. It starts from a linear planning and gives reasons about the propagation of symbolic constraints and the accumulation of errors. Sensors are inserted in plan frame works, in case it is required to reduce the uncertainties. However, the uncertainties created by the sensors itself must be taken care of.

This use is taken up and extended by the GRIPE system /Doyle 86/. This work primarily consists of constructing and monitoring task execution in the plan. The plan here is a totally ordered sequence of actions for one agent. Each action in the plan has a pre- and postcondition consisting of assertions. For these assertions, appropriate requests and expectations for perceptions by different sensors are generated. Verification operators map these requests on assertions and on possible subgoals.

A further use to construct the sensor instructions systematically during the planning phase is given by /Gini 86/. The principle of this use is based on the fact that the additional knowledge from the task and the environment is used to connect the sensor information and significant external events. The system is inserted at various points in the plan to use appropriate sensor requests and to allow automatic error recovery.

The use of robot specification for reasoning with uncertainty must also be considered in the light of newly developed expert system and AI-techniques. In addition for some uses such as the special parameterized uncertainty calculations that are used for pruning deduction paths of low confidence /Freedman 85/, or in bayesian networks /Pearl 85/, there exist the most important developments that are defined by nonmonotonic reasoning /Mc Dermott 80, 82/. Nonmonotonic reasoning is used in situations where:

(a) the information is unreliable and inconsistent (eg., camera images)
(b) the information is incomplete
(c) the problem domain is unstable
(d) use of relatively simpler assumptions for modelling complex connections.

On the basis of such assumptions, it is clear that readily prepared propositions must be reconsidered on the basis of new informations available. Essentially point (c) leads to the frame problem /Mc Carthy 69/. The solution of the frame problem is essentially the start towards nonmonotonic reasoning. Typical formalisation which has been developed with respect to this method is the belief revision (TMS), the default reasoning /Reiter 80/, the qualitative reasoning, the temporal reasoning /Allen 85/, /Vilain 86/, the persistency and the constraint manipulations /Sussmann 80/, /Dechter 85/.

For qualitative reasoning, some special world models have been constructed. Examples are physics manifesto of /Hayes 85/, the robot-oriented description of everyday physics /Schmolze 86/, and the active knowledge representation about the manufacturing process (factory scheduling, vehicle control, time), /Sathi 85/.

The integration of various techniques e.g., default reasoning and the introduction of frame axioms in the nonmonotonic reasoning is brought into light in the work of /Hanks 86/. In this work, it is shown that a simple extension of classical (first order) logic may express an important aspect of nondeductive reasoning. But, the term "consistent" (modal operator M) is more complex than what a simple intuition would suggest.

Realistic use of planning are based on the fact that new plannings are required for non-expected events or uncertain informations. Besides the perception, the use of planning also belongs to this, which works not only top-down but also in a bottom-up manner. The role of AI in computer vision is not very clear till today /Shapiro 85/.

Use of mobile robots for solving a production task give rise to many more agents which must be synchronised. This activity is called distributed planning. Sub-problems are passed between specialized planning experts and must be controlled by a executive. A multi agent plan synchronizer can perform this coordination /Stuart 85/. The execution of a plan corresponds to a sequence of messages between the agents and the environment. The plan steps will be synchronized by inserting send operations and supervising these operations by a synchronization skeleton. The resulting plan defines all possible action sequences of the multiple agents which always allow for plan termination. Also, work is in progress to construct state spaces which contain loops in order to describe iterative behaviour /Drummond 85/. This use may be taken as a basis for reactive planning.

2.2 Search space control

The old method of planning is based on the principle that the plans may be performed in one step.This application gives rise to a huge search-space which may not be pruned by the use of efficient heuristics. Therefore planning systems are developed around the year 1975 which gives rise to many abstraction/hierarchy levels in the search space. A point in the search space represents essentially a partially completed plan. From some skeleton plan, a set of decisions are made which generates more elaborate plan "islands" to meet the refined constraints of the goal criteria.

In complex cases the backtracking method is not suitable. For example, it may be mentioned that in the scope of belief revision, the use of backtracking may lead to contradictory inferences. In order to avoid this contradiction therefore, several assumptions which have been made are taken back. This withdrawing of assumptions are called dependency-directed backtracking. In addition this backtracking method is also used in case of constraint propagation through networks.

A new extended searching method is the opportunistic search /Hayes-Roth 85/. In this method the problem is solved by identifying the "best" aspects of the problem (eg., more powerful heuristics, more information is available) and the development of one or more candidate solutions. The effectiveness of opportunistic problem solving lies in the ability to identify the solution path that solves the planning process quickly. The opportunistic search is brought into light only when it is not possible to build up a complete search space and it is not possible to achieve a strong hierarchy of various goals according to their prefered importance.The technique of meta planning is very frequently used again by opportunistic planners.

