第5卷 第2期
1988年6月

控制理论与应用
CONTROL THEORY AND APPLICATIONS

Vol. 5, No. 2
June, 1988

Expert-System-Based Robot Planning

Cai Zixing

(Department of Automatic Control Engineering, Central South University of Technology, Changsha)

King-Sun Fu

(School of Electrical Engineering, Schools of Engineering, Purdue University, USA)

Abstract

A new high-level robot planning system, the expert-system-based robot planning system, is developed and introduced. The simulation and comparison show that this system can improve the planning performances both in the capability and the planning speed.

1. Introduction

Up to present, several different high-level robot planning systems have been developed. Some of them[1-2] focused on the resolution theorem prover, using the general purpose search heuristics, with the desired goal expressed in terms of logical calculus. Another existing planning system has used the supervised learning to speed up the planning process and to improve its problem-solving capability. The PULP-I[3] is a typical robot planning system with learning; it is based on the concept of analogy. Several other planning systems, including the nonlinear planning, the planning applying deduction, and the hierarchical planning and so on, have been developed and used in the recent years. All of these systems have some limitations. One problem is that they can not be applied to the large system with many independent subproblems. Another one is that they can not distinguish the major goal from the general goals. As a result, a large amount of time and the internal memory may be spent in considering a part of the planning in detail.

Recently, the expert systems have been applied to robot plan-

---

Manuscript received Feb. 27, 1986, revised Dec. 31, 1987.

ning in the various levels[4-6]. In this paper, we would like to introduce some research work that we have done in this area.

2. Expert System and Robot Planning

We want to build up some more generalized and advanced systems for robot planning, and call them as ROPES, the RObot Planning Expert Systems, using the rule-based expert system and C-PROLOG language. The simplied block diagram of ROPES is constructed in Fig. 1.

For building an expert system, one has to acquire the expertise carefully and correctly. The acquired expertise is stored in a knowledge base. In order to build the knowledge base in a form of the computer representation, a knowledge engineer should discuss the expertise with the experts or/and read some related references, etc.

The basic control strategies for the ROPES are SEARCHING, MATCHING and BACKTRACKING when using the PROLOG language[7,8].

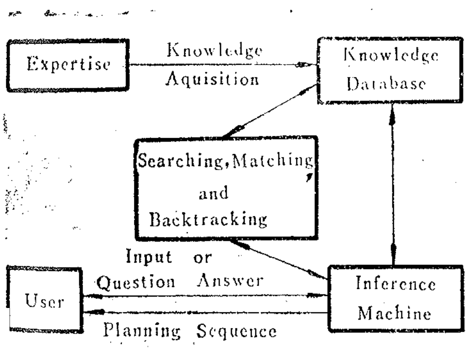

Fig. 1 Major architecture of the ROPES

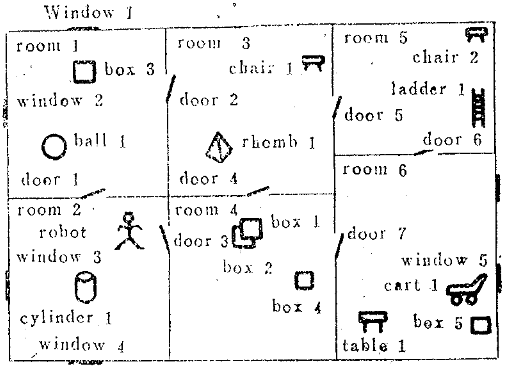

Fig. 2 Initial world model of PULP-24

3. Hypotheses and Rules

Fig. 2 shows the initial world model of the PULP-24, a subsystem of the ROPES system. We suppose that every room has its doors connecting to its neighbours' room. Every object has its features such as the size, the color, and the initial position and situation. We store all these data into the knowledge base. If the object to be moved is a heavy one, or there are two or more objects to be moved, then the system uses a cart. Otherwise, the cart is not necessary.

There are 40 rules to be applied in the PULP-24. Using these rules with the build-in evaluable predicates of C-PROLOG, the system can output the reasoning results quickly.

If there are more than one solutions, then the system can output every possible result.

The following are some further hypotheses related to the PULP-24: go (A, B), go from location A to location B, where

A=(roomA, Xa, Ya), at position (Xa, Ya) in room A;
B=(roomB, Xb, Yb), at position (Xb, Yb) in room B;
(Xa, Ya), the horizontal and vertical coordinates of the Cartesian coordinate system in room A;
Xb, Yb similar to Xa and Ya, but in room B.

gothru(A, B), go through a door from location A to location B.
push (A, B), push an object from A to B.
pushthru(A, B), push an object through a door from A to B.
move(A, B), move a heavy object from A to B.
movethru(A, B), move a heavy object from A to B through door by a cart.
open(X), open a door or a window X.
close(X), close a door or a window X.
load(M,N), load a heavy object or a group of objects M on cart N.

