RoboCup: The Robot World Cup Initiative*

Hiroaki Kitano[1], Minoru Asada[2], Yasuo Kuniyoshi[3], Itsuki Noda[3], Eiichi Osawa[1]

Sony Computer Science Lab.[1] 3-14-13 Higashi-Gotanda Shinagawa, Tokyo 141 Japan

Dept. of Mechanical Engineering[2] Osaka University Suita, Osaka 565 Japan

Electrotechnical laboratory[3] 1-1-4 Umezono Tsukuba, 305 Japan

Mailing-list: `RoboCup@csl.sony.co.jp`
Web: `http://www.robocup.org/RoboCup/`

Abstract

The Robot World Cup Initiative (RoboCup) is an attempt to foster AI and intelligent robotics research by providing a standard problem where wide range of technologies can be integrated and examined. The first **RoboCup** competition will be held at IJCAI-97, Nagoya. In order for a robot team to actually perform a soccer game, various technologies must be incorporated including: design principles of autonomous agents, multi-agent collaboration, strategy acquisition, real-time reasoning, robotics, and sensor-fusion. Contrary to AAAI robot competition, which is tuned for a single heavy-duty robot, 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 paper describes technical challenges involved in RoboCup, rules, and simulation environment.

RoboCup as a Standard AI Problem

We propose a Robot World Cup (RoboCup), as a new standard problem for AI and robotics research. This is a proposal to use a soccer game as a platform for a wide range of AI and robotics research, such as design principles of autonomous agents, multi-agent collaboration, strategy acquisition, real-time reasoning, and sensor-fusion. RoboCup aims at providing a standard task for research on fast-moving multiple robots, which collaborate to solve dynamic problems. 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. In addition, we intend to create an award for an expert robot, which demonstrates a high-level of competence for a specific task, such as shooting, intercepting, etc. Thus, RoboCup consists of three competition: the real robot competition, the software robot competition, and the special skills competition.

Standard AI problems have been the basic driving force for AI research. Research on computer chess, which is the most typical example of a standard problem, lead to the discovery of various powerful search algorithms. Other problems including, the Yale Shooting Problem and the Monkey-Banana, contributed to AI research by illustrating the essential difficulties involved in everyday reasoning. Criticisms against using such problems often focus on the fact that these are abstract tasks, which ignore essential difficulties of real world problem solving. Proponents of such criticism argue that the real world problem must be the target of serious research. While there is truth in such a claim, solving real world problems inherently involves domain-specific constraints and often social and economic constraints, which are not necessary common in other domains. In addition, research on usable real world systems are beyond the manpower and funding of many research groups. This hampers comparative studies of techniques for real world tasks. Thus, we need to setup a standard problem which is realistic, but affordable for many research groups.

The RoboCup is designed to meet the need of handling real world complexities, though in a limited world, while maintaining an affordable problems size and research cost. RoboCup offers an integrated research task covering the broad areas of AI and robotics. Such areas include: real-time sensor fusion, reactive behavior, strategy acquisition, learning, real-time planning, multi-agent systems, context recognition, vision, strategic decision-making, motor control, intelligent robot control, and many more.

Already, numbers of research results are reported (Drogoul and Duhaut 96; Stone and Veloso 96b; Bowling *et al.* 96; Achim *et al.* 96; Mizuno et al. 96; Yamaguchi *et al.* 96; Ch'ng and Padgham 96; Noda and Matsubara 96; Stone and Veloso 96b). A similar effort, but limited to micro-robots, was also proposed by KAIST, named MIROSOT.

*

Research Issues for RoboCup with Real Robots

In this section, we discuss several research issues involved in realizing real robots for RoboCup.

Design of RoboCup player and their control

Existing robot players have been designed to perform mostly single behavior actions, such as pushing/driblling/rolling (Connel and Mahadevan 93a; Asada et al. 95; Sahota 94), juggling (Rizzi and Koditschek 93; Schaal and Atkeson 94), or hitting (Watanabe *et al.* 94). 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; which often involves the common behavior of avoiding the opponents. Roughly speaking, there are two ways to build RoboCup players:

1. Design each component separately, which is specialized for a single behavior and then assemble them into one.
2. Design one or two components that can perform multiple subtasks.

