AI Magazine Volume 18 Number 1 (1997) (© AAAI)

RoboCup

A Challenge Problem for AI

Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, Eiichi Osawa, and Hitoshi Matsubara

■ The Robot World-Cup Soccer (RoboCup) is an attempt to foster AI and intelligent robotics research by providing a standard problem where a wide range of technologies can be integrated and examined. The first RoboCup competition will be held at the Fifteenth International Joint Conference on Artificial Intelligence in Nagoya, Japan. A robot team must actually perform a soccer game, incorporating various technologies, including design principles of autonomous agents, multiagent collaboration, strategy acquisition, real-time reasoning, robotics, and sensor fusion. RoboCup is a task for a team of multiple fast-moving robots under a dynamic environment. Although RoboCup's final target is a world cup with real robots, RoboCup offers a software platform for research on the software aspects of RoboCup. This article describes technical challenges involved in RoboCup, rules, and the simulation environment.

The World Cup Robot Soccer (RoboCup) is an attempt to promote AI and robotics research by providing a common task for evaluation of various theories, algorithms, and agent architectures. For the robot (physical robot and software agent) to play a soccer game reasonably well, a wide range of technologies need to be integrated, and numbers of technical breakthroughs must be accomplished. The range of technologies spans both AI and robotics research, such as design principles of autonomous agents, multiagent collaboration, strategy acquisition, real-time reasoning and planning, intelligent robotics, and sensor fusion. The first RoboCup, RoboCup-97, will be held during the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97) in Nagoya, Japan, as part of IJCAI-97's special program. A series of competitions are planned afterward, just like the Formula One Championship. RoboCup consists of three competition tracks: (1) *real robot league:* physical robots playing soccer games, (2) *software robot league:* software agents playing soccer games on an official soccer server over the network, and (3) *expert robot competition:* competing robots that have special skills but are not able to play a game.

Although RoboCup's primary objective is a world cup with real robots, RoboCup also offers a software platform for research on the software aspects of RoboCup. The software robot league, also called the simulator league, enables a wider range of researchers to take part in this program. It also promotes research on network-based multiagent interactions, computer graphics, and physically realistic animations; a set of technologies potentially promotes the advanced use of the internet. In addition, we intend to create an award for an expert robot that demonstrates a high level of competence for a specific task, such as shooting and intercepting.

Although it is obvious that building a robot to play a soccer game is an immense challenge, readers might wonder why we propose RoboCup. It is our intention to use RoboCup as a vehicle to revitalize AI research by offering a publicly appealing but formidable challenge. One of the effective ways to promote engineering research, apart from specific application developments, is to set a significant long-term goal. When the accomplishment of such a goal has significant social impact, it is considered a grand-challenge project (Kitano et al. 1993). Building a robot to play a soccer game itself does not generate significant social and economic impact, but

Copyright © 1997, American Association for Artificial Intelligence. All rights reserved. 0738-4602-1997 / $2.00

	Chess	RoboCup
Environment	Static	Dynamic
State Change	Turn taking	Real time
Info. Accessibility	Complete	Incomplete
Sensor Readings	Symbolic	Nonsymbolic
Control	Central	Distributed

Table 1. Comparison between Chess and RoboCup.

the accomplishment will certainly be considered a major achievement in the field. We call this kind of project a *landmark project*. RoboCup is a landmark project as well as a standard problem.

The successful landmark project claims to accomplish an attractive and broadly appealing goal. The most successful example is the Apollo Space Program. In the case of Project Apollo, the United States committed itself to the goal of "landing a man on the moon and returning him safely to earth" (Kennedy 1961). The accomplishment of the goal itself marks the history of humankind. Although the direct economic impact of having someone land on the moon is slim,[1] the important issue for the landmark project is to set the goal high enough so that a series of technical breakthroughs are necessary to accomplish the task; this goal needs to be widely appealing and exciting. In addition, a set of technologies necessary to accomplish the goal must form the foundation of the next-generation industries.

In the case of RoboCup, the ultimate goal is to develop a robot soccer team that can beat the Brazil world-cup team. (A more modest goal is to develop a robot soccer team that plays like a human team.) Needless to say, the accomplishment of the ultimate goal will take decades of effort, if not centuries. With the current technologies, it is not feasible to accomplish this goal in the near term. However, this goal can easily create a series of well-directed subgoals. Such an approach is common with any ambitious, or overly ambitious, project. In the case of the American space program, the Mercury Project and the Gemini Project, which was the manned orbital mission, were two precursors to the Apollo mission. The first subgoal to be accomplished in RoboCup is to build real and software robot soccer teams that play reasonably well with modified rules. Even to accomplish this goal undoubtedly generates technologies that affect a broad range of industries.

Another view of RoboCup is as a standard problem so that various theories, algorithms, and architectures can be evaluated. Computer chess is a typical example of the standard problem. Various search algorithms were evaluated and developed using this domain. With the recent accomplishment by the Deep Blue team, which beat Kasparov, a human grand master, using the official rules, computer chess challenge is close to its finale. A major reason for the success of computer chess as a standard problem is that the evaluation of the progress was clearly defined. The progress of the research can be evaluated as a strength of the system. However, because computer chess is about to complete its original goal, we need a new challenge, one that initiates a set of next-generation technologies. We believe that RoboCup fulfills such a demand. Table 1 illustrates the difference between the domain characteristics of computer chess and those of RoboCup.

RoboCup is designed to handle real-world complexities, although in a limited world, while it maintains an affordable size and research cost. RoboCup offers an integrated research task covering the broad areas of AI and robotics, including real-time sensor fusion, reactive behavior, strategy acquisition, learning, real-time planning, multiagent systems, context recognition, vision, strategic decision making, motor control, and intelligent robot control.

Research Issues of RoboCup

In this section, we discuss several research issues involved in the development of real robots and software agents for RoboCup. One reason why RoboCup attracts so many researchers is that it requires the integration of a broad range of technologies into a team of complete agents; it is not a task-specific functional module. The following list gives some of the research areas involved in RoboCup: (1) agent architecture; (2) combined reactive and modeling-planning approaches; (3) real-time recognition, planning, and reasoning; (4) reasoning and action in a dynamic environment; (5) sensor fusion; (6) multiagent systems; (7) behavior learning for complex tasks; (8) strategy acquisition; and (9) cognitive modeling.

