Multi-Agent Systems: A Survey from the Robot-soccer Perspective

Jong-Hwan Kim and Prahlad Vadakkepat

Department of Electrical Engineering,

Korea Advanced Institute of Science and Technology (KAIST),

373-1 Kusong-dong, Yusong-gu, Taejon 305-701, Republic of Korea.

{johkim,prahlad}@@vivaldi.kaist.ac.kr

(Int. J. Intelligent Automation and Soft Computing, 6: (1) 3-17, 2000.)

Contents

List of Figures

List of Tables

Multi-Agent Systems: A Survey from the Robot-soccer Perspective

Jong-Hwan Kim and Prahlad Vadakkepat
Department of Electrical Engineering,
Korea Advanced Institute of Science and Technology (KAIST),
373-1 Kusong-dong, Yusong-gu, Taejon 305-701, Republic of Korea.
{johkim,prahlad}@@vivaldi.kaist.ac.kr
(Int. J. Intelligent Automation and Soft Computing, 6: (1) 3-17, 2000.)

Abstract

One of the most challenging goals in artificial intelligence (AI) is the development of artificial intelligent autonomous agents with human-level performance. The past few years has witnessed a tremendous interest in research and discussions on intelligent agents. With the ever increasing number of robots in an industrial environment, scientists and technologists are often faced with issues on cooperation and coordination among different robots and their self governance in a common work-space. This has led to developments in multi-robot cooperative autonomous systems. With an aim to study issues such as group architecture, resource conflict, origin of cooperation, learning and geometric problems, groups of robots are constructed. The proponents of multi-robot autonomous systems needed a model to validate the theories being proposed and to test their efficacy and efficiency. It is no surprise that they started focussing on robot soccer. Robot soccer makes heavy demands in all the key areas of robot technology, mechanics, sensors and intelligence. It does so in a competitive setting that people around the world can understand and enjoy. This paper gives a survey, though not exhaustive, on the present scenario of multi-agent systems and cooperative robotics. The robot-soccer system is discussed as a test benchmark for multi-robot systems.

Keywords: Agents, multi-agent systems, multiple robotic systems, robot soccer systems, evolutionary robotics and MiroSot.

1 Introduction

The word *agent* has become modish and popular in recent years. There exist agent technologies for entertainment, information retrieval, service management, manufacturing management, business process management, service robotics, cooperative tasks management, to cite a few. Agents can have capabilities like autonomous behaviors, interaction with humans or other agents, goal directed actions and adaptation to changes in environment, making it possible to bring new features to applications. The use of agent technologies in different areas is in a juvenile state and many research activities are being reported from various corners recently. A Foundation for Intelligent Physical Agents (FIPA) [1] was established recently, to share efforts to produce specifications of generic agent technologies and to help agent technologies get across to the market place.

The autonomous behaviors of an agent make it interactive with the real world. An autonomous agent can be alone in the environment or there can be others. The multi-agent scenario can be homogeneous or heterogeneous and it may be possible for them to communicate with each other often. Agents can be equipped with knowledge, motivations, reasoning and planning capabilities.

The agent technologies and agent-oriented programming are widely used in computer science and they play a crucial role in the development of complex distributed systems. Robot agents and multi-robotic systems are in the focus within the robotic community. Robotics/intelligent machines has become a matured and accepted area of technology in developed countries. The research on robotics is related to the needs of the scientific community, industrial and governmental policies/strategies, social needs, availability of labor and the international competitiveness [2].

Multi-agent systems deal with research subjects such as cooperation protocol by distributed control, effective communication and fault tolerance, while having efficiency of cooperation, adaptation, robustness and being in real-time [3]. Multi-agent systems (control architecture, communication scheme, vision tracking algorithm, sensing, fault tolerance, etc.) have been studied by several research groups in recent years. It is a basic problem to determine what action an agent should take in a given situation. Research on such action selection problems has been done, see [4, 5, 6]. In particular, a dynamic environment with competitions between agents makes the problem more difficult and complex.

The present day industrial environments deal with much complex systems and multiple robots are easier, cheaper, flexible and fault-tolerant to build and use. Multiple mobile robots can accomplish performance benefits and are not spatially constrained as a single powerful robot. The coordination of multi-agents is associated with lot of supplementary work in advance. The associated issues are position correction, prevention of communication congestion, local information sensing, in addition to the need for imitating human-like decision making. The research directions in multi-robotic systems will help to unveil fundamental issues in social sciences as more and more researchers are imbibing realistic models from them.

The MiroSot (Micro-Robot World Cup Soccer Tournament) initiated in 1996 is involved with micro robots playing on a playground of $130(cm) \times 90(cm)$ as per well defined game rules [7]. The Federation of International Robot-soccer Association (FIRA) [8] was formed in 1997 to bring together researchers and scientists working in the areas of robotics, AI, intelligent control, computer technology, sensor technology, etc. FIRA has different categories of games namely, MiroSot, Nano Robot World Cup Soccer Tournament (NaroSot), Robot World Cup Soccer Tournament (RoboSot), Khepera Robot World Cup Soccer Tournament (KheperaSot) and Humanoid Robot World Cup Soccer Tournament (HuroSot) with different sizes of robots.

Robot soccer game is associated with cooperation, decision making, planning, modeling, learning, robot architecture, vision tracking algorithm, sensing, communication, and so forth. It also has an explicit performance measure, the score.

Different types of schemes for soccer robot control are available [9]. In a remote-brainless (vision-based) soccer robot scheme, a host computer controls the robots by commanding the robot velocities like a radio-controlled car. In a brain on-board (vision-based) soccer robot scheme, the robots will make moves as per the vision data, keeping away from obstacles autonomously. In a robot-based scheme, each autonomous robot can make a decision based on information it collects with its sensors and, if needed, can communicate with other robots. Robots which can be operated autonomously are needed for the robot-based system. The control structure, behaviors and actions of robots will greatly influence their performance in robot-soccer games.

This paper is organized as follows. Section 2 deals with agents and multi-agent scenarios briefly. Multiple robotic systems and associated issues are discussed in Section 3. Robot soccer and some representative robot-soccer systems are discussed in Section 4. The conclusion is given in Section 5.

2 Agents and Multi-Agent Systems

The widespread use of distributed information services will radically alter the way in which both organizations and individuals work. There are many indicators of this coming information revolution.

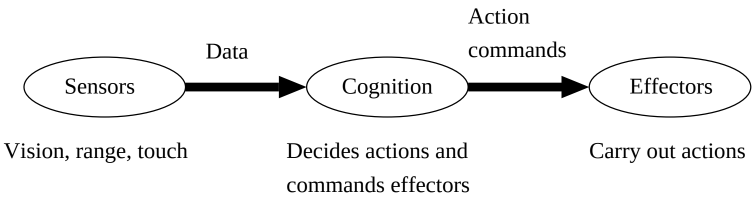

Figure 1: Basic architecture of an autonomous agent

2.1 Autonomous agents

Russell and Norvig [10] define an agent as

"Anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors."

Though there exists no generally accepted definition, an *agent* can be thought of as an entity with domain knowledge, goals and actions. An autonomous agent is capable of interacting independently and effectively with its sensors and effectors in order to achieve specified or self-sired goals (Figure 1). The study of autonomous agent behaviors, the way in which an autonomous agent acts, is advancing rapidly as a sub-field of *distributed artificial intelligence* (DAI). Creating an intelligent autonomous agent to the level of human performance is one of the dreams of AI researchers.