A major problem which a planning process (problem solving) encounters, is the subgoal interaction and the goal ordering coupled with that. The linear planning is based on the principle that a linear ordering of subplans are prepared mostly to achieve the objective. This use is brought into light, in case the subplans continues to be independent. It may also be required to destroy some of the already prepared subplans and make a permutation of the total operations in order to achieve a solution. The non-linear planners as described in NONLIN /Tate 77/ do not however make a complete ordering of subplans, rather insert the subplans in a partial ordering network. The network consists of actions and goals, and the ordering is made when the solution demands it. The advantage of nonlinear planning is clearly described in TWEAK /Chapman 86/. This system is based on an incomplete plan and it is refined further depending on the constraints which are added to it. These constraints take time constraints and several goals into account.

The inference techniques for handling uncertainities depend on the type of uncertainity and can be classified in those which infer **with** uncertainity and those making inference **about** uncertainity /Mamdani 85/. These two techniques are not mutually exclusive and may both be present in a real application.

The belief revision problem of the nonmonotonic logic is a typical example of the inference about uncertainity. Nonmonotonic logic is particularly useful. The reasoning process uses as many assumptions as possible without generating inconsistency.

The techniques of inference with uncertainity are the statistical use and the use of fuzzy sets. One can use Bayes theorem on inference chain in rule-based expert systems, in which case not only the a priori probabilities but also the joint probabilities are calculated before the probability of a hypotheses can be generated. This calls for an enormous overhead, so the inference rules have to be modified by heuristics.

Fuzzy sets can be used to represent imprecision /Goodman 85/. The prior information in fuzzy sets correspond to the possibility of distribution of evidence. Where joint possibilities do not exist. Fuzzy possibilities (eg., most, a few etc.) make it possible to have a conceptual framework. The use of dispositions (propositions which are preponderantly true) allows default inferences that are correct unless evidence to the contrary is available. This is an another way to do default reasoning instead of nonmonotonic logic.

A new concept which uses ATMS /de Kleer 86/ and may also be applied in robotics is described by /Morris 86/. It organizes a search through a space of alternatives and assists detection of inconsistent contexts and the maintenance of the derived results. But, it has till now no mechanism to integrate actions. In this work, the procedural nets of NOAH are extended to the ATMS world. Each individual world describes the affect of an action by a world node and the explicit set of assumptions to justify the world node environment. Beside the ATMS-application for robot-actions, it may be said that all planning components which has been explained in this paragraph are used only on the basis of well established AI-techniques.

It is to be established how the new applications, as for example, the uncertainty propagation and the qualitative reasoning may lead towards the future, so that the planning applications becomes more autonomize and at the same time more applicable in real practical applications.

3. Towards Autonomous Vehicles

3.1. Route Planning

The goal of a route planning is to analyze all available information to produce a route that is optimal in the light of predetermined mission constraints and requirements. The requirements which are to be fulfilled for the route planning of an autonomous vehicle may be defined in three levels:

Mission planning, global route planning, and local route planning.

The mission planning determines the requirements and the constraints for the global task specification /Pearson 85/. Inputs to this planning module are given by mission rules /Gilmore 85/. The planner must then analyse these rules and determine a problem solving strategy which includes the additional required constraints like visibility and trafficability.

The global planning is generated on the basis of geographical goals (places) defined through the mission planning. These geographical goals are dependent on the space availability eg. forest, roads, natural and industrial terrain, the mobility constraints, the vehicle behaviour etc. This path is an approximation of the actual path, where details of the local elements (e.g. obstacles) are not taken into account.

The local planning determines the real path finding on the global route which takes care of the unexpected obstructions. At that point a local world model is constructed which is realized from the assimilated perceptual data. In the work of /Payton 86/ the local planning does not give the locomotion commands to the pilot (vehicle actuators), but there exist an additional so called reflexive planning level.

The planning activities of mobile robots distinguish itself from those of stationary robots essentially in following two points:

(a) Replanning may easily be performed at all planning levels.The reponse time ranges from seconds (come across an obstacle) to minutes (to find a new global route) upto several hours (division on the basis of mission order).

(b) Through the mobility, the sensor processing (multiple sensor data fusion) as also the knowledge manipulation as for example in the form of heuristic rules, is placed to a special high requirement level.

Plans for autonomous vehicles have typically a hierarchical structure with goals and subgoals at multiple levels. In each individual level, as for example, on the level of the local route planning the requirements of the stationary and mobile robot are similiar if the mobile robot is considered without replanning. This gives rise to the fact that the free space technique for the obstacle avoidance (configuration space, free way, mixed representation) may also be applied for mobile robots /Kuan 84/. On the contrary, the task of mission planning is primarily related in case of mobile robots.

An application which distinguishes itself clearly from the stationary method is described by /Linden 86/ for the Autonomous Land Vehicle (ALV). It uses dynamic programming in order to bring the global and local route planning together. The required cost matrix is generated for symbolic as also for object-oriented representations of terrains. Replanning happens at several levels with the cost of replanning proportional to the scope of the changes. A plan monitor mediates for route generation and replanning.

The application of the dynamic programming is also elucidated by /Parodi 84/. This approach connects it with relaxation in order to define an efficient graph search algorithms on freeway graphs. More special applications to solve the navigation problem have been made by /Moravec 85/.