In addition, providing a network-based soccer server with high-quality three-dimensional graphics requires advanced technolo-

RoboCup Regulations

Real worldness in RoboCup arises mainly from the vast complexity of the overall situation as a result of interactions between behaviors and strategies of the ball and the players that cannot fully be predicted or controlled.

In the real robot session, we expect to have significantly greater complexity and, hence, much stronger reality than in the simulation session. This complexity results from the uncertainty and uncontrollability in the structures and functions of the real robots along with real physical phenomena. Therefore, we lean toward the least commitment policy in the game regulations so that they do not obstruct surprises and creativity.

Because of the technical difficulty and unpredictability, the regulations can be adjusted to the overall situation of the participating teams in each contest. However, the modifications must maintain fairness to all the participants and must be announced in advance of the contest, with the approval of the RoboCup technical committee.

The following sections summarize the current regulations. The rules have undergone two major changes since the first announcement in 1995. Also, prior to the announcement, several changes have been made since 1993 when we first drafted the RoboCup rules. The recent major changes include the size of the field, the size of the robot, and the creation of the defense zone. The field was based on a Ping-Pong table so that most people could purchase one at low cost anywhere in the world. It is important to consider the availability of the material supply. The field, balls, and other materials were chosen so that the greatest possible number of researchers could easily access and purchase them at a low cost.

Further modifications to the regulations will be made to reflect the research progress of the participants. The RoboCup real robot league basically has three different classes based on the size of the robots and the field: (1) small-size robot, (2) medium-size robot, and (3) large-size robot. Other classes, such as special robots supported by sponsoring corporations, might be created after discussion by the committee. A more detailed and more recent version of the regulation is available from the RoboCup home page, http://www.robocup.org/RoboCup.

Regulations for the Small Robot League

Field Size: A Ping-Pong table is used for the official match. The size and color of the table are determined by the international standard for Ping-Pong. It is 152.5 centimeters (cm) by 274 cm; the color is green. Details are given in figure 1.

Robot: The maximum diameter of a circular robot is 15 cm, and the maximum length of a rectangular robot is 18 cm, with a width of 10 cm. These dimensions provide for the same-size robot in terms of surface area, which is approximately one-tenth the length of the shorter end of the field.

Team: A team should consist of no more than five robots.

Goals: The width of the goal is 50 cm, which is approximately one-third the length of the shorter end of the field.

Ball: An orange golf ball is used.

Colorings: The colors of each part of the field are as follows: The field is green. The wall is white. The ball is orange. The lines are drawn in white. Some markers on the corners and the goals are in green.

Length of the game: The games consist of the first half, the break, and the second half. Each period lasts 10 minutes.

Wall: A wall that is the same height as the golf ball is placed all around the field, except around the goal areas. The wall is white.

Defense zone: The defense zone surrounds the goal area for each side. It is 22.5 cm from the goal line and is a width of 100 cm. The border of the defense zone is painted white and is 1-cm wide. Only one defense robot can enter this area. A brief passing and the accidental entry of other robots are permitted, but intentional entry and stay are prohibited.

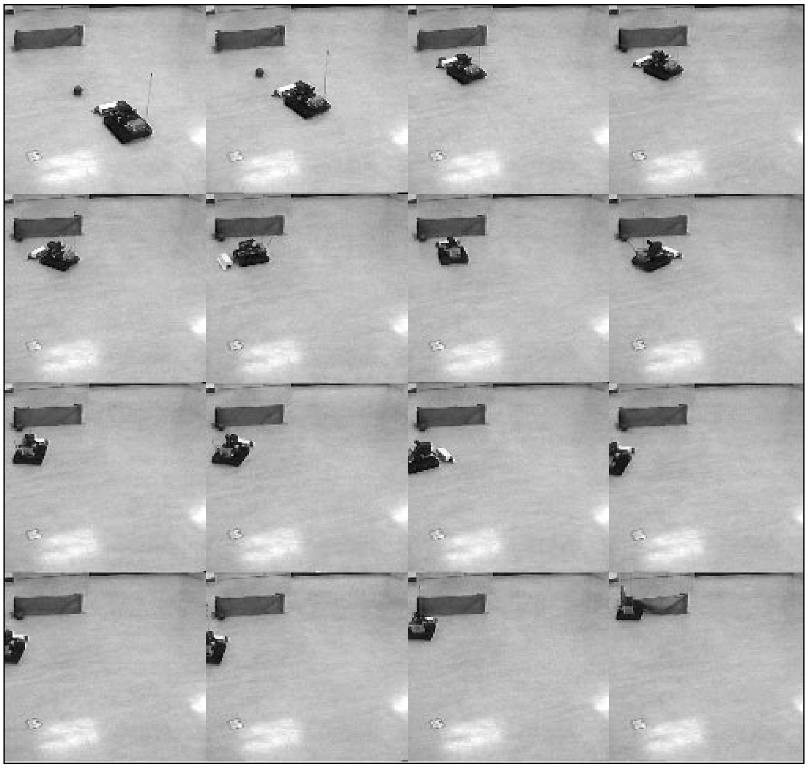


Figure 1. Photographs Showing a Robot Shooting a Ball into a Goal.

gies for the real-time animation of simulated soccer players and the network-based interactive multiuser server system. These are key technologies for network-based services in coming years. In this article, we briefly analyze some of these issues.

Agent Architecture

Existing robot players were designed to perform single behaviors such as pushing-dribbling-rolling (Asada et al. 1995; Sahota 1994; Connel and Mahadevan 1993a), juggling (Schaal and Atkeson 1994; Rizzi and Koditschek 1993), or hitting (Wattanabe et al. 1994). A RoboCup player should be designed so that it can perform multiple subtasks, such as shooting (including kicking), dribbling (pushing), passing, heading, and throwing a ball, all of which often involve a common behavior—avoiding the opponents.

