AI Magazine Volume 22 Number 1 (2001) (© AAAI)

RoboCup-2000

The Fourth Robotic Soccer World Championships

Edited by Peter Stone

with Minoru Asada (humanoid), Tucker Balch (workshop), Raffaelo D'Andrea (Cornell team), Masahiro Fujita (legged), Bernhard Hengst (UNSW team), Gerhard Kraetzschmar (mid size), Pedro Lima (engineering challenge), Nuno Lau (FC PORTUGAL team), Henrik Lund (RoboCup Jr.), Daniel Polani (scientific challenge), Paul Scerri (simulation), Satoshi Tadokoro (rescue), Thilo Weigel (CS FREIBURG team), and Gordon Wyeth (small size)

■ The Fourth Robotic Soccer World Championships (RoboCup-2000) was held from 27 August to 3 September 2000 at the Melbourne Exhibition Center in Melbourne, Australia. In total, 83 teams, consisting of about 500 people, participated in RoboCup-2000, and about 5000 spectators watched the events. RoboCup-2000 showed dramatic improvement over past years in each of the existing robotic soccer leagues (legged, small size, mid size, and simulation) and introduced RoboCup Jr. competitions and RoboCup Rescue and Humanoid demonstration events. The RoboCup Workshop, held in conjunction with the championships, provided a forum for the exchange of ideas and experiences among the different leagues. This article summarizes the advances seen at RoboCup-2000, including reports from the championship teams and overviews of all the RoboCup events.

RoboCup is an international research initiative that encourages research in the fields of robotics and AI, with a particular focus on developing cooperation between autonomous agents in dynamic multiagent environments. A long-term grand challenge posed by RoboCup is the creation of a team of humanoid robots that can beat the best human soccer team by the year 2050. By concentrating on a small number of related, well-defined problems, many research groups both cooperate and compete with each other in pursuing the grand challenge.

The Fourth Robotic Soccer World Championships (RoboCup-2000) was held from 27 August to 3 September 2000 at the Melbourne Exhibition Center in Melbourne, Australia. In total, 83 teams, consisting of about 500 people, participated in RoboCup-2000. Over 5000 spectators watched the events. RoboCup has been advancing steadily, both in terms of size and technological level since the first international event in 1997 that included 35 teams (Asada et al. 2000; Coradeschi et al. 2000; Noda, Suzuki et al. 1998). Specifically, RoboCup-2000 showed dramatic improvement in each of the existing robotic soccer leagues (legged, small size, mid size, and simulation) and introduced RoboCup Jr. competitions and RoboCup Rescue and Humanoid demonstration events.

In addition to the simulation-based and robotic events, the RoboCup-2000 Workshop provided a forum for the exchange of ideas and experiences among the different leagues. Twenty oral presentations and twenty posters

