

HOLONIC MANAGEMENT SYSTEM FOR HIERARCHICAL ROBOT GROUPS

Yasumichi Aiyama

University of Tsukuba, aiyama@esys.tsukuba.ac.jp

This paper introduces a management system for robot groups with "holonic architecture" which is one of distributed and hierarchical system architectures. In order to manage collaboration of many robots, we put sub-management system on each work sub-space. A management system on a sub-space divides a task given by upper space into some sub-tasks and gives lower sub-spaces those sub-tasks. With this architecture we obtain distributed and hierarchical management system for the whole work space. We made a demonstration system and showed an example of task division and execution.

1. INTRODUCTION

There are several studies of multi-agent system. In the field of robotics, one of the most efficient targets of multi-agent system is production plant. In the production plant, there are several assembly robots, AGVs, machining tools, storages etc. They are considered as agents that have various characteristics.

Current production plant systems adopt central control method. It is because that the most efficient method is the lowest cost method and it can neglect the cost to make system motion plan. It depends that once we make a production line, it is not changed long time and also it products very large number of product.

But today, consumers' needs have become diversified. The life cycle of products has been shorten in the area of manufacturing. So, in these days, several researchers have started various studies with multi-agent system concept for highly flexibility against change of production plan. For example, Iwata et al. introduced RMS (Random Manufacturing System) (Iwata, 1994) and Ramos introduced similar structured distributed production system (Ramos, 1996). In Denso Corp., Hanai et al. realized practical autonomous mobile robot system APS (Adaptive Production System) (Hanai, 2001).

As an international joint research program, there is IMS (Intelligent Manufacturing System) program. IMS has several projects. One of them is HMS (Holonc Manufacturing System) project that was started at 1996 and it aims to research flexible and agile production system. Concept of holon is proposed by Koestler, a philosopher in 1967 (Koestler, 1967). Holon is an agent that has autonomy and cooperability. It is a component of hierarchical system; that means that all system components is a part of a stratum and acts as the whole for lower

stratum component and as a part for upper component. For example, a human body consists of various organs, an organ consists of several cells, a cell consists of various small elements, and so on. Koestler names these elements as “holon” and this hierarchy as “holarchy.”

In HMS project, we introduce the concept of holon and holarchy into production system to give flexibility and agility against demand. Production plant can be regarded as one of holonic system. Factory holon that manages the whole of the factory consists of some division holons that manage manufacturing, material management, design et al. They cooperate to achieve the function of the factory. And each division holon also consists some sub-divisions with cooperation for achievement of the function of the division.

There are several systems of HMS. Valckenaers compared holonic architecture, holarchy, to hierarchy and heterarchy and the three are assessed according to six criteria (Valckenaers, 1997). Ramos studied dynamic scheduling on the holonic system (Ramos, 1996). Arai et al. constructed holonic robot system that has similar architecture with Ramos's system (Arai, 1997)(Arai, 1999).

These researches are mainly concerned with a single production cell. In this paper, we pay attention to hierarchy of the production plant elements; factory, shop, cell and robot. We will introduce management architecture that has recursive structure for each hierarchical layer.

2. HOLONIC MANAGEMENT SYSTEM FOR HIERARCHICAL ROBOT GROUPS

2.1 Management of Robot Groups by Hierarchization

In case we intend to manage a number of cooperation tasks by two or more robots, two major methods can be considered. One is that a single management system manages intensively. The other is that each robot manages in autonomous distributed control. However, by these methods, there are some problems such as fault tolerance and increase of the amount of communication. So in these methods, to manage of great many robots is difficult.

