13

TECHNICAL PAPER

MS94-236

Strategy for Cooperative Motion of Multiple Robots in a Decentralized Robotic System

authors

HAJIME ASAMA
Research Scientist
Saitama University
Saitama, Japan

KOICHI OZAKI
Saitama University
Saitama, Japan

YOSHIKI ISHIDA
Lecturer
Kyushu University
Fukuoka, Japan

KAZUTAKA YOKOTA
Assistant
Utsunomiya University
Japan

AKIHIRO MATSUMOTO
Associate Professor
Toyo University
Saitama, Japan

HAYATO KAETSU
Senior Technical Scientist
The Institute of Physical and Chemical Research (RIKEN)
Saitama, Japan

ISAO ENDO
Chief Research Scientist
The Institute of Physical and Chemical Research (RIKEN)
Saitama, Japan

abstracts

Cooperative motion of multiple robots in a decentralized robotic system is classified into individual action and collaborating action. For conflict measure in individual motion, two types of resource management for conflict avoidance, and a multi-level conflict resolution for conflict resolution are proposed. For starvation measure in collaborating action, sympathetic consideration algorithm for starvation avoidance is proposed. These strategies are applied to task assignment, path planning with environment management, collision avoidance with dead lock solution, and collaborative team organization. Finally, by integrating the proposed methods, task processing scenario was designed for cooperative motion of multiple robots in a decentralized robotic system.

conference

FIFTH WORLD CONFERENCE ON ROBOTICS RESEARCH
September 27-29, 1994
Cambridge, Massachusetts

terms

Robotics
Systems Engineering
Robots
Computers
Planning
Automation

Society of Manufacturing Engineers
1994

Sponsored by Robotics International
of the Society of Manufacturing Engineers
One SME Drive • P.O. Box 930 • Dearborn, MI 48121
Phone (313) 271-1500

1 Introduction

Though flexible and robust robotic systems have been required in various fields, sufficient functionality has not yet been realized with currently available technologies in spite of all the efforts to develop intelligent robots. Instead of single-sophisticated-robot-oriented researches, multi-agent robotic systems (MARS) have recently attracted the attention of many researchers as a new approach for flexible and robust systems. In recent symposiums and conferences on robotics, large number of papers related to MARS, such as those on multi-robots, distributed robotic systems, cellular robots, self-organizing robots, and collective intelligence (swarm intelligence), were presented, and discussion on distributed autonomous robotic systems[1][2], and needs for researches of cooperating robots[3] have been actively done.

We have developed a decentralized autonomous robotic system, ACTRESS (ACTor-based Robots and Equipments Synthetic System) for high-level tasks such as maintenance in nuclear power plants or flexible automation in manufacturing plants[4]. ACTRESS has been designed as an intelligent robot system which is composed of heterogeneous multiple robotic agents called *robotors* including various types of autonomous robots, equipments, and computing systems. Communication between arbitrary agents is available for cooperative motion. The most characteristic feature of ACTRESS is fusion of parallel, individual actions and cooperative actions by *robotors*. With this feature, not only flexibility for dynamic requirements and unexpected situations, but also efficiency by parallel processing, and fault tolerance in case of faulty conditions in any *robotors* are realized. In this paper, strategy for cooperative motion of multiple robots discussed in ACTRESS research are mentioned.

2 A Decentralized Robotic System, ACTRESS

2.1 Basic Concept of ACTRESS

In order to realize a multi-functional, flexible, and robust robotic system, we adopted a concept of *functional distribution in design, and cooperation in operation*, where the required tasks are processed by cooperation of multiple robotic agents with rather simple function instead of a single sophisticated robot. The concept of the ACTRESS is illustrated in Fig.1, which is composed of multiple robotic agents called *robotors*. In this concept, communication between agents is assumed, taking account of difficulty of message translation only by sensory information. Thus, each *robotors* should be autonomous and have ability to communicate with other *robotors* by exchanging messages. Communication framework consists of communication protocol to establish physical communication link and message protocol to exchange messages corresponding to necessary information. Message protocol core as common communication means and five levels of message exchange are provided for message protocol. Five levels are listed as control level, physical level, procedural level, knowledge level, and concept level. In addition, the main functions of message protocol core are negotiation, inquiry, offer, announce, and synchronize. Diverse problem such as task assignment or resource sharing are solved by negotiation.