Roughly speaking, there are two ways to build up a RoboCup player. The first approach is to design each component to be specialized for a single behavior and assemble the components into one. The other approach is to design one or two components that can perform multiple subtasks. The first approach seems easier to design but more difficult to assemble than the second. In addition, the problem of how to combine the reactive and deliberative approaches is a major research issue. To quickly react against the ball and move around the field, the use of subsumption architecture (Brooks 1986), or another reactive approach, might be effective. However, soccer players need to have global strategy as well as local tactics, which cannot be accomplished by mere reactive systems. The deliberation-based approach, which involves planning and reasoning, might be too slow to react to a quickly moving ball and cope with a dynamically changing environment. The agent architecture for RoboCup players needs to combine these approaches.

Physical Components

Because the RoboCup player should move around quickly, it should be compact; therefore, the development of the integrated multifunctional module should be a new target for the mechanical design of the RoboCup player. We need compact and powerful actuators with wide dynamic ranges. Also, we have to develop sophisticated control techniques to realize multiple behaviors by as few components as possible using low energy consumption.

The ultimate goal of a RoboCup player is a humanoid type that can run and kick or pass a ball with its legs and feet, can throw a ball with its arms and hands, and can do a heading with its head. Because the building of a team of the humanoid type seems impossible within the current technology, we are simply working on a demonstration track for now. However, we expect that sometime in the future, participants of RoboCup will overcome these technical difficulties and participate with humanoid robots.

In addition, an attempt is being made to provide standard physical components for robots. We are currently discussing the possibility of making a standard for autonomous robots. The standard OPEN R is not necessarily designed for RoboCup, but RoboCup is one of its significant application areas (Fujita and Kageyama 1997).

Vision and Sensor Fusion

The visual information is the richest source of information for perceiving not only the external world but also the effects of the robot actions. The computer-vision researchers have been looking for accurate three-dimensional geometry reconstructed from two-dimensional visual information, believing that three-dimensional geometry is the most powerful and general representation to be used in many applications, such as view generation for video database and robot manipulation and navigation. However, the time-consuming three-dimensional reconstruction might not be necessary or optimally encoded for the task given to the RoboCup player. To react to the situation in real time, the RoboCup player needs the information about which behavior to select for which situation. Because vision is a part of a complex system that interacts in specific ways with the world (Aloimonos 1994), we should make clear the role of vision in the context of a given task. RoboCup is one of such worlds that make it clear and evaluate the performance of the image processing that has been left ambiguous in the computer-vision field.

In addition to vision, the RoboCup player might need other sensing, such as sonar, touch, and force-torque, to discriminate the situations that cannot be discriminated from, or covered by, only visual information. Again, the RoboCup player needs the real-time processing for multisensor fusion and integration. Therefore, the deliberative approaches to obtain robust estimation by a multisensor system do not seem suitable. We should develop a method of sensor fusion-integration for RoboCup.

Learning Behaviors

The individual player has to perform several behaviors, one of which is selected depending on the current situation. Because of the uncertainties in sensory data processing and action execution, it is infeasible to program the robot behaviors to consider all situations; thus, robot-learning methods seem promising. As a method for robot learning, reinforcement learning, with little or no a priori knowledge and a higher capability for reactive and adaptive behaviors, has recently been receiving increased attention (Connell and Mahadevan 1993b). However, almost all the existing applications have been done with computer simulations in a toy world; real robot applications are few (Asada et al. 1995; Connel and Mahadevan 1993a). Because the prominence of the role of reinforcement learning is largely determined by the extent to which it can be scaled to larger and more complex robot-learning tasks, RoboCup seems a good platform.

One example of research on this issue (among other research, such as Stone and Veloso [1996]) is a project at Osaka University (Asada et al. 1996; Uchibe, Asada, and Hosoda 1996). Here, we only show some pho-

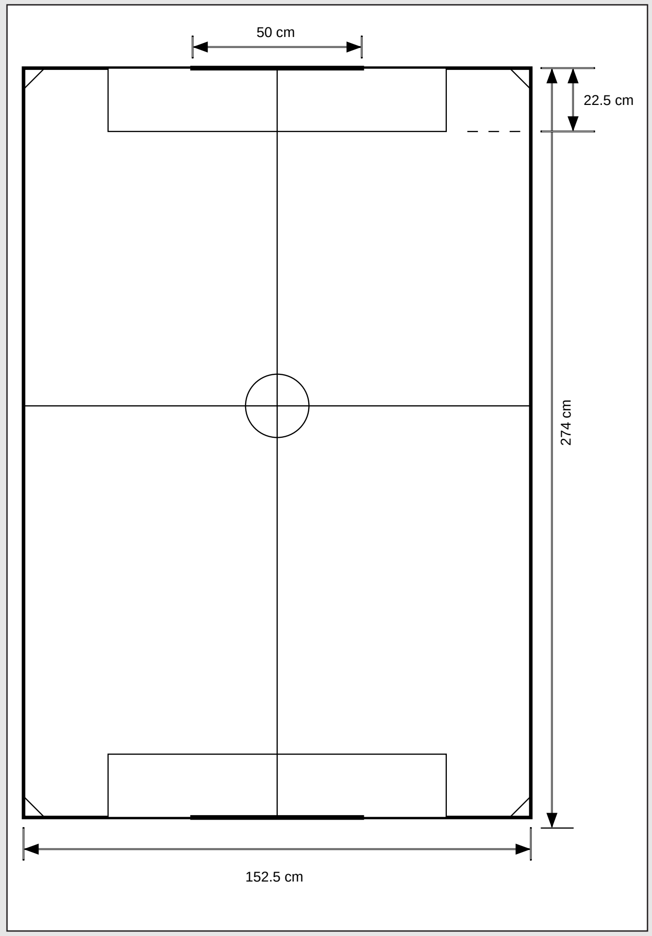


Figure 1. Top View of the Field for Small Robots.

Robot marking: Each robot should put at least one colored Ping-Pong ball on top of its body, at approximately 15 to 20 cm high. At their discretion, teams can use two Ping-Pong balls (of differing colors) on each robot to determine the orientation of the robot as well as its position. The color(s) of the Ping-Pong ball(s) identifies friend and enemy as well as positions using the global vision system.

