Control program structure of Humanoid Robot

Shigeto Aramaki*1, Hiroshi Shirouzu*2, Kentarou Kurashige*1,Taketoshi Kinoshita*1

*1) Faculty of Engineering Fukuoka University *2) Kyushu Matsusita Electric Co.,LTD.

key words: humanoid, control program, parallel processing, production system

Abstract

In this paper, the new control program structure for humanoid robot which consists of multi-CPU, multi-OS and multi-task is described. And, the method for planning action from multiple sensor information by using the Production system is also described. Task program has hierarchical structure like as human being has conscious action and unconscious action. Our design concept is generality of task. We want to make a control program independently on the architecture of a robot and other tasks. We also want to make software architecture simple. Task program is divided into three levels. Tasks must communicate each other. However, if one task can communicate another task at random, the amount of communication increases exponentially. Therefore, we made a rule of task in order to make the amount of communication linearly increase. On the other hand, a robot must determine next action from multiple sensor information. We introduced the reasoning mechanism of Production system for determining next action. It becomes easy to determine next action from multiple sensor input by using this method. We could make a robot grasp a block by using this method and simulator, and we confirmed the effectiveness of methods mentioned above.

I. Introduction

Now, the humanoid robot has been developed in many research institutes [1]-[5]. The development of the integration mechanism of action and sense has been also carried out[6]. However, the algorithm which decides the action by processing the multiple sensor information has not been developed so much.

We built the humanoid robot which behaves the same in the space as the human[7]. The mechanism and functions, as vision sensor, acoustic sensor and loudspeaker, are mounted on this robot in order that the multi-modal bidirect communication between a robot and the human is possible. We adopted the parallel process by multi-CPU and multi- OS. The reason is that we can develop independently the software of each function (vision sensor etc.) and the amount of computer process increases according to the high intelligence of a robot in the future.

In this paper, the new control program structure for humanoid robot which consists of multi-cpu, multi-OS and multi-task is described. Moreover, the method for planning action from multi sensor information by using the Production system is also described.

II. Structure of Control Program

A. Hierarchy

Generally speaking, human being does two actions. One is conscious action. The other is unconscious action. For example, global planning and error recovery etc. are conscious actions. On the other hand, simple routine work and conditioned reaction are unconscious actions. I think that driving a car is unconscious action because we can drive it while thinking other things. The speed of unconscious action is faster than that of conscious action.

We are studying based on this design approach. Design concept is generality of task. We want to make a control program independently on the architecture of a robot

0-7803-7474-6/02/$17.00 ©2002 IEEE

and other tasks. We also want to make software architecture simple. Our software system has following hierarchical structure. Task program is divided into action task and hardware control task. Action task is also divided into conscious task and unconscious task. The software hierarchical structure is shown in Fig.1.

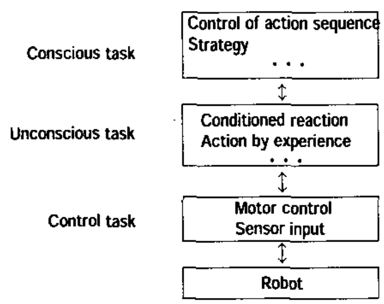

Fig.1 Software structure

The conscious task sends the task command to unconscious task. Unconscious task analyzes it and sends the result to control task. Control task program makes a robot move. Global planning, control of action sequence and error recovery etc. belong in the conscious task. And unconscious action, action by experience and conditioned reaction etc. belong in the unconscious task. The control task program is dependent on the architecture of a robot. On the other hand, Conscious task and unconscious task program are independent on the architecture of a robot.

A. Realization method of tasks

Tasks must communicate each other. But, if one task can communicate another task at random, the amount of communication increases exponentially. Therefore, we made a rule of task in order to make the amount of communication linearly increase and keep the hierarchy characteristics. The task which exists in upper level of hierarchy is called "parent task". The task which exists in lower level is called "child task". The child task is executed by the parent task. The parent task has multiple child tasks and the child task has only one parent task. The parent task can send "event" to the child task and the same level tasks of hierarchy. These tasks can add necessary data with event. When the task received event, "acknowledge" is sent to the sender task of event with necessary data if needed. "Message" can be used when the child task sends data to the upper tasks of hierarchy. Therefore, the child task cannot communicate the specified parent task directly. Therefore, "Post-office" is used to solve this. When the parent task wants to acquire the event information from the child task, the parent task must register at "Post-office". This event information is called "message". This relation is shown in Fig.2.

