

A MULTI-LEVEL PLANNING AND NAVIGATION SYSTEM FOR A MOBILE ROBOT;
A FIRST APPROACH TO HILARE

Georges Giralt, Ralph Sobek*, and Raja Chatila

Laboratoire d'Automatique et d'Analyse des Systèmes du CNRS
7, avenue du Colonel-Roche
31400 Toulouse - France

This paper describes the current state of HILARE: a modular progressively-built mobile robot aimed at general robotics research. The computer organization comprises of local mini and micro processors coupled with a remote time-shared system acting as a consulting facility. A multi-level decision-making system is presented with world models, inference rules, and algorithms particular to each level. The navigation planner uses a geometric model wherein 2-space is partitioned into polygonal areas based on perceptual and/or initial information. A cost function is proposed which provides support for optimal or ϵ -optimal path finding.

1. GENERAL DESCRIPTION AND PURPOSE OF HILARE

The HILARE** project started in Sept. 1977 as an attempt to provide the LAAS robotics group, as well as several other teams interested in that field in Toulouse, with a flexible and powerful experimental support for advanced research work on Robotics: a particular blend of Artificial Intelligence, Computer Science, Control Theory, and Instrumentation.

The choice of an autonomous mobile robot was made mainly for two reasons: 1) capacity to provide a variety of problems at different levels of generality and difficulty in a large domain - including perception, learning, decision-making, communication, etc., which all have to be considered within the scope of the specific constraints of robotics: on-line computing, cost considerations, operating ability and reliability; and 2) possibility to design a modular system which could be incrementally built and yet providing at every step for interesting and useful research in many directions.

Research goals were programmed in these domains in accordance with the development of the system configuration. We emphasize that HILARE has no practical purpose in itself yet we are deeply concerned in every stage of our work with methods and techniques, software and instrumentation for advanced applied robotics.

* Presently at LAAS; visiting researcher from the University of California, Berkeley, Calif. Visit made possible in part by the National Science Foundation Grant INT78-09263.

** Heuristiques Intégrées au Logiciel et aux Automatismes dans un Robot Evolutif.

To date the system architecture (see Fig. 1) consists of (cf. /1/): 1) Perception: video camera and laser range-finder mounted on a two-axes scanning system and integrated to provide fast rough scene analysis consistent with navigation needs; 2) Locomotion: one free front wheel and two motorized rear drive wheels; the microprocessor controlled step motors allow for a variety of speed and trajectory commands: straight lines, arcs with specified angular shift and rotation center; and 3) Computer System.

One of the more relevant characteristics which will aid advanced robotics with real-world constraints, such as cost-effectiveness, is an appropriate computer system structure. Our organization (being investigated in several projects at LAAS) consists of 1) local distributed computing facilities coordinated by a 16-bit MITRA 15 minicomputer, and 2) a remote time-shared IBM 370/168 system. Local facilities include on-

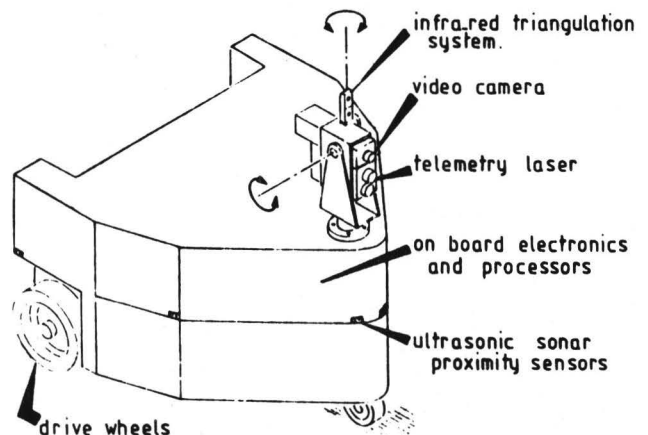


Fig. 1 - Present Configuration of HILARE

board specialized processing units (e.g., image preprocessing) together with 3 Intel 8085 micro-computers. The MITRA 15 supports executive, decision, and communication functions. The remote computer is programmed as a consultant system (see §2); it provides powerful resources in tasks such as learning, reprogramming, procedure optimization, large data-base consulting, etc.

2. DISTRIBUTED DECISION-MAKING

In the system design of HILARE we have decided on a distributed decision-making capacity which allows robustness, efficiency, economy of operation and timely system response. Our design can be viewed as decisions through multiple cooperating expert modules/5/ together with a high-level coordinator in a hierarchical means-ends structure.