Goal keepers: The goal keeper can hold and manipulate a ball for as long as 10 seconds within its penalty area. After releasing the ball, the keeper must not hold the ball until it touches any opponent or an alley outside the penalty area. If the ball released by the keeper reaches the other half-end of the court without touching any other player, the opponent is given an indirect free kick positioned anywhere along the half-way line (borrowed from the Futsal [a version of soccer to be played by five men] rule).

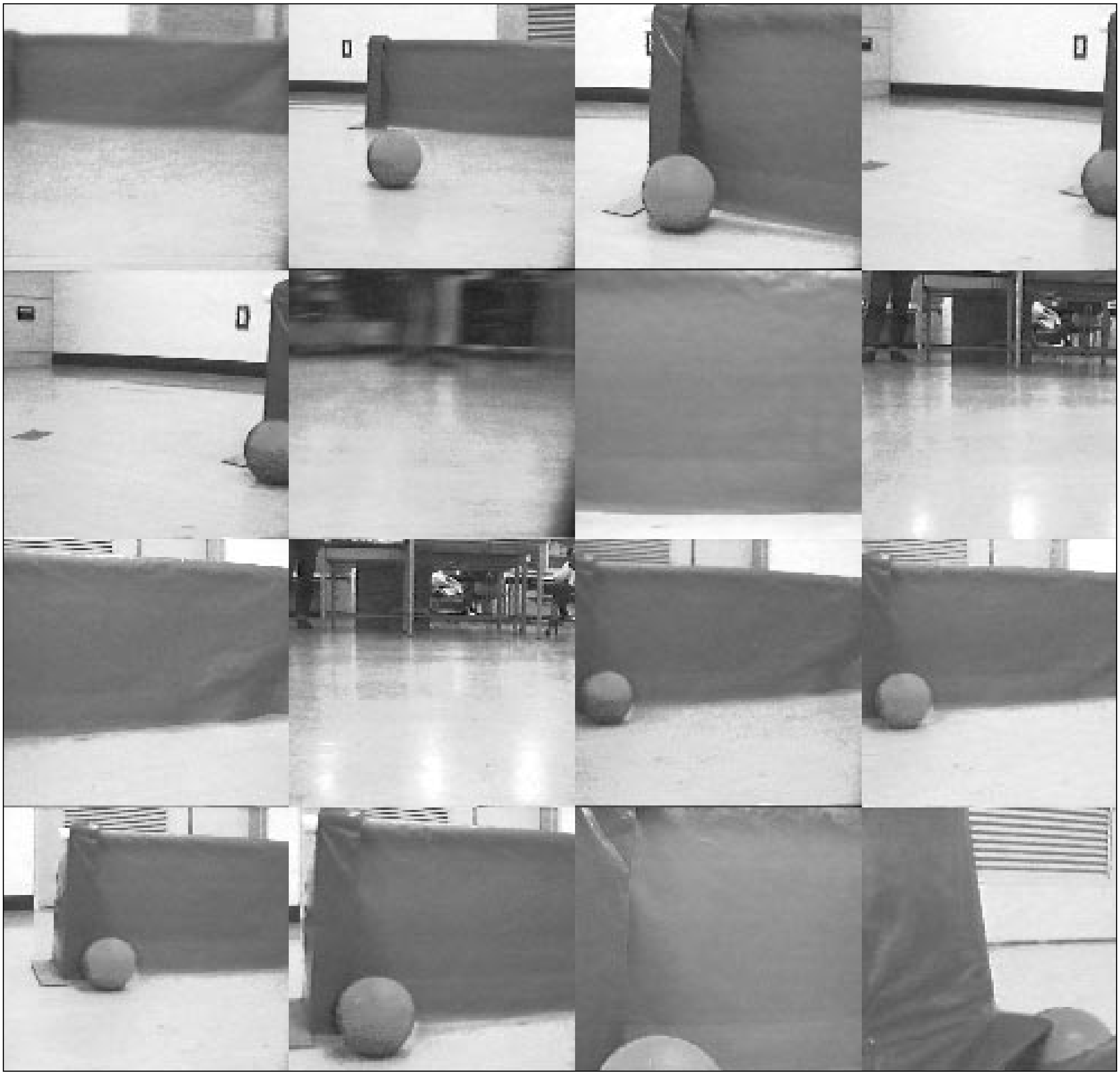


Figure 2. Photographs Taken by the Robot during the Task Execution Shown in Figure 1.

tographs of the robots in action; interested readers can access Asada's papers for details.

Figure 1 shows a real robot shooting a ball into a goal by using the state and action map obtained by the method proposed in Asada, Noda, and Hosoda (1996) and Takahashi, Asada, and Hosoda (1996). Sixteen images, taken every 1.5 seconds, are shown in raster order from the top left to the bottom right; the robot tries to shoot a ball but fails, then moves backward to find a position to shoot the ball, and finally succeeds in shooting. Figure 2 shows a sequence of images taken by the robot during the task execution shown in figure 1. Note that the backward motion for retry is just the result of learning and is not hand coded. They used an offline learning method for this execution. Currently, however, they use an online learning method (Uchibe, Asada, and Hosoda 1996).

Because the player has to consider the opponent's motions, the complexity of the problem is much higher than the problem of shooting without an opponent. To reduce the complexity, task decomposition is often used. Asada, Noda, et al. (1994) proposed a method of learning a shooting behavior and avoiding a goal keeper. The shooting and avoiding behaviors are independently acquired and are coordinated through learning. Their method still suffers from the huge state space and the perceptual aliasing problem (Whitehead and Ballard 1990) that results from the limited visual field.

Figure 3 shows a sequence of images in which the robot shoots a ball into a goal, avoiding the opponent (a goal keeper) (Asada, Uchibe, et al. 1994).