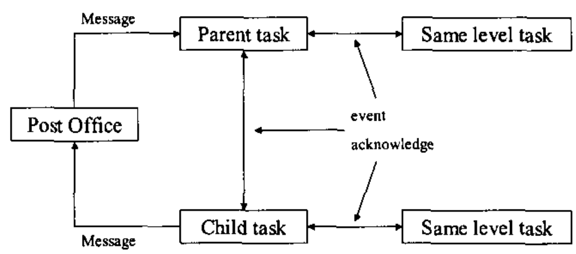

Fig.2 Communication between tasks

Though the control level of hierarchy depends on OS and hardware, the conscious and unconscious level do not depend on them. Moreover, the communication among each task does not also depend on OS because these tasks communicate by standard LAN protocol. Humanoid robot consists of Multi-cpu, Multi-OS and Multi-task. Therefore, "Supervisor computer" is used because it is necessary to control individual

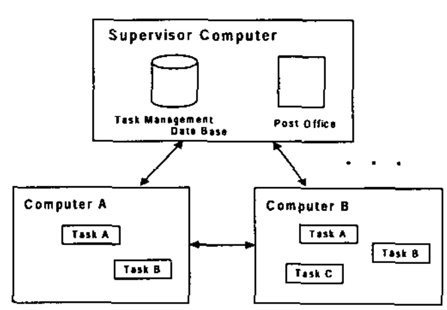

Fig.3 Supervisor computer

task. "Supervisor computer" manages the information of individual task as Task Management Data Base(TMDB) (Fig.3). Post-Office also exists in this computer. By using this computer, it becomes easy to start, terminate another task, get the other tasks information and send the data to another task.

C. Task example(Grasping a block)

The example of grasping a block task is shown in this section. The step of grasping a block is as follows.(Fig.4)

(1) Find the position of block
(2) Command for arm movement
(3) Move a finger to block position
(4) Confirmation of final finger goal position
(5) Grasp a block

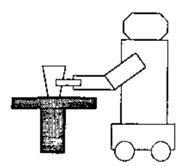

Fig.4 Grasping a block

Task hierarchy for grasping a block is shown in Fig.5.

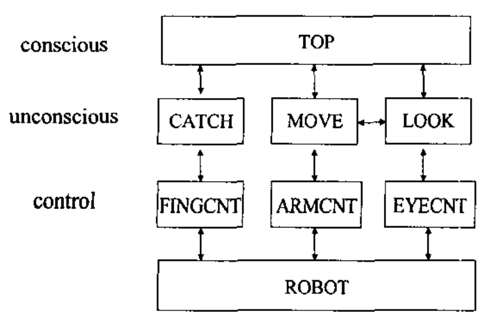

Fig.5 Task Hierarchy

TOP task manages the sequence of grasping a block. CATCH controls finger action. MOVE control arm. LOOK controls eye motion and observe a object. In this case, MOVE and LOOK is executed cooperatively. FINGCNT, ARMCNT and EYECNT really make finger, arm and eye move respectively.

For example, the state transition of LOOK task is shown in Fig.6.

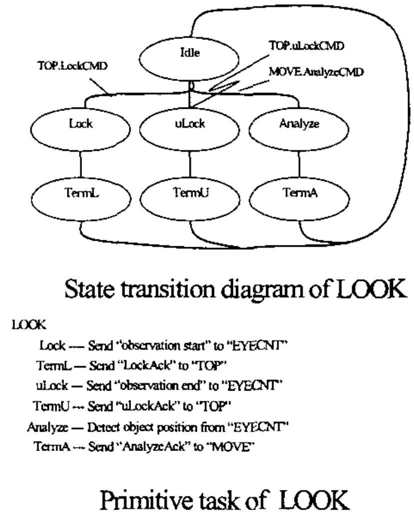

Fig.6 State transition of LOOK task

LOOK task consists of many primitive tasks. Primitive tasks return to idle state after ending its task.

For example, Lock task send "observation start" to "EYECNT". Next, TermL send "LockAck" to "TOP" and the state returns to Idle state. Other primitive tasks do similarly.