Autonomous agents vary from software agents (*softbots*) [11], to autonomous mobile robots or vehicles. Categorization of autonomous agents is tantamount to ascertaining the degree with which they interact with the real world. Two major features, *embodied* or *situated*, help to categorize autonomous agents. The agents with least interaction with the real world are neither embodied nor situated. Computer simulation of actual agents falls in this category. The traditional industrial robots are embodied (they have physical bodies) and not situated (their behavior does not depend on the current state of the environment). A computerized banking system is situated (changes its behavior depending on the transactions), but is not embodied (being not truly physical). Autonomous mobile robots are embodied and situated. Adaptive, robust, tactical and versatile behaviors are the requirements for agents to be autonomous [12], though all of these are not possible to impart to them with the present state of the technology. Adaptive behavior makes an agent accommodate the environmental variations. Fault tolerance to environmental properties makes them robust. Tactical behavior is related to maintenance of multiple goals and the pursuit of a particular one among them as per the situation. Multi-purpose behavior (capable of carrying out a variety of tasks) makes them versatile in nature. Different properties of agents help to classify them (Table 1) [14].

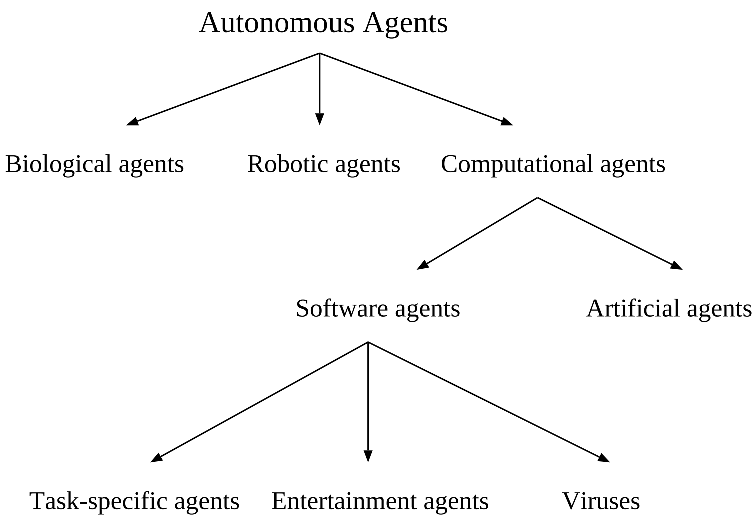

Figure 2: Taxonomy of autonomous agents

Figure 2 shows the taxonomy of autonomous agents [15]. The basic models for agents are either biological or mathematical in nature. In the first level, agents are classified into biological, robotic and computational in type. The later is further subdivided into software and artificial agents. Task-specific agents, entertainment agents and viruses fall within the software class.

Further classification based on control structures, environments (data base, file system, network, internet), language, and applications are also possible [14]. Centralized versus distributed classes stem from the nature in which agents are controlled.

Man made agents work in a rational manner to perform a function with specified *goals* with their success rates determined through some performance measures. Such agents vary from chess-playing programs to the Sojourner Mars exploration robot [16]. Robot agents designed to play the game of soccer have the specified objective to score goals while keeping the ball away from the opponent robot agents.

Autonomous agents are helpful for operation in hazardous environments, can aid the disabled and, are useful in outer space, deep-sea and planetary explorations. The use of autonomous agents in repetitive and strenuous tasks is quite a boon to humans. They can increase productivity and quality, making way for economic profitability. Researchers are hopeful of making personal robots in the following century.

2.2 Agent theories, languages and architectures

Agent theories, *agent architectures* and *agent languages* are the different axes in which study on agents is being carried out [17, 18, 19]. The properties of agents, and the formal represen-

Table 1: Classification of agents

Property	**Meaning**
Reactive	responding to environmental variations in a timely fashion
Autonomous	able to exercise control on its own actions
Goal-oriented	pro-active purposeful, not acting simply in response to the environment
Temporally continuous	continuously running
Communicative	communicates with other agents, perhaps including people
Learning/adaptive	behavior changes as per experience
Mobile	able of transporting itself
Flexible	non-scripted actions
Character	emotional state and personality

tation and reasoning methods of the same are dealt within agent theories. The study on agent architecture is related to the construction of agents, satisfying a set of defined properties, using appropriate software and/or hardware structures. Programming, compilation, and execution of agent languages with right primitives is a field of research of its own. Agent languages allow one to incorporate beliefs, desires and other concepts put forward by agent theorists in computer programs. *Agent oriented programming* helps to fix the states of the agents consisting of beliefs, capabilities, and decisions with precisely defined syntax [20].

The traditional top-down design of autonomous agents is known as the *deliberative* approach and the recent bottom-up design as the *reactive* approach. The top-down approach uses modularized cognitive abilities (perception, world modelling, planning, etc.), thus functionally decomposing the cognitive component. One or more of the agents will monitor the global performance of the system and restructure its organization when needed. The traditional planning is very time consuming and unless the human-level intelligence is clearly understood, it is difficult to decompose it into the proper components with right interfaces among them. The bottom-up design starts with the implementation of simple behavior, covering the complete range from perception to action, and then incrementally adding more sophisticated behaviors [12]. The *reactive* agents with simple behavior schemes, react to environmental changes and there is no need for world modelling and planning. The agents based on local estimates of the performance of the organization will initiate local restructuring within its neighborhood. However, they have limitations while handling tasks requiring knowledge about the world that must be obtained by reasoning or from memory, rather than from perception [21]. A combination of these two approaches with high-level reasoning and low-level reactive capabilities as a *hybrid* approach is suggested in [22]. A discussion on the same is beyond the scope of this article.

2.3 Multi-agent systems

An agent may have intelligence, autonomous behaviors, and can be equipped with knowledge, motivations, reasoning and planning capabilities. Self-adaptation, learning and capability to perceive environmental changes are some of the features associated with autonomous agents. Agents can be constructed and maintained separately.

Multi-agent systems deal with the construction of complex systems involving multiple agents and their coordination. It inherits from the distributed AI. Distributed AI (DAI) systems are not capable of emergent contexts and changing problem-solving roles for concerned agents. There are different issues faced by DAI researchers [23, 24] namely,

1. how to formulate, describe, decompose, and allocate problems and synthesize results among a group of intelligent agents,
2. how to enable agents to communicate and interact: what communication language or protocol to use, and what and when to communicate,
3. how to ensure that agents act coherently in making decisions or taking action, accommodate the nonlocal effects of local decisions and avoid harmful interactions,
4. how to enable individual agents to represent and reason about actions, plans and knowledge of other agents in order to coordinate with them,
5. how to recognize and reconcile disparate viewpoints and conflicting intentions among a collection of agents trying to coordinate their actions, and
6. how to engineer and construct practical DAI systems; how to design technology platforms and the development methodologies for DAI.