Multiagent Collaboration

From the viewpoint of distributed AI and multiagent research, a soccer game is a specific but attractive real-time multiagent environment. In a game, we have two competing teams. Each team has a teamwide common goal, namely, to win the game. The goals of the two teams are incompatible. The opposing team can be seen as a dynamic and obstructive environment, which might disturb the achievement of the common team goal. To fulfill the common goal, each team needs to score, which can be seen as a subgoal. To achieve this subgoal, each team member is required to behave quickly, flexibly, and cooperatively, taking local and global situations into account.

The team might have some sorts of global (teamwide) strategies to fulfill the common goal and both local and global tactics to achieve subgoals. However, consider the following challenges: (1) *the game environment*, that is, the movement of the team members and the opposing team, is highly dynamic; (2) the perception of each player could be limited locally; (3) the role of each player can be different; and (4) communication among players is limited, and therefore, each agent is required to behave flexibly and autonomously in real time under the resource-bounded situation. These restrictions are realistic and provide an interesting avenue of research for multiagent systems. Let's briefly look at multiagent research that addresses cooperative planning in a dynamic environment, where various resource and communication restrictions exist.

In cooperative distributed planning for common global goals, important tasks include the generation of promising local plans for each agent and the coordination of these local plans. When the dynamics of the

Fouls: Five different fouls are defined: (1) multiple defense, (2) ball holding, (3) court modification, (4) robot halting, and (5) offsides.

Multiple defense means more than one defense robot enters the defense zone to substantially affect the game. The foul is called, and the penalty kick is declared. *Ball holding* means taking full control of the ball by removing its entire degrees of freedom, typically fixing a ball to the body or surrounding a ball using the body to prevent access by others. A player cannot hold a ball unless it is a goal keeper in its penalty area. A free kick is declared. If ball holding occurs in the defense zone by the defense team, a penalty kick is declared. *Court modification* is modification or damage to the court and the ball. It is strictly forbidden. Should this foul occur, the game is suspended, and the appropriate restoration is done immediately, before the game can resume. *Robot halting* means all the players must be halted prior to kickoff or the restarting of the game. The judges check or adjust the placements of the players and declare the completion of adjustment after five seconds, before cuing a kickoff or a restart action. During these five seconds, the players can move. *Offside* rule is not adopted.

Charging: Unless striving for a ball, a player must not attack another. Such an act is regarded as a violent action. The umpire presents a red card to the responsible player, ordering it to leave the game. The judgment is based on external appearance. Throughout the game, if a player uses a device or an action that continuously causes serious damages to another robots' functions, the umpire can present a yellow warning card to the responsible player and order it to go outside the court and correct the problem. Once the correction is made, the robot can resume playing the game with the approval of the umpire. If the problem reoccurs, the umpire presents a red card to the responsible player, telling it to leave the game.

Aside from these items, no regulations are placed against possible body contact, charging, dangerous plays, obstructions, and so on.

Regulations for Medium-Size Robots

The regulations for medium-size robots are basically the same as for small robots. All sizes are multiplied by 3.

Field: The dimensions are 457.5 cm by 822 cm.

Robot: It should be within a 50-cm diameter.

Ball: The ball is a soccer ball (FIFA size 4 Futsal ball).

Goal and defense zones: The size of the goal and the defense zones are enlarged three times over those for small robots. Details are shown in figure 1.

Regulations for Other-Size Robots

On request, we define regulations for robots that do not fit the previously defined regulations. A larger-size robot

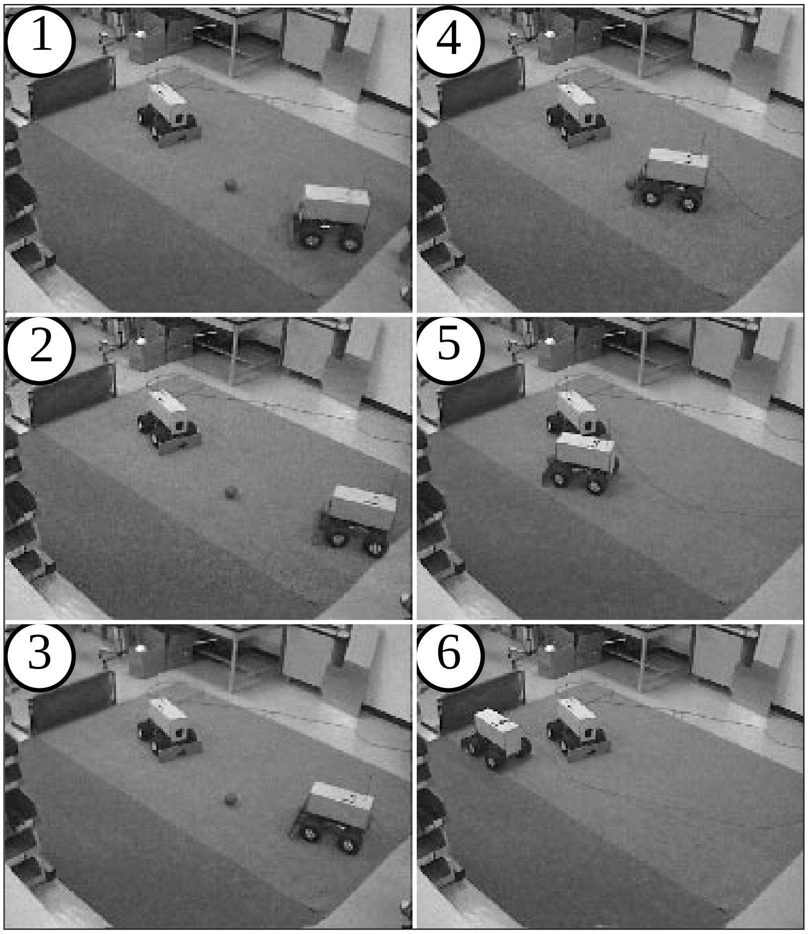


Figure 3. Photographs Showing the Robot Shoot a Ball into a Goal and Avoid the Opponent.

problem space, for example, the changing rate of goals compared with the performance of each planner, are relatively large, reactive planning that interleaves the plan generation and execution phases is an effective methodology at least for a single agent (Ishida and Korf 1991; Maes 1991; Agre and Chapman 1987; McDermott 1978). Whether this scheme extends naturally to the multiagent environment is an interesting issue.