A special use for an inexpensive navigation system, which is suitable for indoor environments is described by /Crowley 84/. The robot domain is represented as a network of maximum area convex regions (network of places). Connectors (e.g. doors) serve as key points for planning through known environments as sequences of seperate straight line motions. The local route planning is monitored by a finite state process.

More sophisticated cost functions, be it for graph search methods or for dynamic programming can be used . Typical cost functions are: traversibility, fuel limits, travel time, weather conditions etc.

3.2 Navigation and Perception

A planning system for indoor or outdoor autonomous vehicles is basically supported by two major sources of knowledge about its environment such as: digital maps of the environment and the sensor system. These two knowledge sources must be exploited in a connected manner in order to react immediately and also in a flexible way. The adequate modelling for the solution of this navigation problem must start from the a priori data and should be specially capable to actualize the local route planning of the internal world model through the assimilation of various sensor data (recognition adequacy). In addition, for the world modelling, the following criterions are to be noted:

(1) **descriptive adequacy**. All objects and situations in the environment which are mandatory for the vehicle to move must be modelled.

(2) **handling uncertainty**. Locations are determined relative to other known locations (e.g. landmarks) and not with respect to a globally consistent environment map.

(3) **learning**. An autonomous system is required for navigation in complete new or partially known terrain. It should be able to generate adaptive own local terrain map, in an adaptive manner.

The difficulties in the model building for route planning lie in the fact that in addition to an adequate world descripton an extended object recognition is necessary. A vision system which interprets arbitrary natural scenes is not available; merly restricted environment can be interpreted (e.g. ACRONYM, VISIONS).

The perceptual knowledge sources for the global route planning is primarily based on topological descriptions and the nature of terrain. Geometrical restrictions are primarily needed for the local route planning.

In addition to this perceptual knowledge models that maps various abstraction levels are also available. Movement in an outdoor terrain gives rise to a priori available digital terrain map which shows the elevation contours, road networks, buildings, loans etc., which is taken as an input for the global route planning. For indoor scene a three dimensional CAD model may serve as an input. Thereafter one may use high level perception tools, so that side by side with the digital map, an image interpretation may also be carried out, which executes a segmentation (e.g. road segments, road intersections, and cross country regions). After that a two dimensional grid network may be used to have a global route planning. /Giralt 84/ has used polygonal cells for the representation of the indoor environment. The topological representation of these cells is done by a connectivity graph. /Soetadji 86/ uses a CAD System to model a cube based three dimensional model (oct-trees) to describe the environment and the obstacles.

After a global route which includes right, left borders and direction and/or landmarks is generated, different local short (e.g. linear) paths are to be generated, for example for cell clusters of road segments. The navigation model on the local planning level relates itself to the geometric descriptons, which for example assign dimensions to the elements of a connectivity graph (width, boundaries). In addition, the specification of the allowable speed ofthe vehicle, the allowable possibilities of avoidance etc. are necessary. All these special specification which are necessary for the local planning module and which can be established in a local map is first given on a symbolic (predictive) basis and is to be actualized and verified with the help of perceptual knowledge sources.

The corresponding perceptual modelling for the local route planning concentrates itself on specific local features. This includes real objects, which may be encountered as obstacles, the real nature of the ground etc. Curves, regions, texture elements, surfaces, volumes etc. are basic elements of this perceptual structuring. Grouping representations for parallel lines (roads, contours, anomalies in the horizon line etc.) must also be ingredients of this level of perceptual modelling. More general, spatial and temporal features must be defined in order to perform a dynamic adaptation to the current viewing conditions.

3.3 Route Learning and Environment Model Acquisition

The route planning is for every level an a priori activity which must include the ability for replanning. When a real time route segment (plan fragment) can not be traversed with the static goals, the failure must be reported. Then a local replanning should be initiated. If it fails, the global replanner must be invoked. These simple facts show clearly that in addition to quick reactions, which are adapted to the real situations, the aspect of learning also plays a special role. The known and already travelled routes should be reutilized in replanning and it would have been an advantage when the robot can generate the necessary local and global maps by itself. The use of these route learning and the environment model acquisition are readily in part available /Laumond 83/, /Brooks 85/, /Turchan 85/. But, these problems are far from being solved.

An application where integration of the route planning is made along with environment modelling has been proposed by /Iyengar 85/. The map generation is supported by navigational heuristics, like that one which have been proposed by /Chattergy 85/. The starting point is a two dimensional grid map. In order to support this planning process in addition to the spatial graph a Voronoi diagram is constructed. It divides the total space in polygons, in such a way that any point x in such a polygon which is attached to a vertex p of the spatial graph is closer to p than to any other point in the spatial graph.

A further application to build up with the help of exploring travels various types of maps and to find the path by itself is described by /Freyberger 86/. In this system two dimensional grid maps, path nets and strategies are obtained. With the help of an ultrasonic distance measuring system all the obstructions are defined as circles and are classified as protected regions. The path net is actualized during the exploring travel with the help of five rules (free region, isolated obstacle, passage etc.). Although optimal search strategies may not be learned but the robot may move autonomously in a laboratory room consisting of complex obstructions (10 obstacles).