Approach 1 seems easier to design but more difficult to build and *vice versa.* Since the RoboCup player should move around quickly it should be compact; therefore, approach 2 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 for as few as possible multiple behavior components with low energy consumption. The ultimate goal of a RoboCup player would be a humanoid type, that can run, kick and pass a ball with its legs and feet; can throw a ball with its arms and hands, and can do heading with its head. To build a team of humanoid type robots currently seems impossible, this is just future goal.

Vision and sensor fusion

Visual information is a rich source of information to perceive, not only the external world, but the effects of the robot's actions as well. Computer Vision researchers have been seeking an accurate 3-D geometry reconstructing from 2-D visual information, believing in that the 3-D geometry is the most powerful and general representation. This could be used in many applications, such as view generation for an video database, robot manipulation and navigation. However, the time-consuming 3-D reconstruction may not be necessary nor optimally suited for the task given to the RoboCup player. In order to react to the situation in real time, the RoboCup player quickly needs information to select behavior for the situation. we are not suggesting a special-purpose vision system, just that the vision is part of a complex system that interacts in specific ways with the world (Aloimonos 94). RoboCup is one of these worlds, which would make clear the role of vision and evaluate the performance of image processing which has been left ambiguous in the computer vision field.

In addition to vision, the RoboCup player might need other sensing devices such as: sonar, touch, and force/torque, to discriminate the situations that cannot be discriminated from only the visual information nor covered by visual information. Again, the RoboCup player needs the real time processing for multi-sensor fusion and integration. Therefore, the deliberative approaches with rough estimation using multi-sensor system does not seem suitable. We should develop a method of sensor fusion/integration for the RoboCup.

Learning RoboCup behaviors

The individual player has to perform several behaviors, one of which is selected depending on the current situation. Since programming the robot behaviors for all situations, considering the uncertainties in sensory data processing and action execution is unfeasible, robot learning methods seem promising. As a method for robot learning, reinforcement learning has recently been receiving increased attention with little or no *a priori* knowledge giving higher capability of reactive and adaptive behaviors (Connel and Mahadevan 93b). However, almost all of the existing applications have been done only with computer simulations in a virtual world, real robot applications are very few (Asada et al 94a; Asada et al. 95; Connel and Mahadevan 93a). Since the prominence of the reinforcement learning role is largely determined by the extent to which it can be scaled to larger and complex robot learning tasks, the RoboCup seems a very good platform.

At the primary stage of the RoboCup tournament, one to one competition seems feasible. Since the player has to take the opponent's motions into consideration, the complexity of the problem is much higher than that of simple shooting without an opponent. To reduce the complexity, task decomposition is often used. Asada et al. (Asada et al 94b; Uchibe *et. al,* 96b) proposed a method for learning a shooting behavior avoiding a goal keeper. The shooting and avoiding behaviors are independently acquired and they are coordinated through the learning. Their method still suffers from the huge state space and the perceptual aliasing problem (Whitehead and Ballard 90), due to the limited visual field. Sahota (Sahota 94) proposed a reactive deliberation approach to the architecture for real time intelligent control in a dynamic environment. He applied it to a one to one soccer-like game. Since his method needs global sensing for robot positions inside the field, it does not seem applicable to the RoboCup that allows the sensing capability only through the agents (see the rule section).

At the final stage, a many-to-many competition is considered. In this case, collective behaviors should be acquired. Defining all the collective behaviors as a team seems infeasible, especially, the situations where one of multiple behaviors should be performed. It is difficult to find a simple method for learning these behaviors, definition of social behaviors (Mataric 94). A situation would not be defined as the exact positions of all players and a ball, but might be perceived as a pattern. Alternatives, such as "coordination by imitation," should be considered.

In addition to the above, the problems related to the RoboCup such as task representation and environment modeling are also challenging ones. The other is how to construct the state space (Takahashi *et. al,* 96a). Of course, integration of the solutions for the problems mentioned above into a physical entity is the most difficult one.

Multi-Agent Collaboration

A soccer game is a specific but very attractive real-time multi-agent environment from the viewpoint of distributed artificial intelligence and multi-agent research. If we regard a soccer team as a multi-agent system, a lot of interesting research issues will arise.