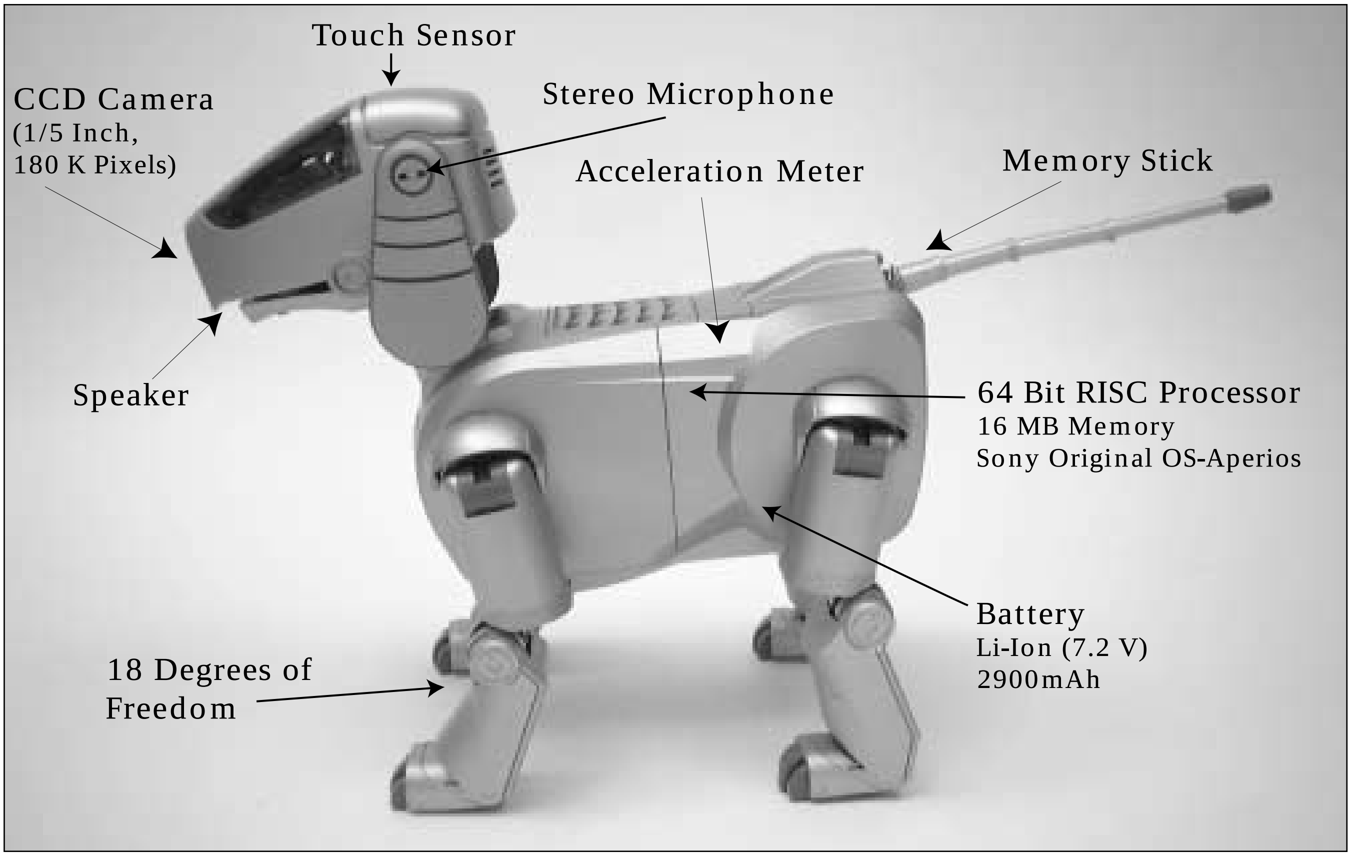


Figure 1. The Legged Robot Platform.
Weight: 1.59 kg. Size: 2.75 x 156 x 266 mm (without tail).

were presented, from which four papers were nominated for the RoboCup scientific and engineering challenge awards. These distinctions are given annually for the RoboCup-related research that shows the most potential to advance their respective fields.

This article summarizes the advances seen at RoboCup-2000. The following four sections describe the four soccer-based competition leagues, including reports from the respective champions, UNSW (legged), Cornell BIG RED (small size), CS FREIBURG (mid size), and FC PORTUGAL (simulation). The next section introduces RoboCup Rescue, a disaster-rescue–based research effort designed to transfer RoboCup-related research to humanitarian goals. RoboCup Jr., the RoboCup education effort aimed at school children, is discussed in the following section. Scheduled to debut as a full league in 2002, the RoboCup humanoid effort held a demonstration in Melbourne, which is described in the third section. The article concludes with overviews of the RoboCup workshop and the two challenge award winners.

The Sony Legged Robot League

Since RoboCup-99, all participants in the Sony legged robot league have been using the quadruped robot platform (Yamamoto and Fujita 2000), which is similar to the commercial entertainment robot AIBO ERS-110 (figure 1). The setup and the rules of the RoboCup-2000 legged competition were based on those of RoboCup-98 (Fujita et al. 2000). Each team has 3 robots, and the size of the field is 1.8 meters x 2.8 meters. Objects such as the ball and goals are painted different colors. In addition, there are 6 poles with different colors at known locations for self-localization. As is the case in human soccer, there are penalties and regulations that govern the play. We introduced two changes from the previous year's rules to keep the game flowing and encourage development of team-play strategies. First, we introduced an *obstruction rule*, which states that a robot that does not see the ball but is blocking other robots is removed from play. Second, we modified the penalty area and applied the *two-defender rule*: If there are two or more defenders in the penalty area, all but one is removed. As a result, the ball became stuck in the corner much less frequently. Moreover, the champion team, UNSW, implemented teammate recognition to avoid obstructing a teammate that was controlling the ball.

Twelve teams from nine countries were selected to participate in the RoboCup-2000 Sony Legged Robot League: (1) Laboratoire de Robotique de Paris (LRP) (France), (2) University of New South Wales (UNSW) (Australia), (3) Carnegie Mellon University (CMU), (4) Osaka University (Japan), (5) Humboldt University (Germany), (6) University of Tokyo (Japan), (7) University of Pennsylvania, (8) McGill University (Canada), (9) SWEDEN UNITED TEAM (Sweden), (10) MELBOURNE UNITED TEAM (Australia), (11) University of Rome (Italy), and (12) University of Essex (United Kingdom). The first nine teams participated in the previous year's competition; the last three teams were new participants.