As we have seen, topological and metric informations are required for route planning. /Mc Dermott 84/ has proposed an assertional data base for topological information and a "fuzzy map" for the metric information. A "fuzzy map" fixes the relative positions, orientations and other geometrical facts of objects only in a default range. For the construction of "fuzzy maps" only partial solutions have been developed. The route generation is modelled as a problem solving task decomposition process. This global and local subtasks are defined in such a high level that it is not possible to apply them in a real robot implementation.

3.4 Techniques for Sensor Data Fusion

Autonomous Vehicles require for their mission and route planning world data which must be delivered through sensors. A local route planning is not possible without real sensor data. But also a global route planning should not be only supported from a priori known facts but it should also be supported by actual sensor information before the actual planning activities are carried out, so that a realistic "planning frame" may be established.

For a mission in a complex environment, one individual sensor is not enough. Either various sensors of one type (e.g. 24 ultrasonic range finders) should be put together or the informations should be extracted from a laser range finder (e.g. for navigation and docking) and pictures from one or more cameras should be integrated /Davis 86/.Special attention made for the road- following problem, has been carried out by the CMU-Terregator, which travels at a speed of ca. 1km/h, /Wallace 86/.

Both the NAVLAB system /Shafer 86/ and the Ground Surveilllance Robot (GSR) /Harmon 86/ use an extended version of the blackboard approach to build up the world model. In the GSR System the originally passive blackboard is added by communication protocols for plans and reports and by active functions in order to provide event driven access to the blackboard information for replanning activities.

The blackboard of the NAVLAB Systems is called a "whiteboard" because it support parallelism in the knowledge source modules and implements special data retrieval requests.

Anyway, a blackboard or a "whiteboard" is a database/ - communication-system which supports but does not solve the sensor fusion problem. The techniques for sensor fusing depend upon the situations. The simplest situation occurs when there is no intersection between measurements because independent situation are observed. If the sensor observation are intersecting (e.g. same object, same property, same spatial location) then the following merging technique are appropriate:

(1) **Competitive**

The hypotheses of each sensor may be in conflict to other different measurements of the same property of an object. A discrete choice between these several hypotheses must be done. The confidence level which can be associated to each sensor data value may provide a tool to make this decision. This technique is typically for sensors observing on overlapping fields of views. A characteristic example for this kind of sensor fusion is the generation of a sensor-based map. Also an averaging techniques can be invoked to fuse the data of an array of acoustic ranging sensors.

(2) **Averaging**

Confidence are used for the percentage calculation for weights in an averaging of sensor data values. This technique might be refined by a more sophisticated calculation of the confidence levels (heuristic and probability theory).

(3) **Complementary**

The value of one sensor is applied to supervize the processing of other more precise sensor estimations. The relative difference of the sensors are used to increase the advantage and to decrease the disadvantage of both. The combination of camera and laser images to build the local terrain map can be done by this approach. In addition, also the converging of vision and touch sensor data can be performed by this technique /Bajcsy 85/.

Another issue in sensor data fusion is the strategy to decide what sensor should be invoked. This **perceptual planning** is specially essential for the competetive and complementary sensor fusion techniques. Since the combination of disparate sensor data can only be performed on a logical level, the sensor specification should be separated from its implementation. This can be achieved by the concept of logical sensors /Henderson 85/.

To speak more generally, the perceptual planning should be supported by a sensor plan. This plan has contributions from the following ingredients:

(1) Definition of a sensor hierarchy which should perform the task (navigation, manipulation). It should be known before the task execution which sensor type should be used, where to be used and when to be used depending on the action and the quantity to be measured. The vision, proximity, and tactile sensors must be specified.

(2) Generation of implicit sensor instructions. Such instractions define merely the desired result of the measurement and does not specify the explicit sensor instructions.

(3) Generation of explicit instructions which determine what exactly is to be measured.

(4) Every object must have sensor specific descriptons how to recognize one object by different sensor (sensor specific object instantiations).

The consideration of such sensor plans into the perceptual planning makes it possible with the help of the principal usabilities of sensors and their dedicated measuring regulations to perform adaptive strategies. Alternative plans can be selected, in order to perform the task depending on the current status of the sensor information.

4.1 NBS-Model

A basic concept of a hierarchical control system for an autonomous robot has been developed by /Albus 84/ at NBS. The control system (computational hierarchy)is constructed from three blocks which are found in each level

(1) compents for task decomposition
(2) Components for functioning of sensor processing
(3) a hierarchical world model

Each level of the task decomposition subdivides subtasks of higher level to a sequence of low level subtasks for the next lower level. The form of these commands are influenced by the sensor feedback at the same level. The world model hierarchy connects the individual components of task decomposition and sensor processing in each level. It generates hypothesis for the expected sensor entries depending on the established tasks.

The essential task of the sensor hierarchy consists of distinguishing the individual predictions, generated from the components of the world model and to compare these predictions with the reality. The world model is steadily actualized by sensor feedback and new predictions are put forward for the next command sequence. If there are serious discrepancies between the predictions and the assessment, a modified action sequence is generated .