According to the above hypotheses, the world model and the related knowledge, we can establish the inference rules. The following is an example of rules, RULE 10, which describes how the robot moves a cart from a room to its diagonal room where an object to be moved is located, and how this object will be placed on the cart.

```
RULE 10
move((Rc,(Xc, Yc)), D2, R3, D5, R5, D6,(Rb,(Xb, Yb)))
  at2(cart1, Rc, (Xc, Yc)), at (Object, Rb, (Xb, Yb)),
  ((connects (D2, Rc, R3), connects (D5, R3, R5),
  (connects (D6, R5, Rb); connects(D6, Rb, R5)));
```

```
    (connects (D2, R3, Rc), connects (D5, R5, R3),
    (connects (D6, Rb, R5); connects (D6, R5, Rb)));
    (connects (D2, R3, Rc), connects (D5, R5, R3),
    (connects (D6, Rb, R5); connects (D6, R5, Rb)));
    (connects (D2, Rc, R3), connects (D5, R5, R3), connects
(D6, Rb, R5));
    (connects (D2, R3, Rc), connects (D5, R3, R5), connects
(D6, R5, Rb))),
    diff (Rc, R3), diff (R3, R5), diff (R5, Rb),
    ((Rb= =room1, Rc= =room6); (Rb= =room6, Rc= =room1);
    (Rb= =room2, Rc= =room5); (Rb= =room5, Rc= =room2)).
sequence((move ((Rc, (Xc, Yc)), (Rc, (X2, Y2))),
          movethru (D2, (Rc, (X2, Y2)), (R3, (X3, Y3))),
          move((R3, (X3, Y3)), (R3, (X4, Y4))),
          movethru (D5, (R3, (X4, Y4)), (R5, (X5, Y5))),
          move((R5, (X5, Y5)), (R5, (X6, Y6))),
          movethru (D6, (R5, (X6,Y6)), (Rb, (X7, Y7))),
          move((Rb, (X7, Y7)), (Rb, (Xb, Yb))),
          loadon (Object, cart1)), robot) :-
move ((Rc, (Xc, Yc)), D2, R3, D5, R5, D6, (Rb, (Xb, Yb))), at
(Object, Rb, (Xb, Yb)), at6 (D2, Rc, (X2, Y2)), at6 (D2, R3,
(X3, Y3)), at6 (D5, R3, (X4, Y4)), at6 (D5, R5, (X5, Y5)), at6
(D6, R5, (X6, Y6)), at6 (D6, Rb, (X7, Y7)).
change (R1, Rb).
change ((X1, Y1), (Xb, Yb)).
```

Those who are familiar with PROLOG language will not have any difficulty in understanding the above rules.

4. Executing Example

Fig. 3 shows a solution graph for a task, task 18, from which we can find $3 \times 3 \times 3 = 27$ solutions. The total operators of this planning are 117, and the CPU time is 3. 5167 seconds.

There are still more examples including two assembly systems, three collision-avoidance systems and one machine part transfer system, etc.. We have discussed some of them in the other papers[7-9],

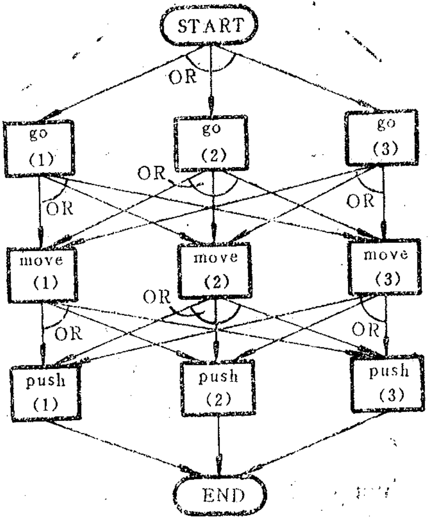

Fig. 3 Solution graph for task 18 of PULP-24

5. Comparison

The ROPES was implemented in C-PROLOG on the DUAL-VAX 11/780 computer and VM/UNIX operating system. Whereas the PULP-I was executed using interpreted LISP on the CDC-6500 computer, the STRIPS and ABSTRIPS were partially compiled in LISP excluding garbage collection on a PDP-10 computer. In fact, the CDC-6500 is estimated to be about 8 times faster than the PDP-10. However, owing to the advatage of partial compilation and the exclusion of the garbage collection of PDP-10, the CDC-6500 gains slightly in speed of processing[3]. The DUAL-VAX 11/780 and VM/UNIX systhm are slower than the CDC-6500. In order to simplify the comparison, we handle the four systems using the same computer time unit, and make a direct comparison on them.