In a game, we have two competing teams. Each team has a team-wide common goal, namely to win the game. The goals of the two teams are incompatible. The opponent 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; by taking local and global situations into account.

The team might have some sorts of global (team-wide) strategies to fulfill the common goal, and both local and global tactics to achieve subgoals. However, consider the following challenges:

1. the game environment, i.e. the movement of the team members and the opponent team, is highly dynamic.
2. the perception of each player could be locally limited.
3. the role of each player can be different.
4. communication among players is limited, therefore, each agent is required to behave very flexibly and autonomously in real-time under the resource bounded situation.

Summarizing these issues, a soccer team can be viewed as a cooperative distributed real-time planning scheme, embedded in a highly dynamic environment. In cooperative distributed planning for common global goals, important tasks include the generation of promising local plans at each agent and coordination of these local plans. The dynamics of the problem space, e.g. 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 known to be an effective methodology at least for a single agent (McDermott 78; Agre and Chapman 87; Maes 91; Ishida and Korf 91) to deal with these dynamic problems.

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 (this is known as the predictability/responsiveness tradeoff). The study of the relationship between the communication cost and processing cost concerning the reliability of the hypotheses in FA/C (Lesser and Erman 80), and the relationship between the modification cost of local plans and the accuracy of a global plan in PGP (Durfee and Lesser 87) illustrate this fact. Also, Korf addressed it theoretically in (Korf 87).

Schemes for reactive cooperative planning in dynamic problem spaces have been proposed and evaluated sometimes based on the pursuit game (predator-prey) (Benda *et al.* 85; Stephens and Merx 89; Gasser *et al.* 89; Levy and Rosenschein 92; Korf 92; Osawa 95). However, the pursuit game is a relatively simple game. Tileworld(Pollack and Ringuette 90) was also proposed and studied(Kinny and Georgeff 91; Ishida and Korf 91). However, the environment is basically for the study of a single agent arichtecture.

We see that a robot soccer game will provide a much tougher, fertile, integrated, exciting, and pioneering evaluation environment for distributed artificial intelligence and multi-agent research.

Rules for RoboCup

Regulations for RoboCup Real Robot Session (Summary)

General Policy 'Real worldness' in RoboCup mainly arises from the vast complexity of the overall situation due to interactions between behaviors and strategies of the ball and the players which cannot be fully predicted or controlled.

In the real robot session, we expect to have significantly greater complexity and hence much stronger reality than the simulation session. This is introduced by 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.

Due to the technical difficulty and unpredictability, the regulations can be adjusted to the overall situation of participating teams in each contest. However,

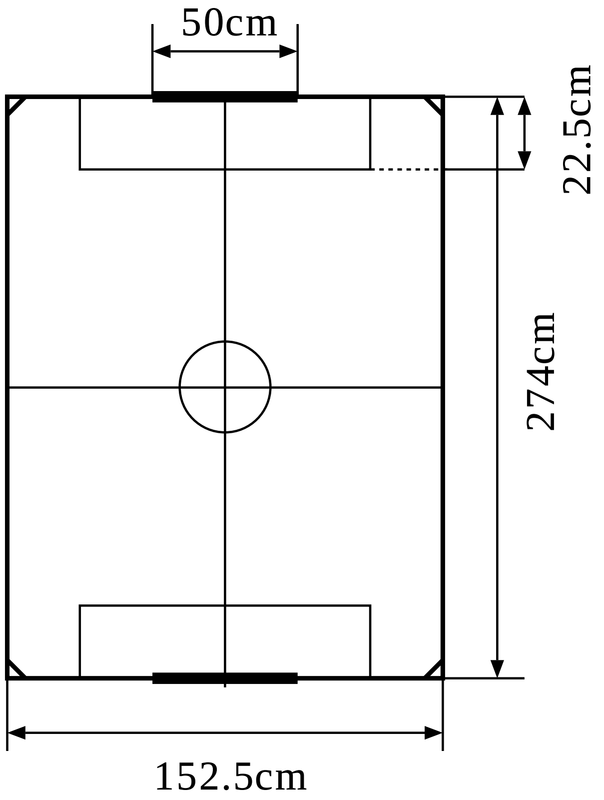

Figure 1: Top view of the field for small robots

the modifications must maintain the fairness to all the participants and must be announced in advance of the contest with an approval by the RoboCup technical committee.