2.3.1 Learning in multi-agent systems

Learning in multi-agent systems can help in each of the issues in Section 2.3. A system can learn problem decomposition by acquiring instances from a human operator and by generalizing on them. A common language helps agents to work together as a system, paving a way for negotiations among them to resolve conflicts. Learning from experience can help an agent to reduce the need for negotiations or make it more directed, avoiding harmful interactions. An agent can learn other agents' intentions and beliefs as well as the characterization of the task environment. The modelling of another agent's goals and beliefs will enhance its ability to reason about other agents and improve its ability to coordinate its own activities. It helps to reconcile the conflicting intentions within and/or between agents. Adaptive agents are those which change their behaviors in uncertain and dynamical environments. To cope with such situations, an agent must be self-adaptive, self-learning, and should have the capabilities to face real situations through cooperative behaviors with other agents, as well as, be capable of perceiving environmental changes. The knowledge and intelligence can not be programmed as such in the beginning, however it can be acquired from real environments and through reactive behaviors with other agents. Reinforcement learning [25, 26] and classifier systems [27] have laid the basis for learning in multi-agent systems.

2.3.2 Multi-agent system structures

A multi-agent system (MAS) is a distributed computing system with autonomous interacting intelligent agents that coordinate their actions so as to achieve its goal(s) jointly or competitively. The agents in a multi-agent scenario, may have homogeneous or heterogeneous structures. In a homogeneous MAS, all of the agents have identical structure (goals, domain knowledge, and set of actions). They may differ by way of their sensor input and effector output. In a heterogeneous situation, agents can have different goals, domain knowledge and actions. The agents in such a system may be friendly (benevolent) or may be inhibiting each other (competitive). The agents, situated differently in the environment, receive different sensory inputs and initiates different actions. Heterogeneity imparts much power to the MAS at the cost of complexity.

Behavior emergence takes place through cooperation which can make use of the individually acquired intelligence as well. Agent based architecture offers modularity, robustness and separation of concerns in addition to other advantages of a distributed system.

The two general types of designs seen among the multi-agent system structures are the hierarchical and behavior structures. They differ on the type of information they process and in their interconnections [28]. In a hierarchical structure, the control issue is divided along functional lines into progressive *levels of abstraction* of data. It uses computational functions for system decomposition. Processes handling data deal with different information content and hence such a system sequences data from a perception process, to a decision-making process, through a series of action processes to the actuators. In case of the behavior structure, the control problem is broken into behaviors without any central intelligent agent present. Through interaction between the competing constituents, emergent behaviors result. Each behavior is nothing but a compound module carrying out the main control computational function. These systems require action arbitration mechanisms, as different actions arise from different behaviors.

Compared with the previous research focus on the management of information among agents, the recent developments centers around the behavior management in an intelligent system. Hybrid control structures [22], a combination of hierarchical and behavior structures, can take the strengths from either of the structures and can get away with the drawbacks associated with both.

2.3.3 Multi-agent cooperation

Cooperation among agents can be explicit or implicit. In case of the explicit information exchange, the agents perform actions to benefit other agents. However, in the implicit case, agents carry on with their own goal-seeking process and these actions will be beneficial to others. Information exchange through communication is an effective way for interaction among the agents. Through communication an agent can get a global view of the problem at hand, helping to take appropriate local decisions with a global view point. Direct communication between agents (a sender and one or more receivers) is purposeful and is very much needed to carry on with explicitly cooperative activities. Observation of the behaviors of other agents and their effects on the environment, fall in the indirect communication scheme.

Based on the agents' structures and communication strategy adopted, MASs can be classified into homogeneous-noncommunicating, heterogeneous-noncommunicating and heterogeneous-communicating categories [29]. Different issues arise in each of these categories and a detailed discussion on the same is not intended in this article. In the homogeneous-noncommunicating category, the associated issues are related to the way agents reason (reactive or deliberative), their perception (global or local), modelling of other's states and their effect on each other. Attitude towards others (benevolent or competitive), agent evolution (stable or evolving), modelling of other's states, resource management (interdependent action), emergence of conventions, and agent roles are some of the issues related to the heterogeneous-noncommunicating category. In case of the heterogeneous-communicating category, agent interaction (understanding each other, language, protocols and coordination), planning communicative acts, attitude towards others (benevolent or competitive), resource management (interdependent action), obligation (commitment or decommitment), etc., arise. Communication among agents opens a multitude of issues in the MAS scenario. Passing wrong information to misguide another agent, decision to stop communication by a single or a group of agents, the degree of sharing information, etc., are issues to be examined in greater detail.

3 Multiple robotic systems

Research interests on multiple autonomous mobile robots has attracted wider attention recently. The coordination of multiple robots is associated with lot of supplementary work in advance. The associated issues are position correction, prevention of communication congestion, local information sensing, in addition to the need for imitating human-like decision making. A robot agent in a multi-agent scenario has to cooperate with other agents and is required to have capabilities to adapt and learn. The evolution of a robot agent is necessary for it to perform certain tasks even under unexpected environmental changes.

An architecture called ALLIANCE is presented for fault tolerance in [4]. Cooperation of heterogeneous mobile robots and a collection of particle-like-small-cell-concept micro-robots, called cellular robots, is reported in [30]. The multiple robotic systems proposed in [31, 32, 33] can have a limited number of agents resulting in narrow fields of application domains.

3.1 Cooperative robotics

Multiple robotic systems deal with research subjects like group architecture, resource conflict, origin of cooperation, learning and geometric problems [34]. Behavior emergence, cooperation, task allocation, herding, etc., are the issues related to multiple robotic systems. Collective behaviors emerge and it is quite interesting to study the impact of an incremental modification to an individual robot on the collective behavior. The collective behavior can be cooperative or competitive in nature. Cooperative behavior needs association with other robots for a mutual benefit and it increases the total utility of the system. In pursuit-evasion and robot soccer systems, competitive behaviors dominate. Selfishness on the part of robots/agents may help the systems to evolve.

Cooperative behavior in robotics (Figure 3) has different fundamental research directions based on the *task* to be accomplished, the *mechanism of cooperation* used and the progress in *system performance* being made. *Task decomposition*, *task allocation*, *learning*, and *rationality* will dominate in studies on joint collaborative behaviors of agents working towards a goal with common interests or rewards. The *mechanism of cooperation* through interaction based on

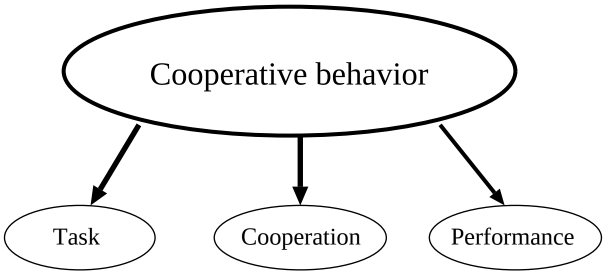

Figure 3: Cooperative behavior

communication relates to the requirements for information or other resources, and to the issues of correctness and fault-tolerance. The *system performance* is concerned with the quantified measures of cooperation.

3.1.1 Group architecture and resource conflict

The group architecture, the basic infrastructure of multiple cooperative robotic systems, can be a centralized, decentralized or a hybrid architecture in nature. Robot heterogeneity/homogeneity, modelling and recognition of other robots, and communication structure are some of the research directions associated with.