For cooperative plan schemes, there are frequent changes in the problem space, or the observation of each agent is restricted locally. There is a trade-off between communication cost, which is necessary to coordinate the local plans of agents with a global plan, and the accuracy of the global plan (known as the *predictability-responsiveness trade-off*). A study of the relationship between the communication and processing costs and the reliability of the hypotheses in FA/C (Lesser and Erman 1980) and the relationship between the modification cost of local plans and the accuracy of a global plan in PGP (Durfee and Lesser 1987) illustrates this fact. Also, Korf (1987) addressed it theoretically. Tambe (1996) specifically used the RoboCup domain to test a scheme for joint-intention generation.

Schemes for reactive cooperative planning in dynamic problem spaces have been proposed and evaluated, sometimes based on the pursuit game (predator-prey) (Benda, Jagannathan, and Dodhiawalla 1985). However, the pursuit game is a relatively simple game. TILEWORLD was also proposed and studied (Ishida and Korf 1991; Kinny and Georgeff 1991). However, the environment is basically for the study of a single-agent architecture.

As is clear from this other research, RoboCup directly addresses a critical issue in multiagent systems research: the generation and execution of a cooperative plan under the dynamic environment. RoboCup provides an interesting and critically important task for multiagent cooperative planning.

RoboCup Simulator

For simulation, we use SOCCER SERVER (figure 4), a simulator of RoboCup developed by Itsuki Noda, ETL, Japan, which is a network-based graphic simulation environment for multiple autonomous mobile robots in a two-dimensional space. Using SOCCER SERVER, each client program can control each player on a soccer field with UDP-IP (user diagram protocol–internet protocol).

SOCCER SERVER provides a virtual field where players on two teams play a soccer (association football) game. Each player is controlled by a client program using local area networks (LANs). Control protocols are simple because it is easy to write client programs using any kind of programming system that supports UDP-IP sockets.

Control via Networks: A client can control a player through LANs. The protocol of the communication between clients and the server is UDP-IP. When a client opens a UDP socket, the server assigns a player to a soccer field for the client. The client can control the player using the socket.

Physical Simulation: SOCCER SERVER has a physical simulator, which simulates the movement of objects (ball and players) and collisions between them. The simulation is simplified so that it is easy to calculate the changes in real time, but the essence of soccer is not lost. The simulator works independently of communications with clients. Therefore, clients should assume that situations on the field change dynamically.

Figure 4. Screen of SOCCER SERVER.

Referee: The server has a referee module, which controls each game according to a number of rules. In the current implementation, the rules are (1) check goals; (2) check whether the ball is out of play; and (3) control positions of players for kickoffs, throw-ins, and corner kicks so that players on the defending team keep a minimum distance from the ball. Judgments by the referee are announced to all clients as an auditory message.

Although the current version of SOCCER SERVER does not implement detailed physical and visual simulations, as seen in Tu

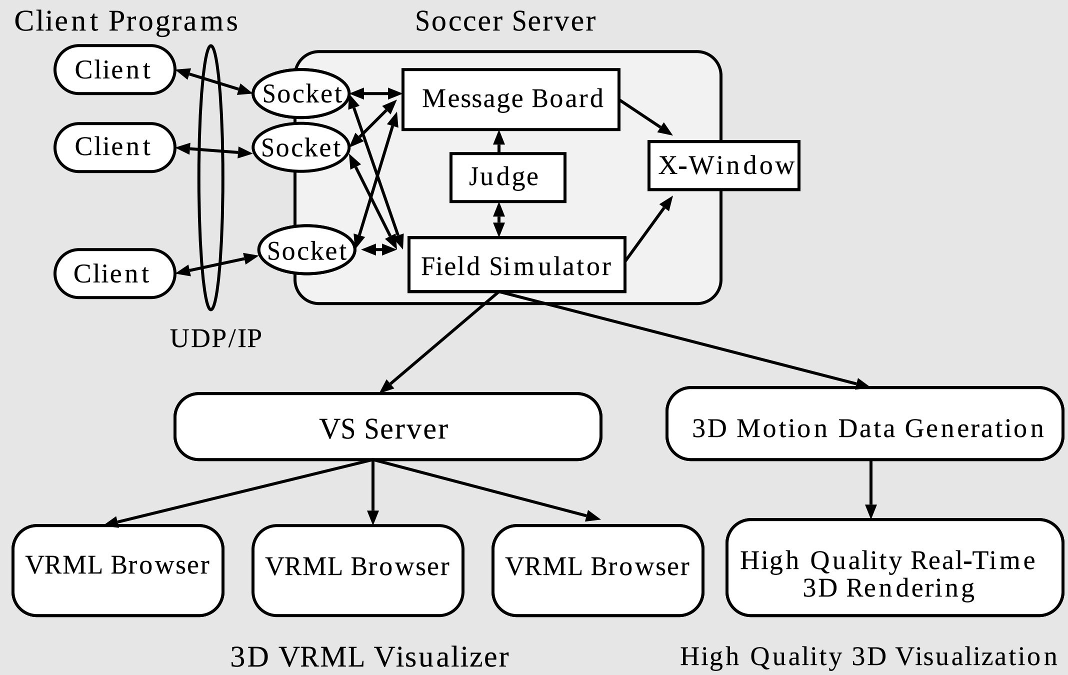


Figure 5. The Architecture of the Simulator and Visualization System.

and Terzopoulos (1994), we are planning to incorporate more realistic simulation in a future version.

Visualization

Although the current version of SOCCER SERVER only provides two-dimensional visualization, an independent server and browser systems provide three-dimensional visualization. Currently, we are planning to develop two types of three-dimensional visualization systems: (1) a high-quality real-time animation system for projection to the big screen or for TV broadcasting and (2) a virtual-reality modeling language (VRML)–based network broadcasting system.