D. Planning by Production system

A robot perceives the state of environment and acts according to it. We suppose the planning method by using Production system. This system is embedded in the conscious task. Fig.7 shows the detail structure of planning.

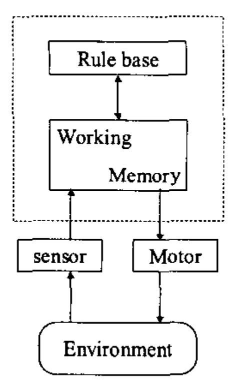

Fig.7 Production system

Table 1 Relation of sensor value and action

sensor \ action			Home	Reaching	Grasp	Pick up	Move	Put down	Release	Move up (goal)
Vision	Separation Of Hand and object	Overlap		O	O	O	O	O	O	
		Separate	O							O
	Movement Of Object	Move				O	O	O		
		Stop	O	O	O				O	O
	Destination Position Of Object	Destination					O	O	O	O
		No destination	O	O	O	O				
Wrist	Reaction	Reaction				O	O	O		
		free	O	O	O				O	O
Finger	Reaction	Reaction			O	O	O	O		
		free	O	O					O	O

A robot action is determined by sensory information. Humanoid robot has many sensor. Therefore, each set of many sensor values is written at Condition part of Production rule.

Production Rule:

If (set #1 of sensor values)

then (Robot action #1)

. . .

. . .

New sensor value is rewritten in the Working memory. Production rule is stored in the Long term memory(Rule base). The example of application for Pick-and-place task is shown as following.

Table 1 shows the relation between sensor states and action for pick-and-place. Each sensor state shows the result of each action. This sensor state determines next robot action. This process continues until goal (move up) by Production system.

III. Experiment

We made experiment in order to confirm the effectiveness of control architecture which we suppose. We made a robot grasp the block. Fig.8 shows the simulator experiment. The starting process for experiment is shown as follows.

① Starting of Supervisor facility(TMDB etc).
② Writing task information on TMDB.
③ Starting each computer.
④ Execusion of TOP task.
⑤ Each task(child task etc) automatically is started.

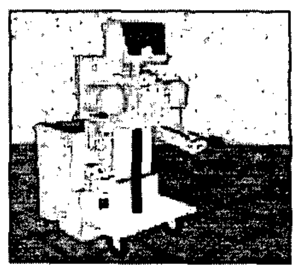

Fig.8 Simulator Experiment

IV. Conclusion

The control program structure of the robot which consists of multi-CPU and multi-OS was described. The advantage of the hierarchy task structure is to be the algorithm which does not choose the platform. The task program can be used without drastic modification again for other robot tasks. It becomes easy to make robot program by using this method, for example, just like as the program for grasping a moving ball is made by replacing a task without drastic program modification. On the other hand, the method for determine robot action by using the reasoning mechanism of Production system storing sensor values in the working memory, was also described. It becomes easy to determine next action from multiple sensor input if this method is used. Now, we are making experiment using real robot. In the future, we are going to make a robot various kind of tasks using this method.

Acknowledgment

The authors gratefully acknowledge the contributions of Hiroaki Tanaka.

References

[1] R.A.Brooks and L.A.Stein, "Building Brains for Bodies", MIT AI Lab Memo#1439, Aug,1993.

[2] A.Takanishi, "Human Communication Oriented Humanoid Robot", J.of the Robotics Society of Japan, Vol.15, No.7, pp.11-14, 1997.

[3] M.Inaba, "Remote-Brained Robots", Proc. of IJCAI-97, pp.1593-1606, 1997.

[4] Y.Kuniyoshi and A.Nagakubo, "Humanoid as a Research Vehicle Into Flexible Complex Interactions", Proc. of 10th IEEE Int. Conf. on Intelligent Robots and Systems, 1997.

[5] K.Hirai, "Current and Future Perspective of Honda Humanoid Robot", IROS'97, IEEE/RSJ, pp.500-509, 1997.

[6] R. Murphy, "Introduction to AI ROBOTICS", MIT Press, 2000.

[7] S.Aramaki et al, "Development of Autonomous Mobile Humanoid Robot", Proc. of IECON'99, pp.529-534, 1999