The following sections summarize the current regulations very briefly. Since the rule may be modified to cope with various requests and technical issues, please obtain the most recent rule form the RoboCup web site. The complete version can be obtained from the RoboCup web site.

The Regulation for the Small Robots League

Field size: A ping pong table (a table tennis table) is used for the official match. The size and color of the table is officially determined as the international standard for ping pong. It is 152.5cm by 274cm, and color is green. Details shall be given in the figure 1.

Robot Size: The maximum diameter of the robot shall be within 15cm. This is approximately 1/10 of the length of the shorter end of the field.

Team: A team should consists of no more than 5 robots.

Goals: The width of the goal is 50 cm, which is approximately 1/3 of the length of the shorter end of the field.

Ball: Orange golf ball shall be used.

Colorings: Colors of each part of the field are as follows:

- Field shall be green.
- Wall shall be white.
- Ball shall be orange.
- Lines are drawn in white.
- Some markers on corners and goals are in green.

Length of the game: The games consits of the first half, break, and the second half. Each of them is 10 minutes.

Wall: A wall which is the same height as the golf ball shall be placed all around the field, except in goals. The wall shall be painted in white.

Defense Zone: Defense zone will be created in surrounding the goal of each side. It is 22.5 cm from the goal line, and width of 100 cm. The boarder of the defense zone will be painted in white, with the width of 1cm. Only one defense robot can enter this area. A brief passing and accidental entry of other robots are permitted, but intensional entry and stay is prohibited.

Robot marking: Each robot should put colored ping pong ball on top of their body, approximately between 15 cm to 20 cm in height. The color of the ping pong ball will be used to identify friend and enemy, as well as positions using the global vision system.

Fouls: Following fouls are defined:

Obstraction: When a robot intentionally attacks other robots, the foul will be called. This results in free kick.

Multiple Defense: When more than one defense robots enter the defense zone to substantially affects the game. The foul will be called, and the penalty kick will be declared.

Handling: When a robot hold the ball. The foul will be called, and the free kick will be declared.

The Regulations for the Medium Size Robots League The regulations for medium size robots basically applies the same rule as the rule for small robot. All sizes are multiled by 3. This means that:

Field: 457.5cm by 822cm

Robot: It should be within 45cm diameter

Ball: Soccer ball (which size of the standard soccer ball will be decided soon.)Size of goal and defense zone will be enlarged by 3 times.

Details will be shown in the figure.

Other size of robots Upon the request, we will define the regulations for robot which do not fit to above regulations.

Regulations of Simulation Track

This is an excerpt from the official regulation for RoboCup simulation track. For the complete rule, refer to the simulation track regulations in RoboCup home page (`http://www.robocup.org/RoboCup`), or directly to

`http://ci.etl.go.jp/~noda/soccer/regulations/regulations.html`

The 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: A 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 via UDP/IP sockets. Therefore users can use any kinds of programing systems that have UDP/IP facilities.

More detailed information about Soccer Server is available from the following WWW homepage.

`http://ci.etl.go.jp/~noda/soccer/server.html`

Sizes and Simulation The soccer field and all objects on it are 2-dimensional. This means that only horizontal movements are taken into account and vertical movements are ignored. The size of the field is $105\text{m} \times 68\text{m}$ [1].The width of goals is 14.64m, that is the doubled size of ordinary rules, because 2-dimensional simulation makes it difficult to get goals.

Players and a ball are treated as circles. Their movements are simulated stepwise one by one in a simple way: Velocity of an object is added to its position, while the velocity decays in a certain rate and increases according to its acceleration given by commands from clients. The collision is also simplified. If an object overlaps another object, that is, two objects collide with each other after a movement, the object is moved back until it does not overlap other objects. Then its velocity is multiplied by -0.1.