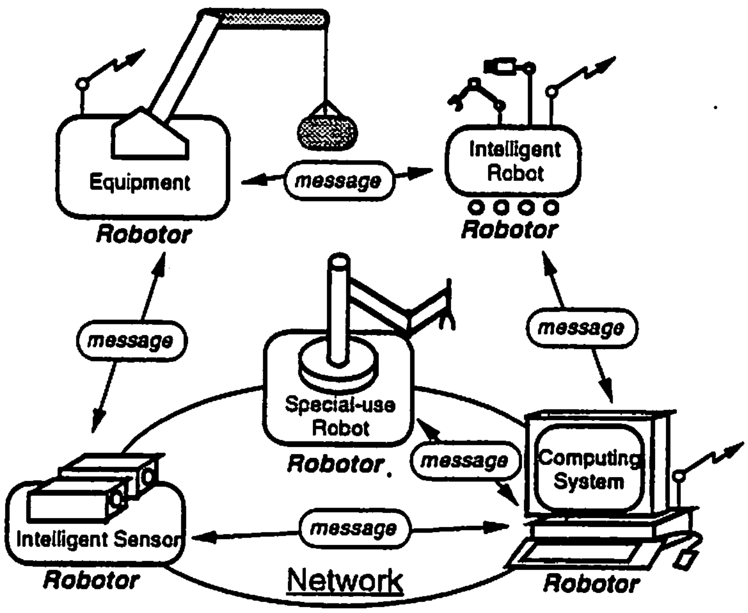

Figure 1. Concept of ACTRESS

2.2 Prototype System

Based on evaluation of functional distribution[5], we have developed a prototype system of ACTRESS. Figure 2 shows the configuration of the environment of the prototype system. This system is composed of three autonomous mobile robots, and several stationary agents (workstations) with different function. Figure 3 is outlook of the mobile robots. Image Processor (IM)[6], Human Interface (HI/F)[7] and Global Environment Manager (GEM) are provided as