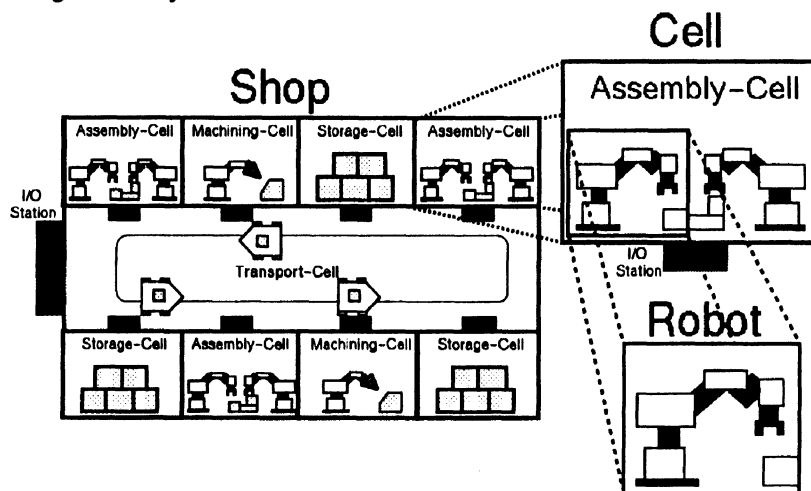


Figure 1 – Production plant

One example of management system that adjusts to great many robots is a production plant. Figure 1 shows a conceptual figure of working area called “Shop” in a production plant. Shop consists of one or more cells, and each cell consists of one or more robots. Inside of a cell, robots execute cooperation tasks such as machining, assembling, and storing. Inside a shop, cells cooperate mutually and execute more complicated tasks. A production plant handles complicated tasks by dividing a task hierarchically in this structure. Therefore it mitigates the complexity of a task in each working area. This mechanism is expected to be effective in managing cooperation tasks that are executed by great many robots. This architecture is suited for the holonic concept. So we adopt holonic system to manage this hierarchical system.

2.2 Unit and Sub-unit

When we intend to manage great many robots with distributed and hierarchical architecture, it can manage them more efficiently if it has the same mechanism on each layer. Therefore, we defined a basic unit that manages tasks in a layer and subunits that compose the basic unit.

Unit is a set of subunits and manages tasks by making subunits cooperate mutually

Subunit is an element that composes the upper unit. A subunit is a unit itself and has subunits in its inside.

2.3 Tasks Executed in a Unit

Tasks executed in a unit are cooperation tasks of subunits. In this research, we defined two kinds of tasks. They are abstracted tasks of production plants. They are classified into the following two tasks.

Proper task is proper action that a unit gives to target objects

Transport task is delivering a target object to adjoining unit

A task can be divided into subtasks called operations, which are distributed to subunits.

3. UNIT MANAGEMENT HOLON

3.1 Architecture of a Unit Management Holon

We embraced the concept of the holonic system on unit management system in order to manage hierarchical architecture. Elements shown in the Figure 2 with ellipse are holons. They manage a unit by mutual cooperation.

A unit management system divides a task into one or more operations and distributes them to subunits. Then they execute and manage the tasks. If the task is a proper task, the system distributes operations by contract net protocol with blackboard model. By this model, a unit management system does not have to have status of subunits. If the task is a transport task, the system contracts with subunits directly without the blackboard model.

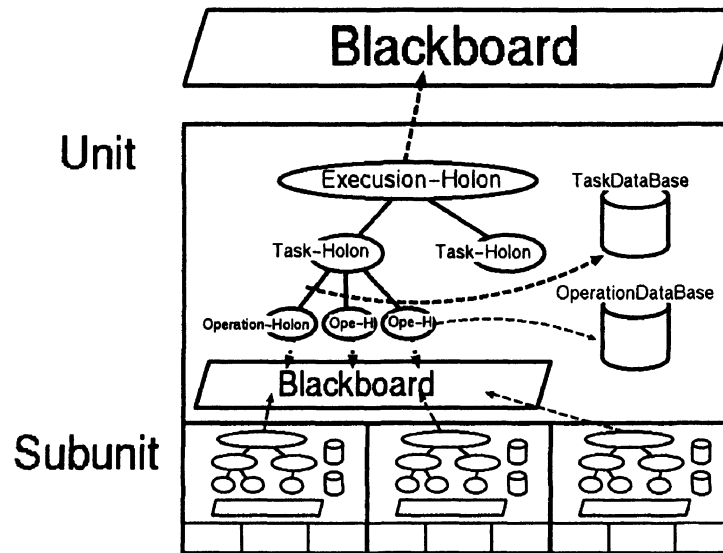


Figure 2 – Unit Management Hierarchy

And the system makes a route for transportation based on contiguity relation of subunits. The relation is described in a database. So all the subunits laying on the route join into a group for transportation.

A unit management system has following 4 layers inside. They have hierarchy for management.

Unit Execution Layer (Execution Holon)

In this layer, there is a single execution holon. The execution holon negotiates with operation holons that exist in its upper layer. And it manages all the tasks that are executed in the unit.