There also exist various landmarks, that is flags, lines and goals, on the field, which are visible but do not move. They are used for clients to determine their players' position. Clients can not know absolute positions of their own players, but know only the relative position to other objects. So clients need to determine their positions from the relative position of these fixed objects.

For more detail, see the manual of Soccer Server.

Players and Teams

Number of Players The simulation track of PreRoboCup consists of 'small track' and 'standard track'. In the small track, each team has $1 \sim 5$ players. In the standard track, each team has $6 \sim 11$ players. There is no goalkeeper because players have no hands.

Even a team consists of fewer players than another team, a match is carried out without any penalties.

Player Client Program As described earlier each player is a client program that connects Soccer Server. A client connects with the server by a UDP/IP socket. Using the socket, the client sends commands to control an assigned player and receives information from player's sensors . A couple of sample programs are available from the following WWW homepage.

`http://ci.etl.go.jp/~noda/soccer/client.html`

Communication between Clients All communication between clients is done via the server. Direct communication is not permitted. A client can send a message using a `say` command, then the message is immediately broadcast to all clients by `hear` information. The length of one message is a maximum of 256 bytes, and the server accepts only a command from a client per one simulation cycle, so that the capacity of communication is restricted.

Referees and Rules The match is basically controlled by the referee-module in Soccer Server automatically. However, because it is difficult to judge some of fouls like 'obstruction' and 'un-genglemanly play', a human referee judges such fouls.

The Match

The Duration of The Match The duration of the match is about 20 minutes (12000 simulation steps), consisting of two periods of 10 minutes with a 5-minute break at halftime. If the score is tied after 20 minutes, endless extra time will be played in 'golden-goal' style, in which the match ends immediately when one of both teams gets the first new goal.

The Format of The Competition The competition shall be played in two rounds.

In the first round, teams shall be divided into several groups of 4 teams. (The number of groups is depend on the total number of teams.) The system of play shall be the league system, each team playing one match against each of the other teams in the same group. Qualification in each group shall be determined as follows:

1. number of wins,
2. goal difference,
3. number of goals scored,
4. score of direct match,
5. drawing lots

[1] The unit is meaningless, because parameters used in the simulation do not be decided based on the physical parameters, but be tuned in order to make matches be meaningful as evaluation of multi-agent systems.

The two teams coming first and second in each group shall qualify for the second round.

The second round shall be played by a system of elimination (cup system).

If too many teams take part in the competition, it shall be played only by a system of elimination. This decision is done by the committee according to the number of teams and the infrastructure.

Robot Platform

Although building one's own robot for RoboCup is the most challenging aproach, not every one can afford to built their own robots. So those who are interested in physical robot competition, but not able to built their own robot, we are considering to set a standard for robot architecture and ask manifactures to provide robots for RoboCup. One candidate for such a standard is OpenR robot standard for entertainment robot (Fujita and Kageyama 97).

RoboCup Simulator

Soccer Server

In the simulation section, we will use Soccer Server, a simulator of **RoboCup** developed by Dr. Itsuki Noda, ETL, Japan, which is a network-based graphical simulation environment for multiple autonomous mobile robots in a 2D space. Using the soccer server, each client program can control each player on a soccer field via UDP/IP. This allows us to compare different types of multi-agent systems through the server, and test how well techniques of cooperation of agents work in dynamical varied situations.

The soccer server provides a virtual field where players of two teams play a soccer (association football) game. Each player is controlled by a client program via local area networks. Control protocols are simple in that 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 via local area networks. 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 via the socket.

Physical Simulation: The soccer server has a physical simulator, which simulates 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.

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; (3) Control positions of players for kick-offs, 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.

Figure 2: Screen of Soccer Server

Visualization

While the soccer server only provides 2 dimentional visualization, an indepdendent server and browser systems provide 3 dimensional visualization. We have developed a broadcast server (VS server) for the soccer server which convert data from the soccer server into data necessary to display in 3D animation. Converted data is broadcasted to Virtual Reality Modeling Language (VRML) browser software, CyberPassage, so that viewers can observe games with 3D animation. In addition, CyberPassage's 3D navigation capability enables viewers to navigate anywhere in the soccer field.

The overall architecture of the soccer server and the visualization system is shown in Figure 3.