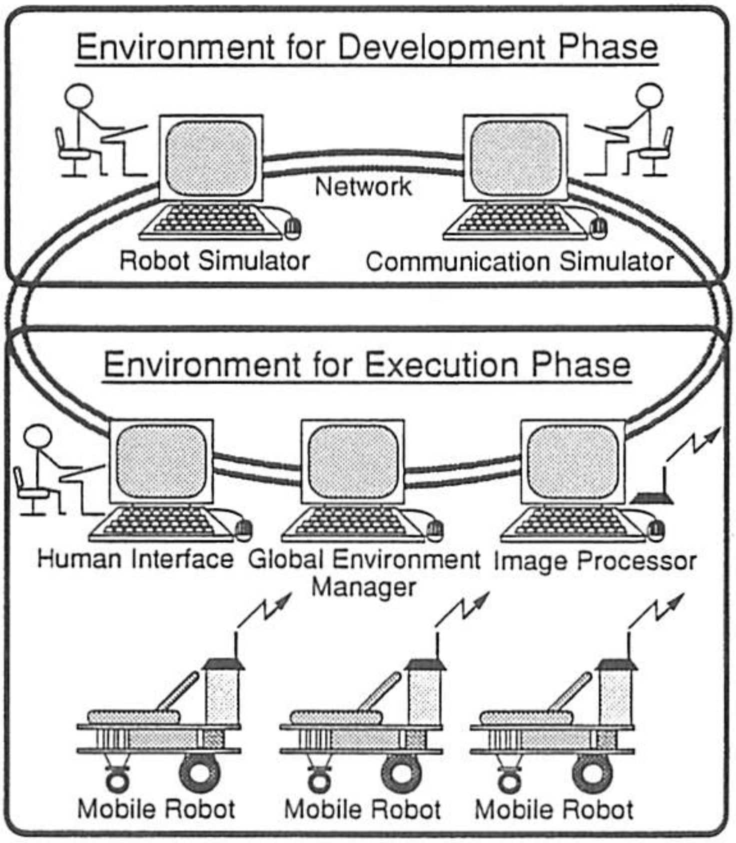

Figure 2. Configuration of prototype system environment

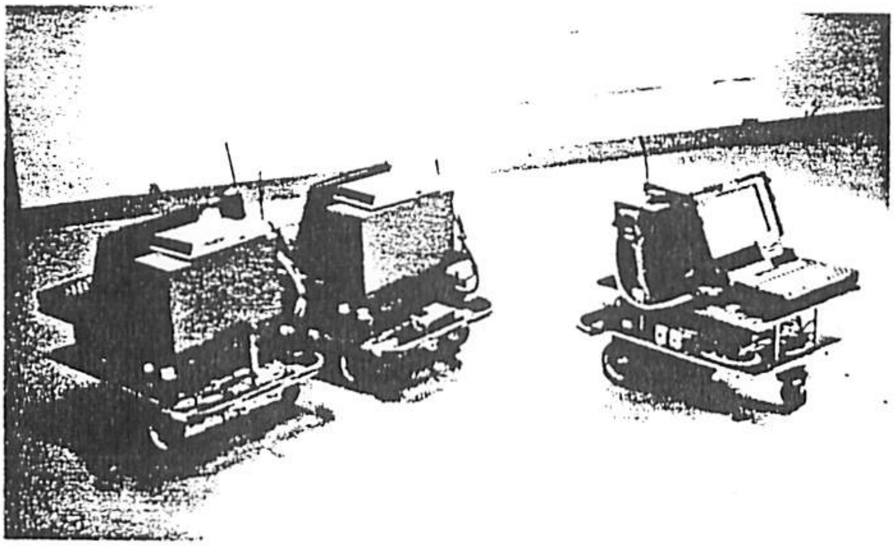

Figure 3. Outlook of three autonomous mobile robots

stationary agents for execution phase. As the mobile robots are not sophisticated enough to process the images taken by CCD cameras, The IM is devoted for processing images, which are requested to process from mobile robots. The HI/F sends messages or instructions from a human operator to other agents, and disply the messages from other agents to the operator. The GEM manages environmental information, updates it according to reports from robots, and supplies it on demand. Robot Simulator (RS)[8] and Communication Simulator (CM)[9] are provided as stationary agents for development phase. The RS analyzes the motion of robots, which is programmed by a developer, and display it to the developer. The CM is utilized for analysis of the messages exchanged by robots, and varification of the negotiation procedures. Based on the design of communication framework[10], we have also developed a communication system by contention access, which enables message passing from any agent to any others[11]. The IM plays a role of a gateway forwarding messages between two communication media, namely radio communication for mobile agents and LAN communication for stationary agents.

3 Strategy for Cooperative Motion

3.1 Consideration of Cooperative Motion

Concerning cooperative motion of multiple mobile robots, some methods for consistent macroscopic behaviors by mobile robots such as arrangement, team organization, have been proposed so far[12][13][14], which were verified only by simulation from the analytical viewpoint. On the other hand, some coordinated motion of actual mobile robots, such motions as error recovery[15], tandem motion[16], shunting motion[17], and traffic rule-based motion[18] have been reported. However, these motions were realized in ad-hoc way focusing on specific aspects. Discussion on cooperation of multiple autonomous robots which has done so far was concerned with only robot control without deadlocks[19]. In order to develop a decentralized robotic system including multiple autonomous mobile robots, and to make the robotic system to achieve various missions according to situations, it is necessary to discuss synthetically how each robot should deal cooperatively with various problems in any aspects, such as task assignment, mutual collision avoidance, deadlock resolution, resource sharing, coordinated control, etc., which arise in the environment with multiple robots.