Task Layer (Task Holon)

In this layer there are some task holons. Each task holon manages a single task. A task holon divides a task into operations and manages them.

Operation Layer (Operation Holon)

In this layer there are some operation holons. Each operation holon manages a single operation. An operation holon negotiates with execution holons of its subunits. Then it selects one or more suitable execution holons and distributes the holons the operation.

Subunit Execution Layer (Execution Holon)

In this layer there are execution holons of subunits. Each subunit has a single execution holon. And each execution holon negotiates with the operation holons of the unit.

In a unit management system, there are some other elements. They are blackboard, task database and operation database. A blackboard is place on which operation holons put public offerings to subunits. And a task database is a database in which data of proper tasks are stored. Operation database is a database in which the data of contiguity relation of subunits are stored.

A unit management system is a part of the whole management system. And the whole system has hierarchical connection of unit management systems.

3.2 Algorithm of Task Management

In a unit management system, holons manage the unit by communicating mutually. The communication of holons travel from upper layer to lower layer hierarchically.

Following explains a process of task management in a unit.

1. An operation holon in an upper layer puts a public offering on the blackboard in the upper layer.
2. The execution holon checks the blackboard. When it finds the public offering, it uses the task database and searches data of the task.
3. If the execution holon succeeds to find, it generates a task holon for estimation of execution time.
4. The task holon starts the estimation. It divides the task into operations. Then it generates operation holons as many as the operations.
5. Each operation holon begins to search suitable subunits by putting public offerings on the blackboard to reserve them.
6. The task holon is observing the operation holons. If all the operation holons succeed to reserve, then it waits request from the execution holon. It is the completion of the estimation.
7. The execution holon is observing the task holon. If the task holon succeeds to estimate, it bids the public offering with the estimation and negotiates with the operation holon in the upper layer. Then it is reserved by the upper operation holon and begins to wait for the request from it.
8. If the execution holon receives request for executing the task from the operation holon in the upper layer, it sends request for executing the task to the task holon.
9. If the task holon receives request from the execution holon, it sends request for executing the operations to each operation holon according to the work procedure.
10. Each operation holon sends the request for executing to subunit execution holons.

Task holons and operation holons show the progress of the tasks and the operations as their states. For example, a task holon takes state of “estimate” when it is in estimation. And an operation holon also takes the state of “estimate” when it puts public offering and waits for bids. And if it succeeds to reserve suitable subunits, it takes state of “ready”.

Holons have some other states such as “execute”, “complete”, and so on. “Execute” states that a holon is executing a task or an operation. “Complete” states that a holon has already completed a task or an operation.

4. SIMULATION

4.1 System Construction Using Java

To construct a simulation system, we used Java language that is an object oriented-language. By using an object-oriented language, we can efficiently develop elements of system as independent modules. If we intend to update a module, we

don't have to mention all the system and just to update the single module. Being compared with other languages such as C, Java provides advanced functions for networking. And we can use easily functions such as two way communication by using a platform provided by Java.

4.2 Composition of the Simulation System

The composition of the simulation system is shown in Figure 3. A shop consists of an assembly cell, a transport cell, and 2 I/Os (I/O is a transport buffer). The assembly cell consists of 3 manipulators (Rob1, Rob2, Rob3), 2 storages (St1, St2) and 2 I/Os (I/O1, I/O2). The transport cell consists of an AGV. Each manipulator has a proper task that the manipulators inserts Part B into Part A. The shop and the assembly cell have a proper task of assembling P. In the shop, task assembling P divides into a single proper operation of assembling P that is a proper task of the assembly cell. In the assembly cell the task assembling P is divided into 4 operations, transporting A, transporting B, inserting B into A and transporting P.