We have already developed a prototype broadcast server for SOCCER SERVER that converts data from SOCCER SERVER into data necessary to display in three-dimensional animation. Converted data are broadcast to VRML browser software, so that viewers can observe games with three-dimensional animation. In addition, the browser's three-dimensional navigation capability enables viewers to navigate anywhere in the soccer field. The overall architecture of SOCCER SERVER and the visualization system is shown in figure 5.

Currently, we are developing several ranges of visualization system, such as a high-end three-dimensional animation system and an enhanced version of the VRML-based system. Our plan is to use kinemation data, either computer generated or captured by the motion-capture studio, of soccer players to drive the body motion of animated characters.

Conclusion

As it is clear by now, RoboCup provides fertile ground for AI and robotics research. The ultimate goal of RoboCup is so difficult that any near-term accomplishment is not feasible. There is a clear path to the stated goal, and each step toward the goal will generate a set of technologies that will affect industries for the next generation. Apart from this impact assessment, we believe that RoboCup contributes to AI and the robotics communities by providing exciting and publicly appealing goals, so that researchers in various fields can collaborate for the common goal. We hope RoboCup offers an opportunity for AI and robotics communities to revitalize their activities. "Let's get AI moving again!"

Note

1. To be fair, the Apollo mission was planned to gain the national prestige and demonstrate technical superiority over the former Soviet Union. Although no direct military advantage was gained by putting a few astronauts on the moon, technologies developed to achieve this goal were so significant that they formed the powerful technological and human foundations of American industry.

References

Agre, P., and Chapman, D. 1987. PENGI: An Implementation of a Theory of Activity. In Proceedings of the Sixth National Conference on Artificial Intelligence, 268–272. Menlo Park, Calif.: American Association for Artificial Intelligence.

Aloimonos, Y. 1994. RPLY: What I Have Learned. *CVGIP: Image Understanding* 60(1): 74–85.

Asada, M.; Noda, S.; and Hosoda, K. 1996. Action-Based Sensor Space Categorization for Robot Learning. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96), 1502–1509. Washington, D.C.: IEEE Computer Society.

Asada, M.; Noda, S.; Tawaratsumida, S.; and Hosoda, K. 1996. Purposive Behavior Acquisition for a Real Robot by Vision-Based Reinforcement Learning. *Machine Learning* 23:279–303.

Asada, M.; Noda, S.; Tawaratsumida, S.; and Hosoda, K. 1995. Vision-Based Reinforcement Learning for Purposive Behavior Acquisition. In Proceedings of the IEEE International Conference on Robotics and Automation, 146–153. Washington, D.C.: IEEE Computer Society.

Asada, M.; Uchibe, E.; Noda, S.; Tawaratsumida, S.; and Hosoda, K. 1994. Coordination of Multiple Behaviors Acquired by Vision-Based Reinforcement Learning. In Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS '94), 917–924. Washington, D.C.: IEEE Computer Society.

Benda, M.; Jagannathan, V.; and Dodhiawalla, R.

such as a Nomad-2000 or equivalent fits into the large robot league. We will create rules and a league when several teams express interest.

Regulations for the Simulation Track

Field of play: The field of play is provided by SOCCER SERVER, a simulator of a soccer field. A match is carried out in a server-client style: The server, SOCCER SERVER, provides a virtual field and simulates all movements of a ball and players. Clients become brains of players and control their movements. Communication between a server and each client is done using UDP-IP sockets on local area networks.

Players and teams: The simulation track of pre–RoboCup consists of a small track and a standard track. In the small track, each team has players. In the standard track, each team has players. There is no goalkeeper because players have no hands. If a team consists of fewer players than another team, a match is carried out without any penalties.

Client programs can be written by any programming systems, with the following restrictions: A client controls only a player. If a client controls multiple players, the different control modules of players are separated logically from each other. Clients cannot communicate directly with each other. Communication between clients must be done by facilities provided by SOCCER SERVER.

Rules: The referee module in SOCCER SERVER controls a match according to three rules: (1) goal, (2) out of field, and (3) clearance. A human referee also controls a match. When he/she judges a player's action to be too rough, for example, surrounding the ball, he/she suspends the match and restarts with a free kick by the opposite team.

Format of the competition: The competition is played in two rounds: In the first round, teams are divided into several groups of four teams. The system of play is the league system, each team playing one match against each of the other teams in the same group. The two teams coming in first and second in each group qualify for the second round. The second round is played by a system of elimination (cup system).

For details, please refer to the following World Wide Web home page and FTP cites: http://ci.etl.go.jp/noda/soccer/regulations/regulations.html, http://ci.etl.go.jp/noda/soccer/manual.newest/main.html, and ftp://ci.etl.go.jp/pub/soccer/server/.

1985. On Optimal Cooperation of Knowledge Sources, Technical Report BCS-G2010-28, Boeing AI Center, Seattle, Washington.

Brooks, R. 1986. A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation* RA-2: 14–236.

Connel, J. H., and Mahadevan, S. 1993a. Rapid Task Learning for a Real Robot. In *Robot Learning,* eds. J. H. Connel and S. Mahadevan, 105–139. Norwell, Mass.: Kluwer Academic.

Connel, J. H., and Mahadevan, S., eds. 1993b. *Robot Learning.* Norwell, Mass.: Kluwer Academic.

Durfee, E., and Lesser, V. 1987. Using Partial Global Plans to Coordinate Distributed Problem Solvers. In Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87), 875–883. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Fujita, M., and Kageyama, K. 1997. An Open Architecture for Entertainment Robots. In Proceedings of the First International Conference on Autonomous Agents. New York: ACM Press. Forthcoming.

Gasser, L.; Rouquette, N; Hill, R.; and Lieb, J. 1989. Representing and Using Organizational Knowledge in Distributed AI Systems. In *Distributed Artificial Intelligence, Volume 2,* eds. L. Gasser and M. N. Huhns, 55–78. San Francisco, Calif.: Morgan Kaufmann.

Ishida, T., and Korf, R. 1991. Moving Target Search. In Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91), 204–210. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Kennedy, J. F. 1961. Urgent National Needs. Speech to a Joint Session of Congress, 25 May 1961, Congressional Record—House, 8276.