The action of cooperative multiple robots is classified into two types:

- **Individual action**
  This is a type of cooperative action where each robot processes different tasks individually and parallelly. In this case, *conflict measure* (i. e. function to cope with conflicts in resources which should be shared by multiple robots) is required for deadlock avoidance or resolution. In other words, this is cooperation as not to disturb other robots' action.

- **Collaborating action**
  This is another type of cooperative action where some robots process a common task in a collaborative manner. In this case, *starvation measure* (i. e. function to cope with situation keeping on waiting for help forever) is required for organization of cooperative teams and collaboration. In other words, this is cooperation as to support other robots' action.

In order to operate cooperative multiple robots, these two functions should be provided for each robot so that both types of action can be performed.

3.2 Strategy for Conflict Avoidance

For resource sharing with minimum conflicts in a decentralized robotic system. the resources are desired to be managed. We have designed the following types of resource management strategies:

- **Centralized Management and Distribution**
  Resource information is managed by a centralized manager. The manager distribute the resources to resource users so as to avoid conflicts.
- **Acquisitive-Type Centralized Management**
  Resource information is managed by a centralized manager. Resource users utilize the resources after they reserve the resources. The manager should give permission to utilize the resource to the agent requesting reservation so as to avoid conflicts.
- **Distributed Management**
  Resource information is managed by resource users distributively. Each resource user declares that it utilizes the resources so as to avoid conflicts, based on the distributively managed resource information. All the other resource users should renew the resource information according to the broadcasted declaration.

By regarding various entities as resources (e.g. not only physical resources such as space occupied by robots, common use tools, etc., but also logical resources such as signals of active sensor, tasks, etc.), which cannot be shared by plural robots at the same time. the strategies managing resources can be applied to various problems. Depending on the properties of the resources, proper management strategy should be applied to each resources.

The *centralized management and distribution* doesn't fit the decentralized system, because it inhibits the advantages of the decentralized system. The *acquisitive-type centralized management* is disadvantageous in fault tolerance and flexibility, but efficient in terms of information processing, and suitable for the problems where the consistency is important. The *distributed management* matches the decentralized system, but may not be efficient in terms of communication, and cannot guarantee the consistency due to loss of broadcast messages.

If it is difficult to define or quantize the resources, it is appropriate to give up managing them. In this case, the resource users search for the resources, and utilize them. As conflicts can occur during operation, the mechanism to resolve them should be provided.

3.3 Strategy for Conflict Resolution

In multi-agent and decentralized system, conflicts are desired to be resolved as locally as possible. However, for complicated situations, global resolution mechanism is demanded. We have proposed a multi-level problem solving strategy. Several levels of conflict resolution mechanisms are provided as follows:

1. Local resolution algorithm without communication
   (only based on the observed information)
2. Local resolution using communication (by negotiation)
   (taking account of the precise information obtained by communication)

3. Global resolution by global resolver
(taking account of surrounding situations and a global goal)

4. Ultimate global resolution by a human operator
(with absolute interaction)

Each robot which meets conflicts tries to resolve it in the most local level. If the situation is so complicated that it fails to resolve it in the local level, it changes the level to resolve them again using communication. When the agent fail to resolve conflicts in local manners, it sends a request message to a global solver (another agent) to resolve it. Even any global solvers cannot resolve them, they should be treated by a human operator (ultimate conflict solver), whose instructions are given via HI/F.

3.4 Strategy for Starvation Avoidance

A collaborative team is necessary to be organized for collaborating action. We have defined that a collaborative team consists of one coordinator which organizes the team and cooperators which support the coordinator. The starvation situation occurs when multiple coordinators appear, which seek for cooperators mutually. As function to avoid starvation, *sympathetic consideration algorithm* was proposed[20]. In this algorithm, whenever a robot needs collaboration, it checks if any collaboration requesting messages from other robots are received before it sends a requesting message. If any message is received, it gives up its own task, and consents to the message to support others' tasks as a cooperator. If no message is received, it becomes a coordinator, and broadcasts a collaboration requesting message for recruitment. Also at every moment a robot finishes task execution, it checks if any collaboration requesting messages from other robots are received.