Championship Competition

For the competition, we divided the 12 teams into 4 groups of 3 teams each. After a round robin within each group, the top two teams in each group proceeded to the final tournament. The 2000 champion was UNSW, followed by LRP in second place and CMU in third place.

One significant improvement this year over past years was ball-controlling technique. In RoboCup-99, the University of Tokyo team introduced the technique of propelling the ball with the robot's head, which can make the ball move a longer distance than an ordinary kicking motion can. In 2000, almost all the teams implemented their own heading motion. Another impressive achievement for controlling the ball was introduced by UNSW. Its robots put the ball between their front legs, turned to change their heading while they controlled the ball and then kicked (pushed) the ball with both legs. This technique is very efficient for shooting the ball a long distance in a target direction.

RoboCup Challenge

In addition to the championship competition, every year we continue to hold the RoboCup Challenge as a technical routine competition. The challenge competition focuses on a particular technology more than the championship competition. In 2000, we had 3 different technical routine challenges: (1) striker, (2) collaboration, and (3) obstacle avoidance.

The *striker challenge* was the simplest. The ball and one robot were placed in randomly selected positions (and orientation) on the field. The robot had to put the ball in the goal as quickly as possible. If it was unable to do so within three minutes, then the distance from the ball to the goal at the end of that period was measured. Note that the initial positions and orientation were selected after all the teams submitted their memory sticks with their developed software.

The *collaboration challenge* was defined to encourage the development of a passing behavior. There were two robots, one of which was put in the defensive half of the field (passer); the other was put in the offensive half (shooter). The passer and the shooter had to stay on their respective halves of the field, and the shooter had to kick the ball into the goal.

The *obstacle-avoidance challenge* was also defined to encourage the development of team strategy as well as the ability to avoid a robot from the opposite team. One robot and the ball were placed on the field as in the striker challenge. In addition, two obstacles—(1) a teammate robot with a red uniform and (2) an opponent robot with a blue uniform—were placed at selected positions. The player had to score a goal without touching the obstacles. In both the collaboration and obstacle-avoidance challenges, the time to score was recorded.

To complete the technical routine challenges, teams had to develop recognition algorithms for other robots, the half line, the ball, and the goals. Localization was also an important technology for the challenges.

In the striker challenge, 6 teams scored goals in an average time of 90 seconds. In the collaboration challenge, 6 goals were scored in an average of 100 seconds. In the obstacle-avoidance challenge, 4 teams scored in an average of 112 seconds. All in all, about half of the participating teams were able to achieve the objectives of the three RoboCup Challenge tasks. UNSW won the challenge competition, Osaka University finished second, and CMU finished in third place.

UNSW: Legged League Champion

UNSW won the RoboCup-2000 Sony Legged Robot League as well as the legged robot challenge event. This section gives an overview of the technical innovations behind its success.

Main Algorithm

UNSW divides the team into two field players and one goalie. The field player robots try to get behind the ball and run at it. The field is divided into regions, and robots behave slightly differently across regions. There are three main skills: (1) dribbling, (2) head butting, and (3) kicking. The skill to be executed depends on the heading of the robot, the heading of the ball relative to the robot, and the region the

Another impressive achievement for controlling the ball was introduced by UNSW. Its robots put the ball between their front legs, turned to change their heading while they controlled the ball, and then kicked (pushed) the ball with both legs.