The programming of robot control system if structured in this way, may be made of different programming languages on each level. Such system have been programmed with state graphs or with state tables .

4.2 The Karlsruhe Mobile Robot System

The Karlsruhe mobile robot is an extension of the Albus concept /Rembold 85/. The system integrates the navigation, manipulation and sensor processing units in an internal world model. Program flow control, scheduling, data management, monitor functions, I/O, event handling and various other cooperating functions an adequate, robot computer architecture /Rembold 86/.

A "High Level Interpreter (HLI)" is robot independant and can control multiple arms, a vehicle and sensors on the task level. It interfaces with the "Low Level Interpreter (LLI)" which present the movement control level of a single arm, a navigator or a control module for a single device like a vision system or peripherals. Below the LLI is a primitive control level which performs the axis control. Each level is composed of specific modules which cooperate among each other to solve the level specific task.

The basic system modules are of the types: sensor, action and knowledge based monitor. Each subsystem is an indepentant process, which communicates with other subsystems. The sensor processing subsystem of the HLI uses preprocessed data from the LLI level or directly from sensors. Thereby it provides symbolic information for the subsystems of the HLI. All subsystems have access to a blackboard internal dynamic world model. In the dynamic world model all sensor, actuator and joint states, and the quantities derived from these states are presented. In addition the actual and expected states are represented.

Robot task sequences are either of open loop or closed loop. An open loop action plan is described by a list of defined goal states to be reached.The transitions between the states are specified by parameters, intermediate states and times. A closed loop action plan is described by a list of defined goal states and the source of the feedback information and information about the type of closed loop control action.The execution of a action sequence plan can be inter - rupted by the monitor which modifies the sequence by changing parameter or reprogramming the sequence plan for a new control actions.

Report messages are used to exchange information about the status task sequence plan, actual situations and dynamic world model. Status reports contain information about the state of an execution of a task sequence, about conflicts and dynamic world model information. Every task sequence plan will generate one or more status reports depending on the success or failure of the task sequence. The status report generators reference the task sequence plan. Reports about dynamic world model information are initiated by a report plan, which is defined by the planner and transmitted via a report interpreter to the HLI. The report plan specifies the kind of information to be reported to the planner. Dynamic world model information is represented in an unified form of object/attribute/value description.

4.3 HILARE

The integrated navigation and motion control system of HILARE is described by /Giralt 84/. Typical for this approach is the fixed hierarchical structure. On the top of this hierarchy is a general planner and an execution monitor. The resources routing, navigation, local obstacle avoidance, the sensor system etc. are used to produce a linear plan and to control the execution of this plan. The sequence of actions generated by the planner are send step by step to the controller. The controller is organized as a rule system.

Each of the specialized decision modules (SDM's) like planner, navigation and execution controller communicate with each other through a common database (blackboard). This blackboard is parti - tioned into two components: an announcement part and an information part. The SDM's put in the announcement database all the subproblems which they cannot solve by themselves. These inputs are considered as assistance requests for other SDM's. All actualized knowledge or new results of plan executions are stored into the information database.

4.4 Ground Surveillance Robot

A control structure, which is a deviation from Albus concept, is proposed by /Harmon 84/. In this approach, three essential components the sensor module, the control module and the inference machine are distinguished.

The inference machine delivers the necessary aids for planning and final reasoning. It supervises the sensor informations and delivers action planning to the control module or planning report to the sensor module. It decomposes these plans to control tasks which are supervised by the control module, and further subdivides it into two parts. The sensor plans are given only to those sensors which have to supervise a definite gripper. The control commands are given to the drives directly. The sensor plan is supervised by the inference machine, in order to help the control module in task analysis and also to recognize the errors in the control module. The communication protocols which are defined for the synchronisation of these modules are accomplished through a transport system in the form of messages which must be worked out through processes. Content of these messages are plans and reports. A plan consists of a name, a start condition, a trigger condition, an end condition and lastly a plan action. The plan itself is defined in the form of production rules.

For each robot subsystem, the world model is represented by a blackboard. All sensor and state of actuators and the expressions associated therein (symbolic descriptions) are fixed at this stage. In addition to the passive knowledge objects there are procedures which manipulate these objects in the real time within the scope of a multiprocess environment. The structure of the blackboard is so subdivided, that it may be constructed as data-oriented and also as model-oriented control flow.

This concept is implemented not only in case of autonomous mobile robot but also in case of automation. It has been shown that this concept works very well in both these applications involving various requirements of different kind. Therefore it has been shown that blackboard mechanism is very suitable as a coordinator in situ - ations where different programmers work in various subcomponents of a control system.

4.5 DAISIE

A further concept for the control architecture is represented by /Orlando 84/. It is the Distributed Artificially Intelligent System for Interacting with the Environment (DAISIE), a testbed system integrating AI Software into robotics by the NASA.