By the sympathetic consideration algorithm, the starvation can be avoided because the agent is always tend to be cooperators prior to being coordinator. Nevertheless, this algorithm is not always optimal for multi-robot operation from the viewpoint of efficiency of the total performance, because the robot which receives the request from other robots should always give up its own task and support them at any condition. A more efficient algorithm which allows coordinator conversion depending on situations (requirements of task, environmental condition, etc.), should be also discussed.

4 Synthesis of Cooperation

4.1 Applications of Strategies to Problems in Various Aspects

The strategies discussed above to operate multiple robots cooperatively in a decentralized robotic system have been applied to concrete problems in various aspects.

- **Task Assignment**
We have developed a method of cooperative task assignment based on decentralized management taking object pushing problem as a typical mission[20]. The *distributed management* is employed for managing the situation of task proceeding. Whenever a robot selects a task, and finishes the task, it broadcasts the message to declare the task selection to report

the task achievement. According to receiving the broadcasted messages, every robot can manage the task processing status by knowing how the work is going on (which robot is processing or has processed which tasks), As long as each robot chooses one of remained tasks and claims it based on the distributed management of tasks, conflicts on task assignment can be avoided eventually.

- **Path Planning with Environment Management**

  For path planning of mobile robots, the environmental information is indispensable. In multi-robot environment, however, the environment is dynamically changing according to the motion and task processing by robots. We have evaluated the communication amount. and obtained optimal functional distribution on path planning[5] and strategy for path planning[21]. Static path plans are made by each robot based on only local stationary environment (static environment including movable objects, but not including mobile robots). Concerning movable objects, the local stationary environment is managed by each robot. but the global environment is managed by a separate centralized agent (GEM). Namely, *acquisitive-type centralized management* is adopted for global environment management. According to demand, each robot updates the local environmental information by inquiry of current global information to the GEM. The robot which processed a task, and changed the stationary environment should send the report message to GEM. Based on the reports, GEM can manage the current global environmental information.

- **Collision Avoidance with Deadlock Resolution**

  Even with static path plans, collision between mobile robots should be avoided dynamically. Collision avoidance is regarded as resolution of conflicts of floor resource occupancy by mobile robots. However, as it is difficult to make the floor workspace discrete, segmented. and managed dynamically, we have decided not to manage it.

  As various situations of collisions including deadlocks are presumed, multi-level conflict resolution strategy has been employed for dynamic path planning with avoiding collisions and deadlocks according to the complexity of the situations, which is composed of the following levels[21]:

  1. Collision avoidance based on local algorithm.
  2. Collision Avoidance based on mobile rules.
  3. Giving way based on priority of tasks.
  4. Deadlock resolution by a leader.
  5. Dynamic path planning by a deadlock solver.
  6. Problem solving by a human operator.

Collision avoidance based on such a local algorithm as the potential method is applicable only in limited cases. In simple situations, it is effective to follow predefined traffic rules. or to give way based on priority of tasks. Nevertheless, in situations where a deadlock occurs, it is demanded to determine a leader among the robots, who solve the deadlock and plan the actions of all the robots involved in the deadlock. However, in more complicated situations where any leaders cannot solve the deadlocks, specialized deadlock solvers should be provided for the robot system according to complexity of the situations. The ultimate deadlock solver may be a human operator, namely human interface system which can transmit instructions from the human operator to mobile robots.

Generally speaking, mainly sensory information is utilized for local conflict resolution with saving communication cost. For global conflict resolution, however, communication becomes indispensable to exchange global information.