Kinny, D., and Georgeff, M. 1991. Commitment and Effectiveness of Situated Agents. In Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91), 82–88. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Kitano, H.; Wah, B.; Hunter, L.; Oka, R.; Yokoi, T.; and Hahn, W. 1993. Grand Challenge AI Applications. In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93), 1677–1683. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Korf, R. 1992. A Simple Solution to Pursuit Games. Paper presented at the Eleventh International Workshop on Distributed Artificial Intelligence, February, Glen Arbor, Michigan.

Korf, R. 1987. Planning as Search: A Quantitative Approach. *Artificial Intelligence* 33(1): 65–88.

Lesser, V., and Erman, L. 1980. Distributed Interpretation: A Model and Experiment. *IEEE Transactions on Computers* 29(12): 1144–1163.

Levy, R., and Rosenschein, J. 1992. A Game-Theoretic Approach to Distributed Artificial Intelligence. In *Decentralized AI, Volume 3,* eds. E. Werner and Y. Demazeau, 91–99. New York: Elsevier.

McDermott, D. 1978. Planning and Action. *Cognitive Science, Volume 2,* 71–110. Norwood, N.J.: Ablex.

Maes, P. 1991. Situated Agents Can Have Goals. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back,* ed. Pattie Maes, 49–70. Cambridge, Mass.: MIT Press.

Osawa, E. 1995. A Metalevel Coordination Strategy for Reactive Cooperative Planning. In *Proceedings of the First International Conference on Multi-Agent Systems,* 297–303. Menlo Park, Calif.: AAAI Press.

Pollack, M., and Ringuette, M. 1990. Introducing the TILEWORLD: Experimentally Evaluating Agent Architectures. In Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90), 183–189. Menlo Park, Calif.: American Association for Artificial Intelligence.

Rizzi, A. A., and Koditschek, D. E. 1993. Further Progress in Robot Juggling: The Spatial Two-Juggle. In Proceedings of the IEEE International Conference on Robotics and Automation, 919–924. Washington, D.C.: IEEE Computer Society.

Sahota, M. K. 1994. Reactive Deliberation: An Architecture for Real-Time Intelligent Control in Dynamic Environments. In Proceedings of the Twelfth National Conference on Artificial Intelligence, 1303–1309. Menlo Park, Calif.: American Association for Artificial Intelligence.

Schaal, S., and Atkeson, C. G. 1994. Robot Learning by Nonparametric Regression. In Proceedings of the IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS ‘94), 478–485. Washington, D.C.: IEEE Computer Society.

Stone, P., and Veloso, M. 1996. Beating a Defender in Robotic Soccer: Memory-Based Learning of a Continuous Function. In *Advances in Neural Information Processing Systems 8,* eds. D. S. Tourtzky, M. C. Mozer, and M. E. Hasselmo, 896–902. Cambridge, Mass.: MIT Press.

Takahashi, Y.; Asada, M.; and Hosoda, K. 1996. Reasonable Performance in Less Learning Time by Real Robots Based on Incremental State Space Segmentation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96), 1518–1524. Washington, D.C.: IEEE Computer Society.

Tambe, M. 1996. Tracking Dynamic Team Activity. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, 80–87. Menlo Park, Calif.: American Association for Artificial Intelligence.

Tu, X., and Terzopoulos, D. 1994. Artificial Fishes: Physics, Locomotion, Perception, Behavior. Paper presented at the SIG GRAPH-94, 24 to 29 July, Orlando, Florida.

Uchibe, E.; Asada, M.; and Hosoda, K. 1996. Behavior Coordination for a Mobile Robot Using Modular Reinforcement Learning. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96), 1329–1336. Washington, D.C.: IEEE Computer Society.

Watanabe, H.; Nihna, Y.; Masutani, Y.; and Miyazaki, F. 1994. Vision-Based Motion Control for a Hitting Task—HANETSUKI. In Proceedings of the

IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS '94), 910–916. Washington, D.C.: IEEE Computer Society.

Whitehead, S. D., and Ballard, D. H. 1991. Learning to Perceive and Act. *Machine Learning* 7(1): 45–83.

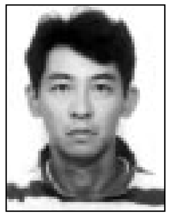
Hiroaki Kitano is a senior researcher at Sony Computer Science Laboratory. He received a B.A. in physics from the International Christian University, Tokyo, and a Ph.D. in computer science form Kyoto University. Since 1988, he has been a visiting researcher at the Center for Machine Translation at Carnegie Mellon University. Kitano received the Computers and Thought Award from the International Joint Conferences on Artificial Intelligence in 1993. His research interests include RoboCup, computational molecular biology, engineering use of the mophogenesis process, and evolutionary systems.

Yasuo Kuniyoshi is a senior research scientist at the Electrotechnical Lab, Agency of Industrial Science and Technology, Ministry of International Trade and Industry, Japan. His current research interests are imitation learning, cooperation by observation, and humanoid interaction. He received the Outstanding Paper Award at the Thirteenth International Joint International Conference on Artificial Intelligence.

Itsuki Noda is a researcher at the Electrotechnical Laboratory, Agency of Industrial Science and Technology, Ministry of International Trade and Industry, Japan. He received his B.Eng. in 1987 and his Dr.Eng. in 1995 from Kyoto University. His research interests include neural networks and multiagent systems.

Minoru Asada is a professor at Osaka University. He received his Ph.D. from Osaka University in 1982. He is a member of the Institute of Electrical and Electronics Engineers Computer, Systems, Man, and Cybernetics, and Robotics and Automation societies.

Hitoshi Matsubara is a senior researcher at the Electrotechnical Laboratory, Ministry of International Trade and Industry, Japan. He received his Ph.D. in computer science from the University of Tokyo in 1986. His current research interests include multiagent systems and game programming.



Ei-Ichi Osawa is a researcher at Sony Computer Science Laboratory. He holds a Ph.D. in computer science and a B.S. in mathematics. His Ph.D. dissertation concerned multiagent collaboration schemes.

For AAAI–97 Conference Information, Point Your Browser to

www.aaai.org/

Conferences/

National/1997/