Typical for this system is the emphasis on the communication structure, the hierarchical control structure and the plan/execution interleaving. On the strategic level, commands can be issued without necessarily waiting for responses. At the lowest level (tactical level) the commands that are issued to the actuators/sensors are very sensitive to changes in the environment. In the intermediate levels "chunks" of plan commands must be issued. The communication system of DAISIE realizes such a hybrid system. The hierarchical control structure is emulated by an overall hierarchical concept that uses the communication system, in order to send plans/signals immediately to the next level, with no intermediate processing. The plan/execution interleaving is adapted to the different abstraction levels in connection with the hybrid communication concept.

4.6 CMU-Architecture

Since 1981, a very ambitious project for mobile robots has been launched. At the CMU it started with the CMU-Rover and has been prolongated to the Pluto, the Neptune and the Uranus vehicles /Moravec 85/.

In the architecture for the distributed control for these three autonomous vehicles expert modules run as concurrent processes which are synchronized by the aid of a blackboard. These modules are distributed over a processor network. Expert modules exist for the sensory data interpretation, for the building of an internal model of the robots environment, for the generation of plan strategies, for the monitoring of the plan execution etc. Each of these expert modules is composed in a master and a slave process. The master retrieves the data from the blackboard which are needed by the slaves and schedules the slave module. The slave module itself is responsible for the processing and problem-solving activities.

A supervisor module abstracts information from a control plan. This plan describes the subtasks and the constraints in their execution. High level information is distributed to the master modules by the blackboard. The consistency of the blackboard date is insured by the monitor. Besides the blackboard, processes can also exchange data directly among themselves.

4.7 MOBOT-2

MOBOT-2 is an autonomous vehicle which is developed at the MIT /Brooks 86/. This approach differs principally from the hierarchical approaches which has been described uptil now. The decomposition of the robot control structure is purely based on task achieving behaviours. The idea is to have a number of levels of competence. These lower levels are:

(0) Avoid objects
(1) Wander aimlessly around without hitting things
(2) Explore the world
(3) Build a map of the environment and plan routes from one place to another.

The entire system is built up by finite state modules, which show a strong similarity to neuron models. Each module has inputs and outputs which can be suppressed and inhibited respectively. Level 2 represents the behaviour of an "explorer", which has no ideas and plans what kind of goal it should achieve.

References

/Albus 84/ Albus, J.S.: Robotics, in Robotics and Artificial Intelligence (eds. Brady M. et al.), 65-93, Springer Verlag, 1984

/Allen 85/ Allen, J.F.; Hayes, P.J.: A Common Sense Theory of Time, IJCAI-85, 528-531, 1985

/Bajcsy 85/ Bajcsy, R.; Allen, P.: Converging Disparate Sensor Data, in: Robotics Research, The Second International Symposium, 81-86, MIT Press, 1985

/Brady 85/ Brady, M.: Artifical Intelligence and Robotic, Artifical Intelligence 26, 79-121, 1985

/Brooks 82/ Brooks, R.A.: Symbolic Error Analysis and Robot Plan - ning, Journal of Robotics Research, Vol. 11, No. 4, 29-68, 1982

/Brooks 85/ Brooks, R.A.: Aspetcs of Mobile Robot Visual Map Making, Robot Research, The Second International Symposium, 369-375, 1985

/Brooks 86/ Brooks, R.A.: A Robust Programming Scheme for a mobile Robot, proc. of the NATO International Advanced Research Workshop on languages for sensor-based control in robotics, Castelvecchio, Italy, September 1985

/Mc Carthy 69/ Mc Carthy, J.; Hayes, P.J.: Some Philosophical Problems from the Standpoint of Artifical Intelligence, Machine Intelligence, No. 4, 463 ff, 1969

/Chapman 85/ Chapman, D.: Nonlinear Planning : A Rigorous Reconstruction, IJCAI-85, 1022-1024, 1985

/Chattergy 85/ Chattergy, R.: Some Heuristics for the Navigation of a Robot, Int. Journal of Robotics Research, Vol. 4, No. 1, 59-66, 1985

/Crowley 84/ Crowley, I.L.: Navigation for an Intelligent Mobile Robot, proc. of the first conference on Artificial Intelligence applications, 79-84, 1984

/Davis 86/ Davis, L.S. et al.: Road Boundary Detection for Auto - nomous Vehicle Navigation, Optical Engineering, Vol. 25, No. 3, 409-414, 1986

/Dechter 85/ Dechter, R.; Pearl, J.: The Anatomy of Easy Problems: A Constraint-Satisfaction Formulation, IJCAI-85, 1066-1072, 1985

/Mc Dermott 80/ Mc Dermott, D.; Doyle, J.: Non-Monotonic Logic I, Artifical Intelligence 13, 41-72, 1980

/Mc Dermott 82/ Mc Dermott, D.: Non-Monotonic Logic II, Journal of ACM., Vol. 29, No. 1, 33-57, 1982

/Mc Dermott 84/ Mc Dermott, D.; Davis, E.: Planning Routes through Uncertain Territory, Artificial Intelligence 22, 107-156, 1984

/Doyle 79/ Doyle, J.: A Truth Maintenance System, Artificial Intelligence 12, 231-272, 1979