The *centralized* approach uses a single control agent. In a *decentralized* approach such a single agent does not exist. Decentralized architectures can be either *distributed* or *hierarchical* in type. From the control point of view, the agents are equivalent in a *distributed* architecture. The *hierarchical* architectures are locally centralized and are commonly used. The control problem in a hierarchical system is divided along functional lines into progressive 'levels of abstraction' of data. The associated computational functions are used as the basis for system decomposition. The *decentralized* architectures are fault tolerant, reliable, and scalable, deriving advantages of the inherent parallelism [35, 36].

Behavior based architectures break the control problem into behaviors being exhibited and they lack a central intelligent reasoning agent. The system may manifest emergent behaviors as a result of interaction among different competing elements. As different actions arise from various behaviors, these systems require action arbitration mechanisms. The relative strengths and weaknesses of the hierarchical and behavior based architectures are available in [28].

In a homogeneous group, the capabilities of individual robots will be identical. However, distinct roles are assigned at the design time or can arise dynamically at run time. Heterogeneous robot groups are complex and task allocation among robots becomes more complicated. The need for modelling other robot agents is significant in a heterogeneous system. Modelling of *intentions, beliefs, actions, capabilities* and *states* of other robots/agents enhance effective cooperation among them. Communication requirements will be less, as each robot can model other agents. However, robots need to communicate when the risk level exceeds a threshold.

Modelling of other robot agents is not required in a system with communicating agents. The communication structure determines the modes of interaction among agents [34]. The different approaches are, interaction via environment, interaction via sensing and interaction via communication. *Interaction via environment* is simple and cooperation is achieved without communication among agents. The objects in the environment function as a communication channel shared by others. In *interaction via sensing*, the agents interact locally by sensing one another without communication. Interaction via sensing is indispensable in modelling other agents. The ability for kin recognition and distinguishing other agents is required in such a system. The explicit communication with other agents, either directly or by broadcasting, requires architectures similar to communication networks.

Resource conflict arises when a single indivisible resource is requested by multiple robots. Conflicts occur, when it is needed to share *space*, *manipulative objects* or *communicative media*. Sharing of space demands for multiple robot path planning and collision avoidance. In real world applications of multi-robot systems, the path planning in advance is impossible. In robot soccer games, the ball (manipulative object) is sought after by competing robots. Through time division multiplexing or by broadcasting over an RF channel, the communication media can be effectively utilized. A design of conflict resolution architectures has to choose a proper balance between centralized and decentralized authority. A conflict resolution architecture in which conflicts are resolved locally by inter-agent coordination is proposed in [38]. An action selection mechanism designed using multi-layer perceptron is used as a high level controller and used in one-on-one robot soccer games in [3].

The interaction of mobile robots in real world situations result in geometric problems and will require multiple robot path planning or assigning of dynamically varying priorities to the robots based on current states.

3.1.2 Learning and evolution

In multiple robotic systems, it is difficult to get the correct values of control parameters leading to desired cooperative behavior. Instead, systems capable of learning to optimize their performance and to adapt to the changes in environment are needed. Learning refers to the changes that are internal to an agent resulting in improved performance through its life time. Learning in robotics generally refers to sensing the changes in the environment and planning accordingly to achieve a goal. Learning helps robots to react effectively to unpredictable external inputs. *Unsupervised learning* is a trial and error approach for exploring the environment to collect data. The evaluation of the data thus collected helps the robot to learn a mapping from its sensor input values to its output effector actions [39].

The method of *experience replay* accelerates learning by temporarily memorizing each trajectory in robot navigation from start to goal state, and after reaching the goal replaying it in reverse order [40, 42]. Reinforcement learning involves the use of feedback to reason about the quality of the robot's condition-action rules. Reinforcement learning enables agents to learn composite foraging behavior using higher level abstractions (conditions, behaviors, heterogeneous reward functions and progress estimators) [41]. Q-learning is a form of model-free reinforcement learning and helps to solve incomplete information Markovian action problems based on the action-value function Q, that maps state-action pairs to expected rewards [42]. Modular Q-learning [43] can produce sophisticated cooperative behavior even if agents do not communicate explicitly. Observation of other agents can facilitate cooperative behavior among agents. Region-based reward assignment is proposed in [44] to solve structural credit assignment problem dealing with continuous actions and states.

The use of memory based learning to allow a player to learn when to shoot and when to pass the ball in a dynamical soccer simulator platform is discussed in [45].

Computational intelligences such as fuzzy system, neural networks, evolutionary computation and some machine learning algorithms like Q-learning, profit sharing plan, could be used for robot intelligence.

3.2 Evolutionary robotics

The research on multi agent/robotic systems relate to the mechanism of coordination, negotiation, cooperation and competition among agents/robots. Efforts are now being made to allow them to emerge adaptively through learning and interaction. Apart from the traditional focus on evolution of robot controllers [46], research is underway for the evolution of evolvable hardware structures [47]. A set of basic interactions which can be used to structure and simplify the process of both designing and analyzing emergent group behavior is proposed in [41].

Automatic development of cooperative strategies for teams of distributed autonomous robots, or software agents is presented and the concept of *tropism-based control architecture* which allows the evolution of cooperative strategies is proposed in [48]. A real time vector field based path planning for attack mode robot and a Petri-net state diagram approach for robot's role selection for robot soccer games are proposed in [49].

The global structural changes and structural emergence help systems to evolve. Structured intelligent robotic systems [50] generate structured intelligence by binding perceptual information and its motion at various levels. In structured intelligent robotic systems, robotic intelligence is based on hardware and software structures. Structured knowledge/skill acquisition helps to develop structured intelligence, and action generation at various levels is done through environmental interaction. Through external and internal evaluations (high level and low level plannings) structured intelligent robots learn and evolve.

Emotions allow agents to manage commitments between each other and it may be used to provide a form of credit assignment in reinforcement learning [51]. Possibilities of incorporating motivations and emotions into animats needs further research.

3.3 Fuzzy logic in autonomous robotics

Autonomous navigation in real-world situation requires facing many uncertainties which are inherent in environment. Research in autonomous mobile robotics is aimed at building physical systems that can move purposefully without human intervention in unmodified environments. The features of fuzzy logic are useful in autonomous navigation of robots. The associated research issues are: how to design robust behavior producing modules, how to coordinate the activity of several such modules, how to use data from the sensors, and how to integrate high level reasoning and low level execution. A detailed review on fuzzy logic in autonomous robot navigation is available in [52, 53].

A reinforcement based evolutionary learning of fuzzy rules is used to coordinate the predefined basic behaviors in autonomous agents in [54]. Simple behaviors can be used to obtain complex actions in autonomous agents. Fuzzy tools help to form logical expressions and can coordinate the activation of basic behaviors. By adjusting the relative weights of each of the behaviors, a robot agent can be made to respond as per its sensor inputs. Fuzzy logic can be used to improve reinforcement learning without loosing the precision of data available from sensors and actuators, see [55].

4 Robot soccer systems

The development of a multi-agent system amounts to searching for a method that will implement an intelligent system composed of multiple agents, cooperating with each other, with independent motion control. Multi-agent robotic systems are more flexible and fault tolerant as several simple robot agents are easier to handle and cheaper to build compared to a single powerful robot for different tasks.