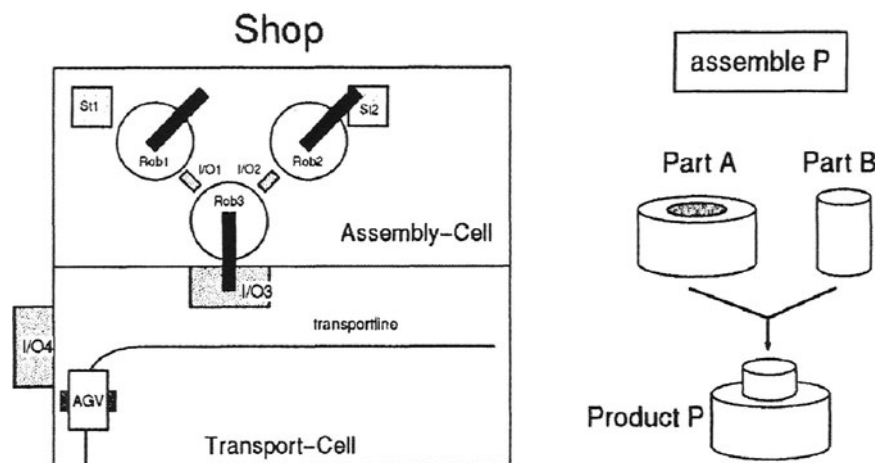


Figure 3 – Composition of Simulation System

4.3 Simulation and Result

In the simulation system, a user is the top level of the hierarchy. The user orders the shop to execute tasks by putting public offerings on a blackboard for the shop. We carried out the simulation on following condition. Time limit of public offerings to a shop is 10[sec], time limit of public offerings to cells is 5[sec], time limit of public offerings to robots is 1[sec], time required by robots for inserting is 10[sec] and time required by AGV for transporting is 20[sec]. Parts required for assembling P exist in the assembly cell beforehand.

The user puts the public offering for assembling P at 0[sec]. Then, 1 second later for completing assembling P, the user puts the public offering for transport P. We carried out this simulation for several times.

Figure 4 shows two results of the simulation. The states of task holons and operation holons in each unit and the actions executed by each unit along time are shown in the figure. The two cases are chosen from various results of simulation.

Comparing result (1) with result (2), it becomes clear that although we carry out the same simulation several times, different results have come out. The reason is that 3 manipulators have the same performance and we do not optimize in negotiation of operation holons and execution holons. So in the assembly cell, the robot group that executed the task had different compositions each time.

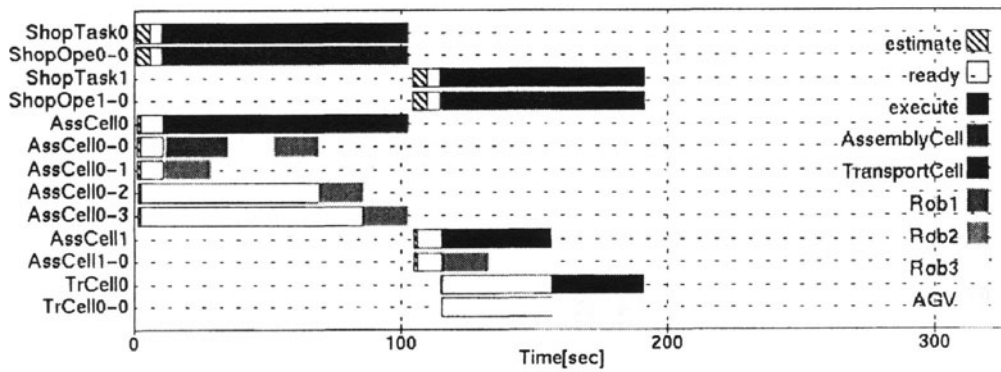
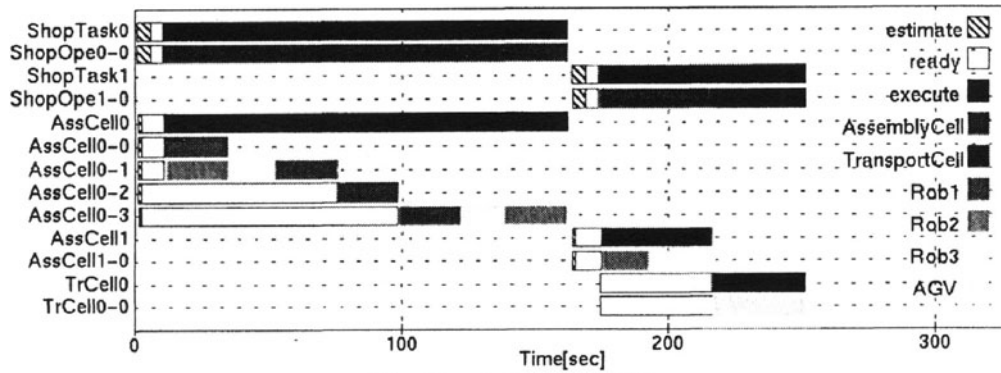


Figure 4 – Two Results of Simulation

4.4 Demonstration System

We have made a demonstration system for visualization of simulation result by using LEGO Mindstorms. LEGO Mindstorms are toy blocks and we can create various mechanisms with them. As they provide motors and sensors, we control them by using the programmable control device called RCX. We have created robots such as manipulators, conveyer and so on. By using LEGO Mindstorms, it needs much less cost for construction of a demonstration system comparing with using actual industrial robots.