- **Collaborative Team Organization**
Taking an example of object pushing task, the sympathetic consideration algorithm was verified[20]. In this task, light objects and heavy objects are scattered in a room, and multiple mobile robots are given a mission to clean up the room by pushing all the objects to the walls. Each robot can process light objects only by individual motion, but it needs collaboration to process heavy objects. A robot which encounters a heavy object during individual motion, tries to become a coordinator and to organize a collaborative team by negotiation. By implementing the sympathetic consideration algorithm, successful organization of collaborative teams was achieved without starvation. In the algorithm, any agent which receives collaboration requesting messages memorizes the message, and consents to it whenever the robot finishes the current task or it finds that it also needs other's cooperation.

Moreover, a new type of communication called *groupcast* (broadcast within group members which belong to a group) and a learning mechanism utilizing knowledge on other agents' performance, which is accumulated as a result of past negotiation, have been implemented. With these functions, effective organization was achieved[22].

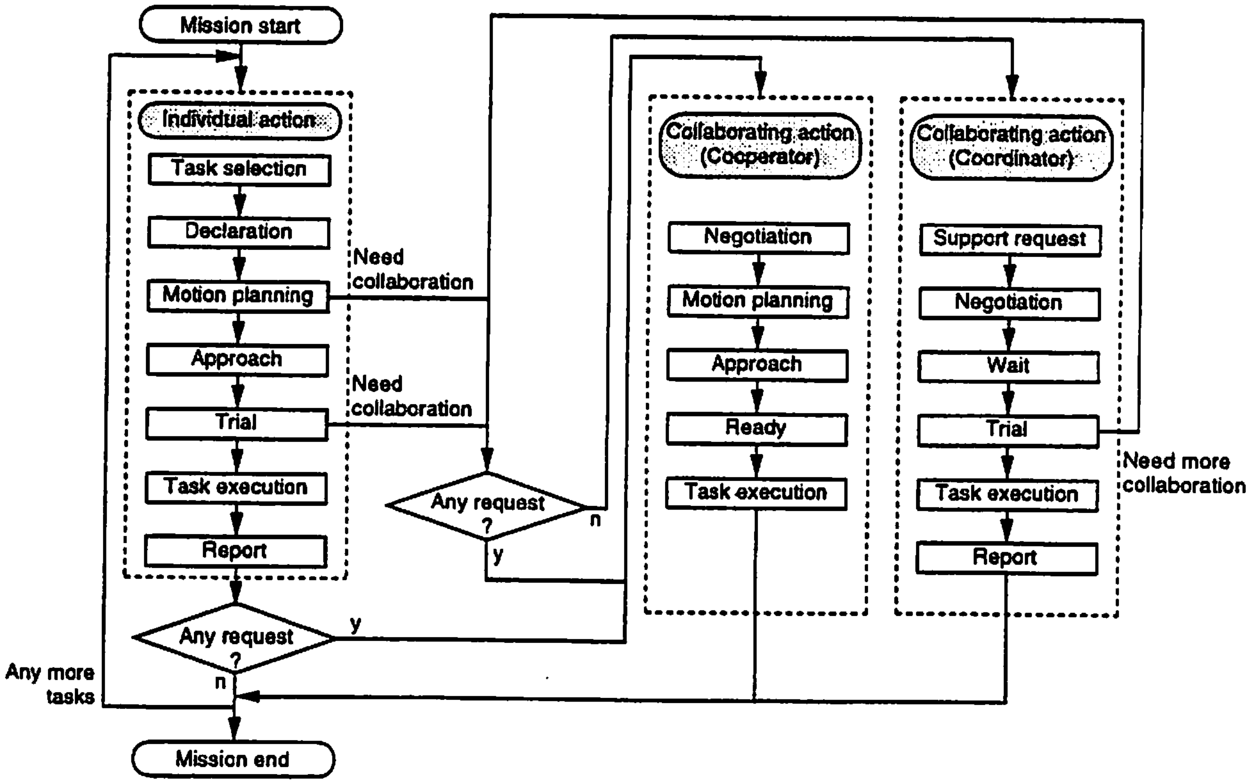

Figure 4. Abstract flow of task processing

4.2 Design of Task Processing Scenario

By integrating elementary application methods developed so far, a task processing scenario has been designed for operating cooperative multiple autonomous robots, taking an example of object pushing task. The abstract flow of task processing is shown in Fig. 4.