MiroSot [7] is a suitable test bench for multi-agent system research. The soccer game is different from other multi-agent systems, in that the robots of one team have to cooperate, while facing competition from the opponent team. The multi-agent control algorithm, in such an active environment, must comprise of low level kinematics and dynamics, and high level strategies to avoid obstacles and to compete with opponent robots. Such an environment needs fast processing algorithms and suitable robot structure. The hope of course, is that by discovering how to get a robot to move with agility, see with acuity, and think perceptively in the limited context of a soccer game, it will be possible to use the same techniques to build robots to carry out other more useful tasks [56, 57].

From the standpoint of multi-agent systems, the soccer game is a good example of the problems in real world, which can be moderately abstracted. Soccer is one of the standard problems for the study on multi-agent systems, which can be used as a benchmark.

Multiple robot systems show better fault tolerance and are easier, cheaper and flexible. The collective behavior of different agents deals with cooperative and/or competitive behaviors. The robots have to cooperate with others to achieve specific objectives. Task allocation and decomposition are associated with collective behavior. It also requires a form of communication for cooperation among different robots. Learning can comprise of knowledge and skill acquisitions. Improving the relationship with environment through interaction, helps agents to adapt.

Recently the area of entertainment robotics is receiving wider attention as can be seen through many robotic games among research communities [8, 7, 58, 59, 60, 61, 62, 63]. Robotic soccer is being used as a test bed for multiple autonomous robotic research. Robot soccer deals with multiple robots collaborating in adversarial environments with specific objectives.

A reactive robot architecture called *reactive deliberation* is proposed in [31] for a one-on-one game played with simple robots. Ball-pass behavior, which is important for cooperation between soccer players, can be learned using a neural network through computer simulation [64]. Distributed control architecture for cooperative robot soccer system can be seen in [3, 9, 65]. Related research results on robot soccer is available from the workshop proceedings of MiroSot'96 and MiroSot'97, held in KAIST, Taejon, Korea during November 9–12, 1996 and June 1–5, 1997 [7].

The real-time decision making problem and action selection mechanism (*ASM*) for each agent given its role such as striker and goal-keeper in a robot soccer game can be achieved using a multilayer perceptron (MLP) to learn human judgment for the action selection. Fouls may be

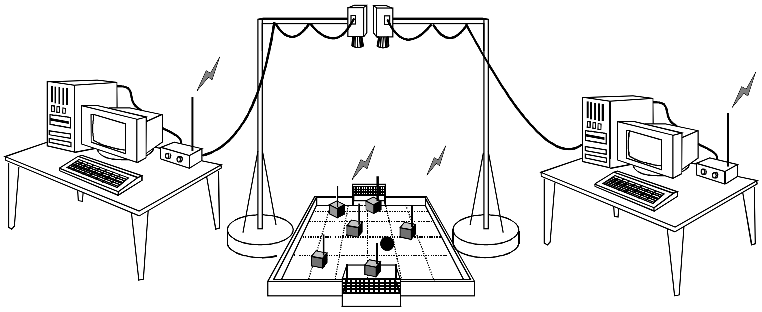

Figure 4: Overview of the soccer robot system with the vision system, host computer and RF communication system.

committed in robot soccer games as in a real soccer. When a penalty-kick, freekick, freeball or a goalkick is called, the game is continued with robots placed at predefined positions as per rules [37]. In the case of a fouls, special strategies can be used, such as 'set-position play' in a real soccer game.

4.1 Classification of robot soccer systems

In addition to the varieties in hardware like CPU, actuators, sensors and so forth, a soccer robot system also differs in control algorithms, strategies and total integration system. The selection of what kind of hardware and software among various methods, is a challenging problem to robot designers. Basically, robots, a vision system, a host computer and a communication system are needed for a robot soccer game (Figure 4). Two operating methods are available for soccer games: *a vision-based soccer robot system* and *a robot-based soccer robot system* [9].

Two implementations of the vision-based system exist. The first one the *remote-brainless soccer robot system*, and the second is the *brain on-board soccer robot system*.

In the *remote-brainless soccer robot system*, each robot has its own driving mechanism, communication part and CPU board. The computational part controls the robot's velocity according to the command data received from a host computer. All calculations on vision data processing, strategies, position control of robots, are done in the host computer which controls the robots like a radio controlled car. The robots can be implemented easily as its structure is simpler than in other systems. Since it has no sensors (encoders), the calculating power of the host computer has to be higher compared to other systems to control the positions of robots accurately. Generally, the position control of mobile robots is done by an embedded processor using its encoders or sensors, but it has to be carried out by the host computer in this case. To control the robots' positions accurately, the sampling time of the host computer must be very small. Hence the vision processing is a key technology for this particular system. As all algorithms of remote-brainless soccer robot system are centralized in the host computer, the communication protocol for multi-robot cooperative system is quite simple. But the problem is that the burden on the host computer multiplies greatly with the increase in number of agents.

In the *brain on-board soccer robot system*, the robots should have functions such as velocity control, position control, obstacle avoidance and so on. The host computer processes vision data and calculates future behaviors of robots according to strategies and sends commands to the robots using an RF modem. The robots make their moves according to these commands keeping away from obstacles autonomously. The robots have to have sensors for position control (encoders) and obstacle avoidance (IR sensors). The calculation load of the host computer is much less compared with the former system.

In the *robot-based soccer robot system*, each robot will have many functions for autonomous behavior. All calculations are done locally in the robots. The host computer processes vision data on the positions of the ball and robots, and forward the same to the robots. Each of the robots decides its own behavior autonomously using the received vision data, their own sensor data and strategies. This can be considered as a distributed control system, where each of the robots has its own intelligence. It may be very hard to implement the robots with established velocity control, position control, obstacle avoidance, communication, decision making, etc. The host computer processes only vision data and can be regarded as a kind of sensor.

4.2 Architecture of soccer micro-robots

The design philosophy is to realize autonomous micro-robots with various functions so that they can be used not only in soccer games, but in a variety of other applications as well. A soccer robot used in MiroSot [7] mainly consists of PCB boards with surface mounting devices, a body frame for safety purposes, actuators for motion and batteries for power. CMOS-type devices can be used considering power consumption. The CPU, side IR sensors, contact sensors and the moving mechanism can make it possible to control robot motion with obstacle avoidance and other intelligent behaviors.

To realize high-intelligence, a high-performance CPU is needed. Considering the limitation on size as per the MiroSot rules and power consumption, a one-chip controller with versatile functions will be suitable. The Intel 16 bit processor 80C196KC (20MHz) is a good candidate along with motion controllers (LM629) for fast control processing.

There are several methods available to determine positions of the robots. An indirect but useful method is to use a vision camera. Each of the robots can have on-board encoders making it possible to calculate its current position. IR sensors on the sides are helpful for obstacle detection. Unlike the conventional method of measuring the amount of light [66], the modulated carrier signal can be hired to make it insensitive to environment light conditions.

Torque, speed and power consumption characteristics are important in choosing the motors. Among various methods of locating motors on board, it is popular to fix the two motors with their axes aligned, with casters in the front and rear sides. If this is difficult due to lack of space, two motors with right-angled wheels [66] or with parallel wheels but not aligned [67] can be used. For smooth dribbling, aligning the wheels or adding a third motor for steering is recommended. Omnidirectional robots will be more agile and are best suited for soccer games [68].