Table 1 shows the complexity of the four systems. It is clear that the ROPES (PULP-24) system is the most complex one, the PULP-I is the next, but more complex than the STRIPS and ABSTRIPS family.

The planning speed of the four systems is plotted in Fig. 4 using logarithmic coordinate. From the curves we can see that the planning speed of PULP-I is much faster than that of the STRIPS

Table 1 Comparison of the world model

planning system	object number				
	rooms	doors	boxes	others	total
STRIPS	5	4	3	1	13
ABSTRIPS	7	8	3	0	18
PULP-I	6	6	5	12	27
PULP-24	6	7	5	15	33

and ABSTRIPS.

Table 2 shows the speed of the PULP-I and PULP-24 in detail. From Fig. 4 and Table 2 we know that the ROPES (PULP-24) is even faster than the PULP-I system. One reason for this is due to the explanation mode of PROLOG language that is not necessary to write a complex reasoning and control program. However, when using LISP language, one may spend a lot of time and computer memory to write, store and execute its program.

Table 2 Planning time

number of operators	CPU time (second)	
	PULP-I	PULP-24
2	1.582	1.571
6	2.615	1.717
10	4.093	1.850
19	6.511	1.967
26	6.266	2.150
34	12.225	—
49	—	2.767
53	—	2.950
62	—	3.217
75	—	3.233
96	—	3.483
117	—	3.517

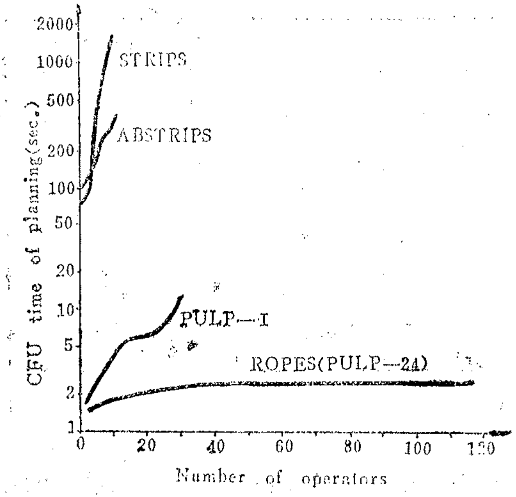

Fig. 4 Comparison of planning speed

6. Conclusion

(1) The expert system can be successfully used to the high level robot planning, it is easy for designers and convenient for users.

(2) The ROPES system can improve the planning performances both in the capability and the planning speed.

(3) ROPES can give out the multiple or a single solution(s).

(4) The PULP-24 are stitable for large scale planning systems.

References

[1] Fikes, R. E., Nilsson, N. J., STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving, 2nd IJCAI, (1971), 608-620.

[2] Fikes, R. E., et al, Learning and Executing Generalized Robot Plans, Artificial Intelligence, 3, (1972), 251-288.

[3] Tangwongsan, S., Fu, K. S., An Application of Learning to Robot Planning, Int. J. Computer and Information Sciences, 8: 4, (1979), 303-333.

[4] Wu, M. C., Liu, C. R., Automated Process Planning and Expert System, Proc. IEEE 1985 Int. Conference on Robotics & Automation, (1985), 186-191.

[5] Gevarter, W. B., et al, An Expert System for the Planning and Scheduling of Astronomical Observations, Proc. ICR&A, (1985), 221-226.

〔6〕 Cai, Z., Some Research Works on Expert System in AI Course at Purdue, Proc. IEEE 1986 Int. Conference on Robotics and Automation, (1986), 1980-1985.

〔7〕 Cai, Z., An Expert Planning System for Transfer Operation in Robot Assembly Line, 1986 National Univ. Symposium on AI, China, (1986).

〔8〕 Cai, Z., A Method for Finding the Robot Path with Collision-Free, IASTED 1986 International Conference on Applied Control and Identification, Los Angeles, (1986).

〔9〕 Cai Z., Fu, K. S., Robot planning Expert Systems, Proc. IEEE 1986 Int. Conference on Robotics and Automation, (1986), 1973-1978.

应用专家系统的机器人规划

蔡自兴 傅京孙

（中南工业大学自动控制工程系，长沙） （美国普渡大学工学院电气工程系）

摘 要

本文介绍一个新开发的高层机器人规划系统——基于专家系统的机器人规划系统。模拟及比较结果表明，该系统既能改善规划性能，又可加快规划速度。