Summary

In this paper, we proposed a RoboCup as a new standard AI problem. RoboCup provides rich research issues for a wide range of AI and robotics studies. Building robot soccer team would require a comprehensive research for the integration of broad range of technologies as well as fundermental breakthrough on some of key issues in agent design. We are currently inviting participation to this initiative, in order to define rules of play, develop a common research environment, and to host competitions and workshops. We wish RoboCup initiative would play an important role in

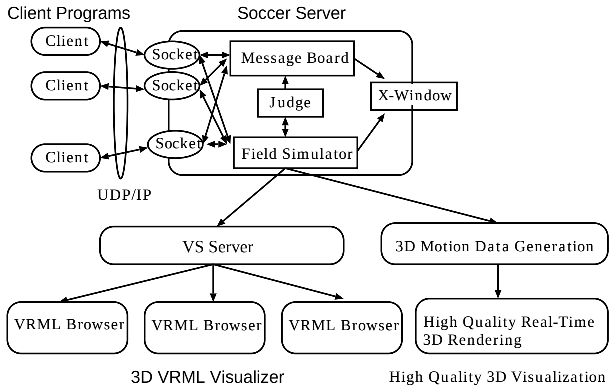

Figure 3: The Architecture of Simulator and Visualization System

promoting the state-of-the-art in AI and robotics research.

RoboCup Information

Web: `http://www.robocup.org/RoboCup`

International Mailing-List:
`RoboCup@csl.sony.co.jp` (Unmoderated) To register, send e-mail to `majordomo@csl.sony.co.jp`, with a content line **subscribe robocup**.

References

S. Achim, P. Stone, and M. Veloso, Building a Dedicated Robotic Soccer System. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

P. Agre and D. Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pp. 268–272, 1987.

Y. Aloimonos. "Rply: What i have learned". *CVGIP: Image Understanding*, 60:1:74–85, 1994.

M. Asada, S. Noda, S. Tawaratsumida, and K. Hosoda. Vision-based reinforcement learning for purposive behavior acquisition. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1995.

M. Asada, S. Noda, S. Tawaratsumida, and K. Hosoda. "purposive behavior acquisition on a real robot by vision-based reinforcement learning". In *Proc. of MLC-COLT (Machine Learning Conference and Computer Learning Theory) Workshop on Robot Learning*, pages 1–9, 1994.

M. Asada, E. Uchibe, S. Noda, S. Tawaratsumida, and K. Hosoda. "coordination of multiple behaviors acquired by vision-based reinforcement learning". In *Proc. of IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS '94)*, pages 917–924, 1994.

M. Benda, V. Jagannathan, and R. Dodhiawalla. On Optimal Cooperation of Knowledge Sources. Technical Report BCS-G2010-28, Boeing AI Center, 1985.

M. Bowling, P. Stone, and M. Veloso, Predictive Memory for an Inaccessible Environment. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

S. Ch'ng and L. Padgham, Role Organisation and Planning Team Strategies. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

J. H. Connel and S. Mahadevan. "Rapid task learning for real robot". In J. H. Connel and S. Mahadevan, editors, *Robot Learning*, chapter 5. Kluwer Academic Publishers, 1993.

J. H. Connel and S. Mahadevan, editors. *Robot Learning*. Kluwer Academic Publishers, 1993.

A. Durogoul and D. Duhaut, MICROB: Making Intelligent Collective Robotics. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

E. Durfee and V. Lesser. Using Partial Global Plans to Coordinate Distributed Problem Solvers. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*, 1987.

M. Fujita and K. Kageyama, An Open Architecture for Robot Entertainment. In *Proceedings of the First International Conference on Autonomous Agents (Agent-97)*, 1997.

L. Gasser, N. Rouquette, R. Hill, and J. Lieb. Representing and Using Organizational Knowledge in Distributed AI Systems. In Les Gasser and Michael N. Huhns, editors, *Distributed Artificial Intelligence, Volume II*, pp. 55–78. Morgan Kaufmann Publishers, Inc., 1989.

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

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

R. Korf. Planning as Search: A Quantitative Approach. *Artificial Intelligence*, Vol. 33, No. 1, pp.65–88, 1987.

R. Korf. A Simple Solution to Pursuit Games. In *Proceedings of the Eleventh International Workshop on Distributed Artificial Intelligence*, 1992.