The robots must carry a power source and re-chargeable batteries. The power specifications, the length of competing time, the size and weight are the key factors in choosing the batteries. According to MiroSot rules, robots must not be wired, so IR-remote control or RF-digital communication system is necessary. In an IR-remote control method, the circuitry is relatively simple when the number of channels is less, but as this number increases it becomes considerably complex. The robot can use full duplex RF-digital communication with FM modulation. The basic information transmitted to robots from the host computer are ball position, other robots' positions, which robot has the ball, etc. The communication between robots will help to facilitate in fulfilling team objectives in a better way.

The robots marked with team colors are visually tracked in MiroSot and the design of an appropriate color pattern is a research direction of its own. For object detection using a vision system, an edge detection or a gray level detection method can be adopted. But, if there are objects with similar shapes or brightness of the object being detected, it becomes difficult. In which case, the color information of the object can make detection easier. Specifically, if the target object has a certain uniform color, its information on position and direction under the given environment can be calculated by the color information through the vision system. Based on this method, the directions and positions of the ball and robots with unique color pattern can be calculated.

4.3 Typical MiroSot soccer robot systems

Three typical micro-robot soccer teams, SOTY (1996), MIRO (1997) and MIRAGE (1997), are discussed here. The MIRO team's design is based on the remote brainless system [49]. All calculations are done in the host computer and the motor velocity commands are communicated to the robots. The SOTY team follows the brain on-board (vision based) soccer scheme in its design [3]. The position information of the ball and robots are communicated to the SOTY robots, and robots decide their own actions. The MIRAGE team is quite on the lines of the robot based soccer scheme [69]. Sensors on the MIRAGE robots detect obstacles and the ball.

4.3.1 The SOTY team

The basic behaviors of the SOTY team were *move*, and *obstacle avoidance* and its basic actions being *shoot*, *position_to_shoot*, *intercept_ball*, *sweep_ball* and *block*. A central controller carries out practical strategies including the above basic behaviors and actions, by selecting an action mode based on the current states, locations of the ball and robots. The robots also determine proper tactics according to their specialized intelligence. The lowest level controller in the control structure is a velocity controller with the shortest sampling time. As it goes to the higher level, the sampling time is longer.

The robots have their own roles as striker, defender and goal-keeper with associated areas of action according to their roles. The concept of modified zone defense is used. A robot in its own area as per its role of action, is given a higher priority for action selection and other robots are allowed to move to any place in the playground.

The move behavior consists of two operations performed sequentially, rotating and running. When the operator or host computer informs the robot of the coordinates of its destination, the robot rotates and runs toward that point. A method to align the destination point with the robot orientation is to calculate the inner product between robot orientation vector and the destination vector (from robot position to destination) and to determine whether this value is within the desired boundary or not. In this behavior, the initial misalignment may lead to severe deviation at the final point. But from experimental results the final deviation is found to be less than $10(cm)$ in the field of MiroSot, which is $130(cm) \times 90(cm)$. Keeping both wheels at same speed is preferable, as different speeds may cause unsmooth motion in caterpillar mechanism.

Sensors detect obstacles in the front, back, right-hand or left-hand sides. When a robot finds an obstacle in its front, to avoid a collision it turns left and moves $10(cm)$ forward, then turns to the right and moves $10(cm)$ forward. It can then turn and move towards the original destination if no more obstacles are detected. However, if the robot detects obstacles in front and on left-hand side it can turn right first and then move towards original destination in a similar way. When an obstacle is in the dead angle of a robot, it can turn to the destination and then may find the obstacle again. In this case, it can repeat the above procedures.

The basic actions are on an higher level than the basic behaviors. Shoot and position_to_shoot actions are used for shooting the ball to the goal or for passing it to other robots. Given the ball and goal positions, two relative positions are calculated. The first is for shoot action and the other is for position_to_shoot action. The shoot action is done when the following two conditions are met: 1) the ball is located between the robot and goal, and 2) a straight line from the robot to the ball passes through the goal area. In particular, to prevent robots from kicking the ball towards the home side, it is designed with position_to_shoot action as in Case (2).

In the sweep_ball action, the robot kicks the ball towards the opponent's area when the ball is located on the home side. The block action intercepts the ball or keeps the ball from opponent robots. The robot accomplishes this behavior by moving its position to the expected attack point, considering the ball and self goal positions. These two actions are mainly used by defense robots.

4.3.2 The MIRO team

The MIRO team (MiroSot'97) used a petri-net based model for robots and a vector field based control for robot soccer experiments. The defense and goal keeper robots are controlled using position and angle fields. The meaning of position field is basically the same as that of the potential field. The robots can move to the desired goal point but the final heading direction can't be guaranteed. When orientation of a robot is not important in its final position, this field can be used. On the other hand, angle field is used to control only the angle without translation. At that time, the translation velocity of robot must be zero. The movements of the defender and goal keeper robots are made based on these two vector fields. As the ball moves, the vector field gets modified. The robot can move behind, and kick/dribble the ball towards the goal.

Obstacle avoidance behavior is obtained by modifying the vector field [70]. When the vector at a position is oriented towards an obstacle, it is reformed to orient to the boundary of the obstacle. It is assumed that the size of an obstacle is larger than its real size and assumed to have a width of that of the robot.

A flexible border region which moves according to the ball position is considered with its flexible center line. The supervisory controller receives the information of the robots and ball, and decides the roles of each robots. In other words, the supervisor does not fix the attacking and defending robots beforehand, but their roles are decided and allocated according to the situation.

The robot with a role to defend, should block the ball before the opponent robots can kick the ball. So, it is reasonable that the defense robot must move to the right side of the ball as soon as possible. The goal keeper robot is assumed to move only within the goal area and its major objective is to block the ball effectively.

The attack controller uses the vector field method to implement a fast attacking strategy. If the goal position of the vector field is that of ball position and the direction is towards the opposite goal, the home robot can kick the ball towards it. It is possible to control very fast movement using this method.

4.3.3 The MIRAGE team

The robot based soccer system is followed in the design of the MIRAGE team. The team uses a design strategy wherein the roles of the micro-robots and the host computer are divided in an efficient manner. Each robot is equipped with abilities to recognize itself, other robots and the ball with the help of various sensors. The robots can gather needed information from the host computer by means of wireless communication.

All calculations related to vision system are processed in the host computer and the necessary information is communicated to the robots through wireless communication. The vision system works at a rate of 30 Hz and the RF module operates at 418 MHz and 433 MHz. A 80C196KC (20MHz) Intel CPU is used as the robot brain.

The MIRAGE team uses sensors to detect obstacles and the ball. The vision system is used to give more accuracy in the position information of the ball. Two kinds of sensors are used for detecting obstacles. The IR sensors obtain the position information of the obstacles and of the ball. Tactile sensors are used to avoid mechanical collisions. Four pairs of IR sensors in the front and one pair at the back obtain the necessary information. One pair in the front is exclusively used for locating the ball. Band pass filters are used to prevent any malfunctioning of the IR sensors due to interferences. The two tactile sensors made from mechanical switches are located in the front on both sides.