5. CONCLUSION

In this research, we noticed the distributed and hierarchical architecture of production plants and suggested a technique for cooperation tasks by great many

robots. Then we constructed a simulation system and visualized it on a demonstration system.

When we carried out simulation, the usage of subunits brought differences of working term. So what remains to be done are following.

- Introduction of a scheduler that manages middle or long term plans.
- Introduction of negotiation based on difference of performance of robots.

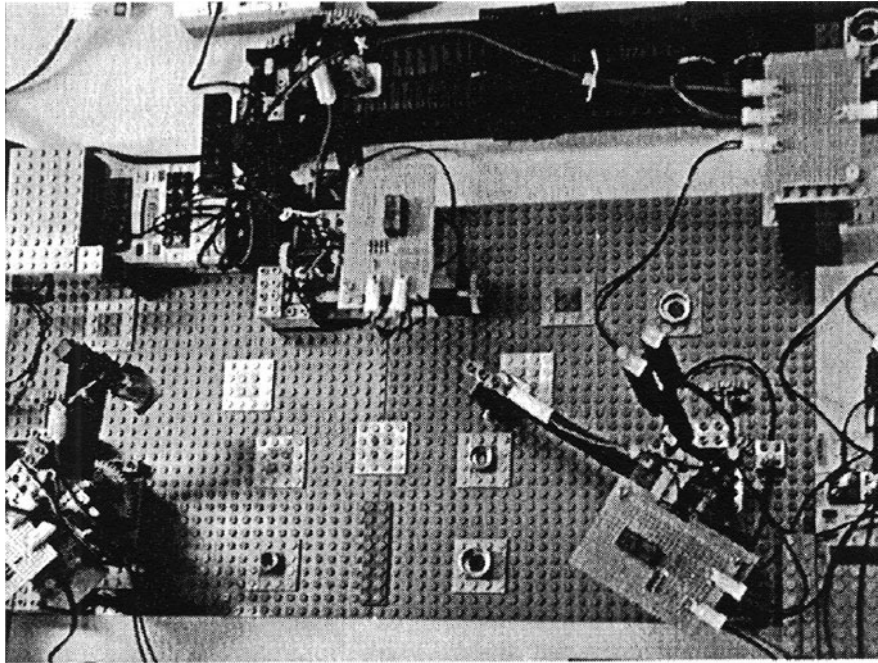


Figure 5 – Overview of the Demonstration System

6. ACKNOWLEDGEMENT

This research is supported by IMS (Intelligent Manufacturing System) / HMS (Holonc Manufacturing System) project.

7. REFERENCES

1. Arai, T et al. "Holonc Storage: An Assembly and Storage Cell by Manipulation using Environment". Proc. 29th CIRP Int. Seminar on Manufacturing systems, 1997: 221-226.
2. Arai, T et al. "Holonc Assembly System with Plug & Produce". Proc. Second Int. Workshop on Intelligent Manufacturing Systems, 1999: 119-126.
3. Hanai, M et al. "Development of Adaptive Production System to Market Uncertainty – Autonomous Mobile Robot System --". Proc. Int. Symp. On Assembly and Task planning, 2001: 61-66.
4. Iwata, K et al. "Random Manufacturing System: A New Concept of Manufacturing System for Production to Order". Annals of the CIRP, Vol.43, No.1, 1994: 379-383.
5. Koestler, A. "A Ghost in the Machine". Arkana, 1967.
6. Ramos, C et al. "A Holonc Approach for Task Scheduling in Manufacturing Systems". Proc. Int. Conf. on Robotics and Automation, 1996: 2511-2516.
7. Valckenaers, P et al. "Holonc Manufacturing Systems". Integrated Computer Aided Engineering, No.4, 1997: 191-201.