V. Lesser and L. Erman. Distributed Interpretation: A Model and Experiment. *IEEE Transactions on Computers*, Vol. 29, No. 12, pp.1144–1163, 1980.

R. Levy and J. Rosenschein. A Game Theoretic Approach to Distributed Artificial Intelligence. In Eric Werner and Yves Demazeau, editors, *Decentralized A.I. 3*. Elsevier/North Holland, 1992.

P. Maes. Situated agents can have goals. In Pattie Maes, editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pp. 49–70. The MIT Press, 1991.

M. J. Mataric. Learning to behave socially. In *Proc. of the 3rd Int. Conf. on Simulation and Adaptive Behaviors – From animals to animats 3 –*, pages 453–462, 1994.

D. McDermott. Planning and Action. *Cognitive Science*, Vol. 2, pp.71–110, 1978.

H. Mizuno, M. Kourogi, and Y. Muraoka, Building shootbot possible to dribble and shoot. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

I. Noda and H. Matsubara, Soccer Server and Researches on Multi-Agent Systems. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

I. Nourbaaksh, et al., The Winning Robots from the 1993 Robot Competition. *The AI Magazine*, Vol. 14, No. 4, 51-62, The AAAI Press, 1993.

E. Osawa. A Metalevel Coordination Strategy for Reactive Cooperative Planning. In *Proceedings of the First International Conference on Multi-Agent Systems*, 1995.

M. Pollack and M. Ringuette. Introducing the Tileworld: Experimentally Evaluating Agent Architectures. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pp. 183–189, 1990.

A. A. Rizzi and D. E. Koditschek. Further progress in robot juggling: The spatial two-juggle. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 919–924, 1993.

M. K. Sahota. Reactive deliberation: An architecture for real-time intelligent control in dynamic environments. In *Proc. of AAAI-94*, pages 1303–1308, 1994.

S. Schaal and C. G. Atkeson. Robot learning by nonparametric regression. In *Proc. of IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS '94)*, pages 478–485, 1994.

L. Stephens and M. Merx. Agent Organization as an Effector of DAI System Performance. In *Proceedings of the Ninth Workshop on Distributed Artificial Intelligence*, pp. 263–292, 1989.

P. Stone, and M. Veloso, "Beating a defender in robotic soccer: Memory-based learning of a continuous function," In David S. Tourtzky, Michael C. Mozer, and Michael E. Hasselmo, editors, *Advances in Neural Information Processing Systems 8,* Cambridge, MA, MIT Press, 1996.

P. Stone, and M. Veloso, Using Machine Learning in the Soccer Server. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.

P. Stone, and M. Veloso, A Layered Approach for an Autonomous Robotic Soccer System. In *Proceedings of the First Conference Autonomous Agents (Agent-97)*, Marina del Ray, 1997.

M. Tambe, "Tracking Dynamic Team Activity " In *Proceedings of AAAI-96,* Portland, 1996.

E. Uchibe, M. Asada, and K. Hosoda. Reasonable performance in less learning time by real robot based on incremental state space segmentation. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96)*, pages ???–???, 1996.

E. Uchibe, M. Asada, and K. Hosoda. Behavior coordination for a mobile robot using modular reinforcement learning. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems 1996 (IROS96)*, pages ???–???, 1996.

H. Watanabe, Y. Nihna, Y. Masutani, and F. Miyazaki. Vision-based motion control for a hitting task - hanetsuki -. In *Proc. of IEEE/RSJ/GI International Conference on Intelligent Robots and Systems 1994 (IROS '94)*, pages 910–916, 1994.

S. D. Whitehead and D. H. Ballard. "Active perception and reinforcement learning". In *Proc. of Workshop on Machine Learning-1990*, pages 179–188, 1990.

T. Yamaguchi, Y. Nomura, Y. Tanaka, and M. Yachida, Acquiring Various Behaviors by Isomorphism of Actions in Reinforcement Learning. In *Proceedings of the IROS-96 Workshop on RoboCup*, Osaka, 1996.