The expert modules have their expertise in a variety of overlapping domains (e.g., object identification, navigation, exploration, itinerary planning). The modules consist of 1) specialized and redundant knowledge bases, 2) algorithms and heuristics, 3) local error-processing capabilities, and 4) communication procedures. Modules may access one another as primitive action-units. The coordinator activates modules based on a means-ends analysis/2/ of the current situation.

2.1 High-Level Decision-Making and Planning

We are designing and programming a relational-level planner at an advisor level. A plan at this level is a flexible dynamic structure which coordinates the achievement of desired goals/8/. Goal understanding includes 1) acknowledging or refusing an user order, 2) handling temporal and logical constraints, and 3) finding action-units which can achieve goals/7/. The planner must also include coordination of error processing, broadcast, and recovery procedures/9/.

Action-units may be primitive or macro actions (viz., entry-points into expert modules) /3/. The planner's required knowledge about actions includes limiting constraints and activation, continuation, and termination conditions.

Due to the incremental and open-ended nature of HILARE's design we consider Production Systems (PSs) a viable research tool for relational-level problem-solving. Thus, we are writing the planner in terms of a PS architecture in the spirit of /6/.

2.2 Relational-Level World Model

The world model is contained in the PS data base: embedded non-recursive list structures which are associatively accessed by the pattern matcher during rule matching. The model is a hierarchy of object-centered concepts of the form

(name feature pattern-body)

which describe space and objects. Space is defined by places (rooms, work-areas), frontiers (doors), and locations. Places have the property of surface connexity and are connected to other places by means of frontiers. Locations are elements within places which can be identified by point coordinates, relational descriptions, or feature descriptions.

Objects are defined by features (e.g., shape, color, and dimension) based on sensory perception. We define macro-objects by recursive assembling and aggregation of objects.

3. NAVIGATION

In this section we consider the basic problem of moving a robot from an initial state (R) to the target (G) within a given place. This involves obstacle avoidance, path-finding, and search trajectory minimization/4,10/.

3.1 Navigation World Model

Currently, obstacles are defined as polyhedra, whose floor projections fully determine the navigation problem, and they can be located either by initial information or by robot perception (see Fig. 2). Each obstacle projection is represented as an ordered list of segments in counter-clockwise sequence: e.g.,

$$Obs_1\{S_1^1, S_2^1, S_3^1, S_4^1\}$$

A segment is coded using the Cartesian coordinates of its leftmost point(X,Y), its angle with a reference axis(ϕ), and its length(r):

$$S_i\{<X,Y>, \phi, r\}$$

Empty areas are defined as convex polygonal cells which include obstacle segments. A trajectory within such cells can be considered as a straight line between entry and exit segments. In such cells ordering of segments S_i is clock-

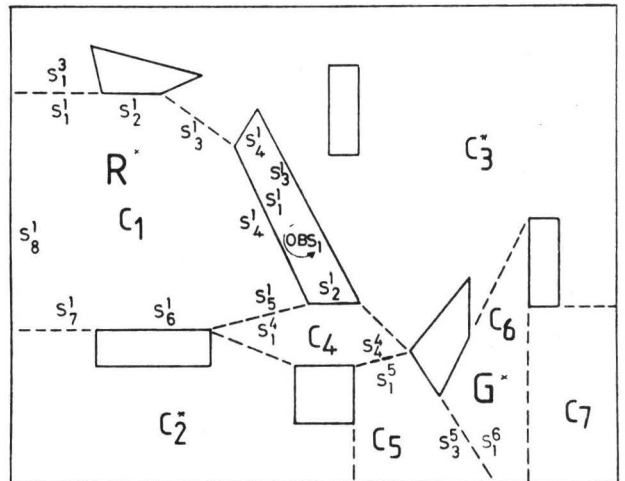


Fig. 2 - Cellular Decomposition of a World

wise, e.g., $C_1 \{S_1^1, S_2^1, \dots, S_8^1\}$.

Our partition algorithm which divides a known world into C-cells and obstacles is as follows. Starting at any vertex a C-cell is defined by following segments in a clockwise order as long as the angle between two successive segments is acute. When not, the algorithm jumps to the next visible segment, such that an acute angle is formed and a convex cell is possible, and continues until a convex polygon is formed.

Any area including possibly unknown obstacles is defined as a polygonal *-cell, which is allowedly non-convex, e.g., $C_3^* \{S_1^3, S_2^3, \dots, S_{15}^3\}$.