The control structure implemented by MIRAGE team can be thought of as a kind of single feedback system. The MIRAGE team design follows the lines of a robot-based soccer system. The micro-controller has to do many processes for cooperative behavior, robots' position calculations and for velocity control. Using only the encoder information for velocity control is a difficult job for a 16 bit controller. The control structure of MIRAGE team uses information from a vision system instead of the encoder information as a feedback signal for motion control. The sampling time of system is 0.33 msec and it is slow to control the robot motion. A gain scheduler is introduced for solving this problem. The gain controller assigns appropriate gains for the control law according to the moving patterns. These are obtained from experimental results based on human decisions. Optimal paths for the robots can be obtained using the gain scheduler. If gains are selected appropriately, a smooth motion trajectory can be generated by the proposed control structure.

It is assumed that each robot has its own role such as striker, defender or goal-keeper. It has also its own area of action according to its role. The concept of modified zone defense with priority of action selection discussed in Section 4.3.1 is used. All information available to the robots is processed and a quick decision taken for winning the game. A vector computation method aided by the neural network technique and automata theory is used to make the computations fast. Each robot has its own behavior pattern pre-assigned with the help of neural networks and automata principles. This decides the individual skills/strategies of the robots.

The numerous possibilities of states that can arise during the game are grouped into some finite sets of rules which in turn are referred and used as per the situation.

5 Conclusion

This paper looks at the multi-agent system from the robot soccer point of view. Different types of agents are discussed with related research issues. Learning helps to solve many of the difficulties otherwise associated with multi-agent systems. Different structures and agent cooperation are covered with related research directions.

Cooperative robotics and related issues of group architecture, resource conflict and geometric problems are discussed. Learning and evolution in multiple robotic systems allow the robots to emerge adaptively through interaction. Fuzzy logic has lot of potential to deal with the uncertainties inherent within the environment the robots work. Incorporation of motivations and emotions into animats needs further research.

Robot soccer is used as a test bench for multi-agent and multiple robotic systems research. Different operating methods of robot soccer systems are outlined with representative robot soccer teams. By discovering how to make a robot move with agility, see with acuity, and think perceptively in the limited context of a soccer game, it will be possible to use the same techniques to build robots to carry out other more useful tasks for the benefit of the mankind.

References

[1] http://drogo.cselt.stet.it/fipa

[2] Kim, J.-H., Shim, H.-S., Jung, M.-J., Kim, H.-S., and Prahlad, V. "Cooperative Multi-Agent Robotic Systems: From the Robot-soccer Perspective." *Proc. of MiroSot'97 Workshop*, (June 1997), pp 3–14.

[3] Shim, H.-S., Kim, H.-S., Jung, M.-J., Choi, I.-H., Kim, J.-H., and Kim, J.-O. "Designing distributed control architecture for cooperative multi-agent system and its real-time application to soccer robot." *J. of Robotics and Autonomous Systems*, Elsevier, Vol. 21, (September 1997), pp 149-165.

[4] Parker, L. E. "Heterogeneous Multi-robot cooperation." Ph.D. thesis, MIT, AI Lab., Cambridge, MA, February 1994.

[5] Tyrrell, T. "Computational Mechanisms for Action Selection." Ph.D. thesis, The University of Edinburgh, 1993.

[6] Tyrrell, T. "The use of hierarchies for action selection." *From Animals to Animats 2*, The MIT Press, 1993, pp 138–147.

[7] http://www.mirosot.org/

[8] http://www.fira.net/

[9] Kim, J.-H., Shim, H.-S., Kim, H.-S., Jung, M.-J., Choi, I.-H., and Kim, J.-O. "A cooperative multi-agent system and its real time application to robot soccer." *Proc. IEEE Int. Conf. on Robotics and Automation*, Albuquerque, New Mexico, Vol. 1, (April 1997), pp 638-643.

[10] Russell, S. J.and Norvig, P. "Artificial Intelligence - A Modern Approach (AIMA)." Englewood Cliffs, NJ, Prentice Hall, 1995.

[11] Jennings, N. R. and Wooldridge, M. "Software Agents." *IEE Review* 42(1), (January 1996), pp 17-21.

[12] Brooks, R. A. "Intelligence without representation." *Artificial Intelligence*, 47(1), (1991), pp 139-159.

[13] C. J. C. H. Watkins. "Learning from delayed rewards." Ph.D. thesis, Phycology Department, Cambridge University, 1989.

[14] http://www.msci.memphis.edu/ franklin/AgentProg.html

[15] Keil, F.C. "Concepts, Kinds and Cognitive Development." Cambridge, MA, MIT Press, 1989.

[16] http://mpfwww.jpl.nasa.gov/

[17] Wooldridge, M., and Jennings, N. R. "Agent Theories, Architectures, and Languages: A Survey." *Proc. ECAI-Workshop on Agent Theories, Architectures and Languages* Amsterdam, The Netherlands, (1994), pp 1-32.

[18] Wooldridge, M., and Jennings, N. R. "Intelligent Agents: Theory and Practice." *The Knowledge Engineering Review*, 1995.

[19] Wooldridge, M., and Jennings, N. R. "Towards a Theory of Cooperative Problem Solving." *Proc. Modelling Autonomous Agents in a Multi-Agent World (MAAMAW-94)*, Odense, Denmark, (1994), pp 15-26.

[20] Shoham, Y. "Agent-Oriented Programming." *Artificial Intelligence* 60(1), (1993), pp 51-92.

[21] Davidson, P. "Autonomous Agents and the concept of concepts." Ph.D. thesis, Dept. of Computer Science, Lund University, Sweden, 1996.

[22] Shim, H.-S., Jung, M.-J., Kim, H.-S., Kim, J.-H., and Prahlad V. "A hybrid control structure for vision based soccer robot system." *Int. J. of Intelligent Automation and Soft Computing*, (to appear in this issue).

[23] Gasser L. "Social conceptions of knowledge and action." *Artificial Intelligence*, Vol. 47, No. 1, (1991), pp 107-138.

[24] M.V. Nagendra Prasad. "Learning Situation Specific Control in Multi-Agent Systems." Ph.D. thesis, Dept. of Computer Science, Uni. of Massachusetts Amherst, May 1997.

[25] Barto, A., Sutton, R., and Watkins, C. "Learning and sequential decision making." *Learning and Computational Neuroscience (ed.)*, Cambridge, MA, MIT Press, 1990.

[26] Sutton, R. "Learning to predict by the methods of temporal differences." *Machine Learning*, Vol. 3, (1988), pp 9-44.

[27] Holland, J.H., "Properties of bucket brigade algorithm." *First Int. Conf. on Genetic ALgorithms and their Applications*, Pittsburgh, PA, (1985), pp 1-7.

[28] Fayek, R. E., et al. "A System Architecture for a Mobile Robot Based on Activities and a Blackboard Control Unit." *IEEE Proc. Int. Conf. Roboics and Auto.*, U.S.A, (1993), pp 267-274.

[29] Stone, P., and Veloso, M. "Multiagent Systems: A Survey from a Machine Learning Perspective." *CMU CS technical report*, No. CMU-CS-97-193.

[30] Kawauchi, Y., Inaba, M., and Fukuda, T. "A Principle of Distributed Decision Making of Cellular Robotic System (CEBOT)." *IEEE Proc. Int. Conf. Roboics and Auto.*, Vol.3, U.S.A., (1993), pp 833-838.