/Doyle 86/ Doyle, J.; Arkinson, D.; Doshi, R.: Generating Per - ception Requests and Expectations to Verify the Execution of Plans, proc. of the 5th AAAI-86 conference, 81-88, 1986

/Drummond 85/ Drummond, M.E.: Refining and Extending the Procedural Net, IJCAI-85, 1010-1012, 1985

/Forbus 86/ Forbus, K.D.: Interpreting measurements of physical systems, proc. of the 5th AAAI conference, Philadelphia, 113-117, 1986

/Fox 83/ Fox, M.S.: Constraint - Directed Search: A Case Study of Job-Shop Scheduling, CMU-Memo, RI-TR-83- 22, 1983

/Freedmann 85/ Freedmann, R.S.; Shooman, A.M.: Merlin: A Tool for Automated Inferencing with Uncertainty , proc. of the 2nd IEEE conference on AI applications, 22-26, 1985

/Freyberger 86/ Freyberger, F.; Kampmann, P.; Schmidt, G.: Ein wissenschaftliches Navigationsverfahren für autonome, mobile Roboter, Robotersysteme 2, 149-161, Springer Verlag, 1986

/Gilmore 86/ Gilmore, J.F.; Semco, A.C.: Knowledge-Based Approach Toward Developing, An Autonomous Helicopter System, Optical Engineering, Vol. 25, No. 3, 415-427, 1986

/Gini 86/ Gini, M.: Symbolic and Qualitative Reasoning for Error Recovery in Robot Programs, proc. of the NATO International Advanced Research Workshop onlanguages for sensor-based control in robotics, Castelvecchio, Italy, Sept. 1986

/Giralt 84/ Giralt, G.G., Chatilla, R., Vaisset, M.: An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots, Robotics Research, The First International Symposium, 191-214, 1984

/Giralt 85/ Giralt, G.: Research Trends in Decisional and Multi - sensory Aspects of Third Generation Robots, Robotics Research: The Second International Symposium, 511-520, MIT-Press, 1985

/Goodman 85/ Goodman, I.R.; Nguyen, H.T.: Uncertainty models for Knowledge-Based Systems, North-Holland, Amsterdam, 1985

/Harmon 84/ Harmon, L.: Tactile Sensing for Robots, in Robotics and Artificial Intelligence (eds.: M. Brady et al.) 109-157, Springer Verlag, 1984

/Harmon 84/ Harmon, S.Y.: Coordination of Intelligent Subsystems in Complex Robots , proc. of the 1th conf. on AI applications, 64-69, 1984

/Harmon 86/ Harmon, S.Y.; Bianchini, G.L.; Pinz, B.E.: Sensor Data Fusion Trough a Distributed Blackboard, proc. of the intern. conf. on robotics and automation, 1449-1454, 1986

/Hanks 86/ Hanks, S.; Mc Dermott, D.: Default Reasoning, Nonmonotonic Logics, and the Frame Problem, proc. of the 5th AAAI-86 conference, 328-333, 1986

/Hayes 85/ Hayes, P.: The Second Naive Physics Manifesto, in: Formal Theroies of the Commonsense World, 71- 108, Ablex, 1985

/Hayes-Roth 85/ Hayes-Roth, B.: A Blackboard Architecture for Control, Artificial Intelligence 26, 251-321, 1985

/Henderson 85/ Henderson, T.; Hansen, Ch.; Bhann, B.: A Framework for Distributed Sensing and Control, proc. of the 9th IJCAI, 1107-1109, 1985

/Iyengar 85/ Iyengar, S.S. et al.: Learned Navigation Paths for a Robot in Unexplored Terrain, proc. of the 2nd conference on Artificial Intelligence applications, 156-161, 1985

/de Kleer 86/ de Kleer, J.: An Assumption-based TMS, Artificial Intelligence 28, 127-162, 1986

/Kuan 84/ Kuan, D.T.; Brooks, R.A. et al.: Automatic Path Planning for a Mobile Robot Using a Mixed Representation of Free Space, proc. of the first conference on Artificial Intelligence Applications, 70-78, 1984

/Laumond 83/ Laumond, J.P.: Model Structuring and Concept Recognition: Two Aspects of Learning for a Mobile Robot, proc. of the 8th IJCAI, 839-841, 1983