The topological properties of a place - divided into C-cells, *-cells, and obstacles - are graph represented. Two adjacent cells have connectivity through common segments which are traversable by the robot. Thus, a connectivity graph (see Fig. 3) provides the structure necessary for path finding.

3.2 Decision Procedure

The example in Fig. 2 produces among possible paths a single path between R and G which is completely determined since it does not include *-cells: $C_1(R) \xrightarrow{S_1^1} C_4 \xrightarrow{S_4^4} C_5 \xrightarrow{S_5^5} C_6(G)$. In most cases several paths would exist and a choice would be made through ϵ -optimality concepts.

We define our cost function as: $F = \sum_i (F_i^1 + F_i^2)$; i = cell index. F_i^1 measures every cost associated with determined trajectory travel. To minimize distance and energy requirements we have:

$F_i^1 = \omega_1 d_i + \omega_2 |\phi_i| + \omega_3 n_i$, where for cell i 1) d_i is path length, 2) $|\phi_i|$ is the angle of planned direction change, 3) n_i is the number of predicted robot stops, and 4) ω_i are weights.

F_i^2 measures uncertainty and is evaluated over *-cells: $F_i^2 = \omega_4 d_i (I \cdot V - 1)$, where I is a decreasing positive function of the cell information which can be obtained by the robot at the entry segment: $I = S_T / S$, and V measures path viability

due to estimated obstacle clustering: $V = 1/(1-\tau)$ S is the maximal perceived area at the considered entry and S_T is the total *-cell surface. τ is the estimated ratio of aggregated obstacle surface over the total *-cell surface (S_T).

The minimization of F provides a method for path-finding within a world which can be stepwise perceptually identified by the robot.

In the case of partially known worlds cell construction is defined by the perceptual system as HILARE discovers its world. It may occur that non-traversable narrow cells are created; whereas another partitioning would allow traversal.

3.3 Object Close-Distance Navigation

Consistent with our viewpoint on multi-level decision-making the problem of navigation can be resolved more efficiently with the introduction of a decision level which takes into consideration a variety of navigation conditions which call for on-board closed-loop control, such as narrow corridor navigation, etc. This control is performed by HILARE using 10 close-range ultrasonic sensors regularly distributed over its surface. The higher decision levels including navigation set the objectives and the constraints for close-distance navigation with take-over and restitution of control locally defined.

REFERENCES

1. Briot, M., G. Bauzil, and R. Chatila. "Présentation d'HILARE, un Robot Mobile." Journées d'Automatique de l'IRISA, Rennes (Jan. 1979).
2. Fikes, R. and N. Nilsson. "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving." *Art. Int.* 2:3/4 (1971) 189-208.
3. Fikes, R., et al. "Learning and Executing Generalized Robot Plans." *Artificial Intelligence* 3:4 (1972).
4. Hart, P., et al. "Artificial Intelligence -- Research and Applications." Annual Technical Report, Contract DAHCO4-72-C-0008. SRI (Dec. 1972).
5. Lenat, D. "Beings: Knowledge as Interacting Experts." In *Proc. of IJCAI-75*. Tbilisi, USSR (Sept. 1975) pp. 126-133.
6. Rychener, M. "Production Systems as a Programming Language for Artificial Intelligence Applications." Ph.D. Thesis, Computer Science Dept. Carnegie-Mellon University (Dec. 1976).
7. Sacerdoti, E. *A Structure for Plans and Behavior*. New York: Elsevier Publishing Co., 1976.
8. Sobek, R. "Automatic Generation and Execution of Complex Robot Plans." Master's Report. EECS Dept., Univ. of Calif., Berkeley (Sept. 1975).
9. Srinivas, S. "Error Recovery in Robot Systems." Ph.D. Thesis. Computer Science Dept., California Institute of Technology (1977).
10. Thompson, A. "The Navigation System of the JPL Robot." In *Proc. of IJCAI-77*. Cambridge, Mass. (Aug. 1977) pp. 749-757.

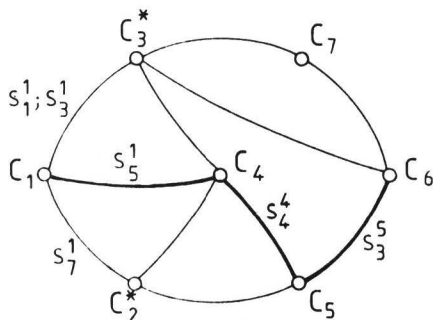


Fig. 3 - Connectivity Graph