The mission composed of several tasks is given by human operator via the HI/F agent. At first, each robot tends to process the task by individual action. In the individual action, the robot selects a task based on distributed task management, declares it by broadcasting, plans its motion, approaches to the target object, tries to push and execute the task. If the task processing by individual action is successfully done, it reports about it by broadcasting, and checks if it receives collaboration requesting messages from other robots. If it receives any requests, it becomes a cooperator for collaborating action; otherwise, it repeats the individual action until all tasks are processed.

When the robot finds that collaboration is required during motion planning, or when it fails to execute the task by individual motion due to lack of function, it gives up processing the task by individual action. As the sympathetic consideration algorithm is incorporated, it checks if it receives collaboration requesting messages from other robots. If it receives any requests, it gives up its own task, and becomes a cooperator for collaborating action. Only if it doesn't receive any requests, it becomes a coordinator for collaborating action. A cooperator in collaborating action negotiates with the coordinator and follows the instructions from the coordinator with motion planning, approach, and task execution. The coordinator in collaborating action broadcasts collaboration requesting message, and waits for replies. Then, it negotiates with agents to determine cooperators, gives instructions to cooperators, executes the task synchronously with cooperators, and reports about it by broadcasting.

For motion planning, each robot inquires environmental information from the GEM. The GEM manages the environmental information according to broadcasted reports, and supplies the information according to demand. Mutual collision avoidance with deadlock solution works during task execution. All the strategies and their applications mentioned above are involved in this algorithm, where conflict measure and starvation measure are incorporated to achieve both of individual action and collaborating action.

5 Conclusion

Strategy for cooperative motion of multiple robots in a decentralized autonomous robotic system was discussed in this paper. The strategies proposed in this paper are summarized in Fig. 5. Cooperative motion is classified into individual action and collaborating action. For individual motion, conflict measure is required, and we proposed two type of resource management for conflict avoidance, and a multi-level conflict resolution for conflict resolution. For collaborating action, starvation measure is required, and we proposed sympathetic consideration algorithm for starvation avoidance. These strategies were applied to task assignment, path planning with environment management, collision avoidance with dead lock solution, and collaborative team organization. Finally, by integrating the proposed methods, task processing scenario was designed for cooperative motion of multiple robots in a decentralized robotic system. As examples of cooperative motion, mutual collision avoidance motion[21], synchronized side-by-side motion[23], and functional complement motion[6] have already realized by actual mobile robots and computing systems. The total operation of actual robots to accomplish a mission is the future plan.

Cooperative motion
- Individual action
  - Conflict avoidance
    - ←*Acquisitive-type centralized management*
    - ←*Distributed management*
  - Conflict resolution
    - ←*Multi-level conflict resolution*
- Collaborating action
  - Starvation avoidance
    - ←*Sympathetic consideration algorithm*

Figure 5. Summary of strategies for cooperative motion

References

[1] *Int. Symp. on Distributed Autonomous Robotic Systems*, (1992).

[2] *2nd Int. Symp. on Distributed Autonomous Robotic Systems*, (1994).

[3] R. Mann and S. Yuta organized, "Needs for Research in Cooperating Robots," in *Textbook of Workshop, IEEE Int. Conf. on Robotic and Automation*, (1993).

[4] H. Asama, A. Matsumoto and Y. Ishida, "Design of an Autonomous and Distributed Robot System: ACTRESS," in *Proc. IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, (1989)283.

[5] H. Asama, M. K. Habib, I. Endo, K. Ozaki, A. Matsumoto and Y. Ishida "Functional Distribution among Multiple Mobile Robots in An Autonomous and Decentralized Robot System," in *Proc. IEEE Int. Conf. on Robotics and Automation*, (1991)1921.