/**Linden 86**/ Linden, Th.,A. et al.: Architecture and Early Experience with Planning for the ALV, proc. of the 1986-IEEE int. conf. on robotics and automation, Vol.3, 2035-2042,1986
/**Mamdani 85**/ Mamdani, A; Efstathion, J.; Pang, D.: Inference Under Uncertainty, proc. of the 5th technical conference of the British Computer Society (Expert Systems 85), 181-194, Cambridge University Press, 1985
/**Meystel 86**/ Meystel, A.: Nested Hierarchical Intelligent Module for Automatic Generation of Control Strategies, proc. of the NATO International Advanced Research Workshop on languages for sensor-based control in robotics, Castelvecchio, Italy, September 1986
/**Moravec 85**/ Moravec, H.P. et al.: Autonomous Mobile Robots, CMU-Annual Report of the Robotics Institute, CMU, 1985
/**Morris 86**/ Morris, P.H., Nado, R.A.: Representing Actions with an Assumption-Based Truth Maintenance System, proc. of the 5th AAAI-86, 13-17,1986
/**Newman 85**/ Newman, P.A.; Kempf, K.G.: Opportunistic Schedu - ling for robotic machine tending, proc. of the 2nd IEEE conference on Artificial Intelligence applications,168-175, 1985
/**Orlando 84**/ Orlando, N.: An Intelligent Robotics Control Scheme, proc. of the con. on Automated Control, San Diego, 204-208, 1984
/**Ow 86**/ Ow, P.S., Smith, S.F.: Viewing Scheduling as an Opportunistic Problem-Solving Process, CMU- internalreport, May 1986
/**Parodi 86**/ Parodi, A.M., Nitao, J., Mc Tameney, L.: An Intelligent System for an Autonomous Vehicle, proc. of the IEEE intern. conf. on robotics and automation, Vol.3, 1657-1663, 1986
/**Payton 86**/ Payton, D.W.: An Architecture for Reflexive Autonomous Vehicle Control, proc. of the IEEE intern. conf. on robotics and automation, Vol.3, 1838-1848, 1986
/**Pearl 85**/ Pearl, J.: How to do with probabilities what people say you cant, proc. of the 2nd IEEE conference on AI applications, 6-12, 1985
/**Pearson 85**/ Pearson, G; Kuan, D.: Mission Planning System for an Autonomous Vehicle, proc. of the second conference on Artificial Intelligence applications, 162-167, 1985
/**Reiter 80**/ Reiter, R.: A Logic for Default Reasoning, Artificial Intelligence 13, 81-132, 1980
/**Rembold 85**/ Rembold, U.; Dillmann, R.: Artificial Intelligence in Robotics, proc. of the SYROCO 85, Barcelona, 7-9 Sept., 1985
/**Rembold 86**/ Rembold, U.; Levi, P.: Sensors and Control for Autonomous Robots, proc. of the conference on Intelligent Autonomous Systems, 8-11 December, Amsterdam, 79-95, 1986
/**Sacerdoti 75**/ Sacerdoti, E.D.: Planning in a Hierarchy of Abstrac - tion Spaces, IJCAI-75, 412 ff, 1975
/**Sacerdoti 77**/ Sacerdoti, E.D.: A Structure for Plans and Behaviour, Elsevier/North-Holland, New York, 1977
/**Sathi 85**/ Sathi, A.; Fox, M.S.; Greenberg, M.: Representation of Active Knowledge for Project Management, IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-7, No. 5, 531-552, 1985
/**Schmolze 86**/ Schmolze, J.G.: Physics for Robots, proc. of the 5th AAAI-86 conference, 44-50, 1986
/**Shafer 86**/ Shafer, S.A. et al.: An Architecture for Sensor Fusion in a Mobile Robot, proc. of the 1986 IEEE intern. conf. on robotics and automation, Vol. 3, 2002-2011, 1986
/**Shapiro 85**/ Shapiro, L.G.: The Role of AI in Computer Vision, proc. of the 2nd IEEE conference on AI applications, 76-81, 1985
/**Soetadji 86**/ Soetadji, T.: A Cube-Based Approach to Find-Path for an Autonomous Mobil Robot, proc. of the NATO International Advanced Research Workshop on languages for sensor-based control in robotics, Castelvecchio, Italy, Sept. 1986
/**Stefik 81**/ Stefik, M.J.: Planning with Constraints (MOLGEN : Part 1), Artificial Intelligence 16, 111 ff, 1981, Planning and meta-planning (MOLGEN: Part 2), Artificial Intelligence 16, 141 ff, 1981
/**Stuart 85**/ Stuart, C.: An Implemetation of A Multi-Agent Plan Synchronizer, IJCAI 85, 1031-1033, 1985
/**Sussman 80**/ Sussmann, J.; Steele, G.L.: Constraints - A Language for Expressing about Hierarchical Descriptions, Artificial Intelligence 14, 1-39, 1980
/**Tate 77**/ Tate, A.: Generating Project Networks, , IJCAI-77, 888 ff, 1977
/**Tate 85**/ Tate, A.: A Review of Knowledge-Based Planning Tech - niques, proc. of the 5th technical conferencef the British Computer Society (Expert System 85) 89-111, Cambridge University Press, 1986
/**Turchan 85**/ Turchan, M.P.; Wong, A.: Low Level Learning for a Mobile Robot: Environment Model Acquisition, proc. of the 2nd conference on Artificial Intelligence Applications, 156-161, 1985
/**Vilain 86**/ Vilain, M.; Kautz, H.: Constraint Propagation Algorithmus for Temporal Reasoning, proc. of the 5th AAAI-86 conference, 377-382, 1986
/**Wallace 86**/ Wallace, R. et al.: Progress in Robot Road Following, proc. of the 1986 IEEE international confeerence on robotics and automation, 1615-1621, 1986