[31] Sahota, M. K. "Real-time intelligent behaviour in dynamic environments : Soccer-playing robots." Master's thesis, The University of British Columbia, August 1994.

[32] Kube, C. R., and Zhang, H. "The Use of Perceptual Cues in Multi-Robot Box-Pushing." *IEEE Proc. Int. Conf. Robotics and Automation*, (1996),pp 2085-2090.

[33] Brumitt, B. L., and Stentz, A. "Dynamic Mission of Planning for Multiple Mobile Robots." *IEEE. Proc. of Conf. Robotics and Automation*, (1996), pp 2396-2401.

[34] Cao, Y. U., Fukunaga, A. S., and Kahng, A. B. "Cooperative Mobile Robotics: Antecedents and Directions." *Proc. of the IEEE/RSJ IROS Conf.*, 1995.

[35] Beckers, R., Holland, O. E., and Deneubourg, J. L. "From local actions to global tasks: Stigmery and collective robotics." *Proc. of Artificial Life*, Vol. IV, MIT Press, (1994), pp 181-189.

[36] Arkin, R. C. "Cooperation without communication: Multiagent schema-based robot navigation." *J. of Robotic Systems*, Vol. 9, No. 3, (1992), pp 351-364.

[37] Kim, J.-H. "A Booklet on MiroSot'96 (Micro-Robot World Cup Soccer Tournament)." Published by MiroSot Organizing Committee, April 1996, accessible from http://www.mirosot.org/

[38] Pappas, G. J., Tomlin, C., and Sastry, S. “Conflict Resolution for Multi-agent Hybrid Systems.” *Proc. of 35th Conf. on Decision and Control*, (1996), pp 1184-1189.

[39] Chapman, K. L., and Bay, J. S. “Task decomposition and dynamic policy merging in the distributed Q-learning classifier system.” *Proc. IEEE Int. Symposium on Computational Intelligence in Robotics and Automation*, CIRA’97, (1997), pp 166-171.

[40] Lin, L.-J. “Programming robots using reinforcement learning and teaching.” *Proc. of AAAI-91*, AAAI Press/MIT press, Menlo Park, CA, July 1991.

[41] Mataric, M. J. “Designing Emergent Behaviors: From Local Interactions to Collective Intelligence.” in *From animals to animats 2*, The MIT Press, (1993), pp 432–441.

[42] Watkins, C. J. C. H., and Dayan, P. “Q-learning.” *Machine Learning*, Vol. 8, (1992), pp 279-292.

[43] Ono, N., and Fukumoto, K. “Multi-agent reinforcement learning: A modular approach.” *Proc. of the 2nd Int. Conf. on Multi-Agent Systems*, AAAI Press, (1996), pp 252-258.

[44] Suh, I. H., Kim, J.-H., and Oh, S. R. “Region-based Q-learning for Intelligent Robot Systems” *Procc. IEEE Int. Symposium on Computational Intelligence in Robotics and Automation*, CIRA’97, (1997), pp 172-178.

[45] Stone, P., and Veloso, M. “Beating a defender in robotic soccer: Memory based learning of a continuous function.” *Advances in Neural Information Processing Systems 8*, MIT Press, (1996), pp 896-902.

[46] Lund, H. H., and Hallam, J. “Evolving Sufficient Robot Controllers.” *Proc. IEEE Int. Conf. on Evolutionary Computation*, ICEC’97, (1997), pp 495-499.

[47] Lund, H. H., Hallam, J., and Lee, W.-P. “Evolving Robot Morphology.” *Proc. of IEEE 4th Int. Conference on Evolutionary Computation*, IEEE Press, (1997), pp 197-202.

[48] Agah, A., and Tanie, K. “Emergent Cooperative Strategies for Robot Team Sports.” (to appear in this issue).

[49] Kim, D.-H., Kim, Y.-J., Kim, K.-C., Kim, J.-H., and Prahlad V. “Vector Field based Path Planning and Petri-net based Role Selection Mechanism for Soccer Robots.” (to appear in this issue).

[50] Fukuda, T., and Kubota, N. "Evolutionary Robotic System." *Proc. of MiroSot'97*, (1997), pp 19-26.

[51] http://netq.rwland.org/isab.texts/sab96rev.html

[52] Saffiotti, A. "The use of fuzzy logic in autonomous robot navigation." *Int. J. of Soft Computing*, Springer-Verlag, Vol. 1, (1997), pp 180-197.

[53] http://iridia.ulb.ac.be/FLAR/papers/

[54] Bonarini, A. "Evolutionary learning of fuzzy rules: competition and cooperation." *Fuzzy Modelling: Paradigms and Practice*, Kluwer Academic Press, Norwell, MA, 1996.

[55] Bonarini, A. "Learning dynamic fuzzy behaviors from easy missions." *Proc. of Information Processing and Management of Uncertainty in Knowledge-Based Systems'96*, (1996), pp 1223-1228.

[56] Casti, J. "A game of three robots." *New Scientist*, (April 26, 1997), pp 28-31.

[57] *The Times*, London, Thursday, September 18, 1997, (http://www.the-times.co.uk).

[58] http://www.robocup.org/

[59] http://www.cc.gatech.edu/grads/b/Tucker.Balch/JavaBots/

[60] http://www.daimi.aau.dk/ hhl/robotDME.html

[61] http://www.ncc.com/misc/rcfaq.html

[62] http://muster.cs.ucla.edu:80/w3r3/

[63] http://www.robotmag.com

[64] Stone, P., and Veloso, M. "Broad learning from narrow training : A case study in robotic soccer." *CMU Technical Report*, November 1995.

[65] Kim, J.-H., Shim, H.-S., Jung, M.-J., Kim, H.-S., Choi, I.-H., and Han, W.-S., "Building a soccer robot system for MiroSot'96." in *Video Proc. IEEE Int. Conf. on Robotics and Automation*, Albuquerque, New Mexico, April 1997.

[66] Mondada, F., Franzi, E., and Ienne, P. "Mobile Robot Miniaturization: A Tool for Investigation in Control Algorithms." *Proc. of the 3rd Int. Symp. on Experimental Robotics*, Japan, Oct. 1993.

[67] Kim, J.-H., Jung, M.-J., Shim, H.-S., and Lee, S.-W., "Autonomous Micro-Robot 'Kity' for Maze Contest." *Proc. of Int. Sym. on Artificial Life and Robotics (AROB)*, Oita, Japan, February 1996.

[68] Blank, R., Koller, J.-M., Lauria, M., and Conti, F. "Omnidirectional robot and fast positioning systems." *Proc. of MiroSot'97 Workshop*, Taejon, Korea, (June 1997), pp 13–14.

[69] Han, K.-H., Seo, B.-I., Lee, T.-H., and Oh, S.-J. "The improvement of MIRAGE I robot system." *Proc. of MiroSot'97 Workshop*, Taejon, Korea, (June 1997), pp 55–58.

[70] Springer, G., Hannah, P., Stonier, R. J., Smith, S., and Wolfs, P. "Simple strategies for collision-avoidance in robot soccer." *Proc. of MiroSot'97 Workshop*, Taejon, Korea, (June 1997), pp 191-196.