[6] Y. Ishida, H. Asama, S. Tomita, K. Ozaki, A. Matsumoto and I. Endo, "Functional Complement by Cooperation of Multiple Autonomous Robots," in *Proc. IEEE Int. Conf. on Robotics and Automation*, (1994)2476.

[7] K. Yokota, T. Suzuki, H. Asama, A. Matsumoto and I. Endo, "A Human Interface System for the Multi-Agent Robotic System," in *Proc. IEEE Int. Conf. on Robotics and Automation*, (1994)1039.

[8] K. K. Habib, H. Asama, Y. Ishida, A. Matsumoto and I. Endo, "Simulation Environment for An Autonomous and Decentralized Multi-Agent Robotic System," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1992)1550 .

[9] Y. Ishida, H. Asama, I. Endo, K. Ozaki and A. Matsumoto, "Communication and Cooperation in the Autonomous and Decentralized Robot System," in *Proc. IFAC Int. Symp. on Distributed Intelligence Systems*, (1991)299.

[10] A. Matsumoto, H. Asama, Y. Ishida, K. Ozaki and I. Endo, "Communication in the Autonomous and Decentralized Robot System ACTRESS," in *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, (1990)835.

[11] H. Asama, K. Ozaki, A. Matsumoto, Y. Ishida, M. K. Habib, H. Kaetsu and I. Endo. "A Communication System between Multiple Robotic Agents," in *Proc. Japan/USA Symp. on Flexible Automation*, (1992)647.

[12] K. Sugihara and I. Suzuki, "Distributed Motion Coordination of Multiple Mobile Robots," in *Proc. IEEE Int. Symp. on Intelligent Control*, (1990)138.

[13] L. E. Parker, "A Performance-based Architecture for Heterogeneous, Situated Agent Cooperation," in *Proc. AAAI '92*, (1992).

[14] L. E. Parker, "Designing Control Laws for Cooperative Agent Teams," in *Proc. IEEE Int. Conf. on Robotic and Automation*, (1993)3-582.

[15] F. R. Noreils, "Integrating MultiRobot Coordination in a Mobile-Robot Control System," in *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, (1990)43.

[16] F. R. Noreils, "Multi-Robot Coordination for Battlefield Strategies," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1992)1777 .

[17] S. Yuta and S. Premvuti, "Coordinating Autonomous and Centralized Decision Making to Achieve Cooperative Behaviors between Multiple Mobile Robots," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1992)1566 .

[18] S. Kato and J. Takeno, "Fundamental Studies on the Application of Traffic Rules to the Mobile Robot World," in *Proc. Int. Conf. on Advanced Robotics*, (1991)1063.

[19] S. Premvuti and S. Yuta "Consideration on the Cooperation of Multiple Autonomous Mobile Robots," in *Proc. IEEE Int. Workshop on Intelligent Robots and Systems*, (1990)59.

[20] H. Asama, K. Ozaki, A. Matsumoto, Y. Ishida and I. Endo, "Development of Task Assignment System Using Communication for Multiple Autonomous Robots," *J. Robotics and Mechatronics*, 4, 2(1992)122.

[21] H. Asama, K. Ozaki, A. Matsumoto, Y. Ishida and I. Endo, "Collision Avoidance among Multiple Mobile Robots Based on Rules and Communication," in *Proc. IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, (1991)1215.

[22] H. Asama, K. Ozaki, Y. Ishida, K. Yokota, A. Matsumoto, H. Kaetsu and I. Endo, "Collaborative Team Organization Using Communication in a Decentralized Robotic Systems," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1994)(to be presented).

[23] K. Ozaki, H. Asama, Y. Ishida, A. Matsumoto, K. Yokota, H. Kaetsu and I. Endo, "Synchronized Motion by Multiple Mobile Robots using Communication," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1993)1164.