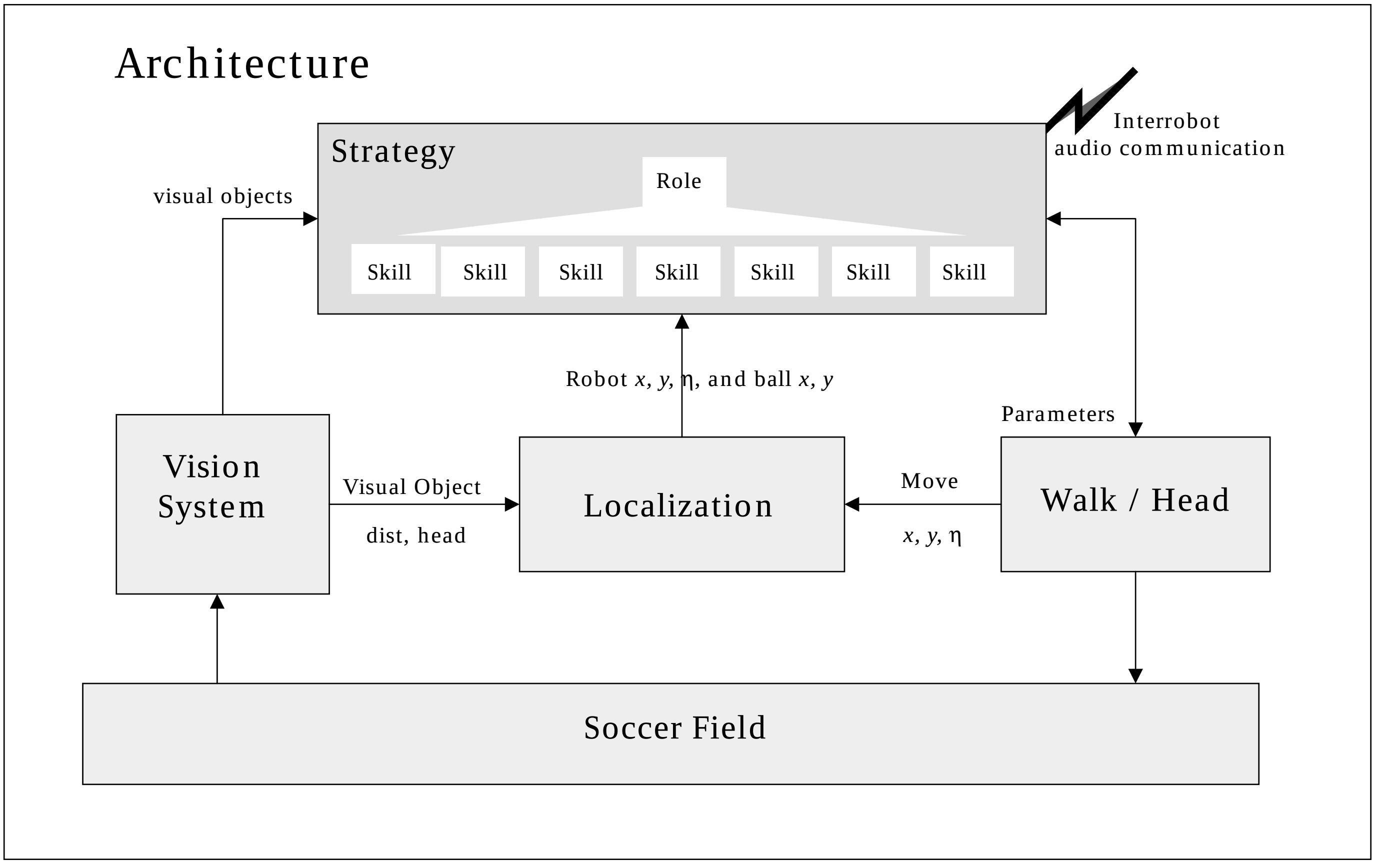


Figure 2. The Software Architecture of UNSW.

robot is in. Near the edge of the field, a slower and more stable walk is adopted so the legs do not get stuck on the edge.

The strategy of the goalie is to stay close to the center of the goal, facing the opposition goal, while it looks for the ball. It localizes itself by looking at the field markers. When it sees the ball, the goalie moves forward to a fixed radial distance from the goal center facing the ball. If the ball comes close enough, the goalie will move forward and attempt to head butt or kick the ball away from its own goal as if it were a field player.

To avoid its own goal, the goalie turns clockwise if it is on the right side of the goal area and counterclockwise if it is on the left side. This skill not only can allow the robot to find the ball but also has the effect of spinning the ball out toward the center of the field.

Regarding team play, UNSW players can recognize robots when they are close using vision and infrared sensors. When a robot sees a teammate, and the ball is not too close, it backs up or walks sideways, depending on the heading of the teammate. When a robot sees an opponent, it takes a more aggressive role: It does not spend much time getting behind the ball. If the robot is not facing its own goal, it runs at the ball in an attempt to take the ball away from the opponent.

UNSW has four main components to its software architecture (figure 2).

First is a vision system that uses color tables to recognize blobs, converting them to objects such as beacons, goals, or the ball. Metrics such as direction and distance are generated at the same time.

Second is a localization routine that updates the position and direction of the robot each time field objects are recognized and each time the robot moves.

Third is a parameterized walking routine that drives the legs and head effectors based on directions from the strategy module.

Fourth is a strategy module that combines various skills or behaviors that have been coded using a hierarchical rule-based format.

Vision System

The images captured by the robot's camera are initially represented as a YUV image. The Y (intensity of a pixel) plane (0–255) is divided into 14 different planes. For each plane, UNSW tries to draw a polygon to fit the training data for each color; so for each plane, there is one bit-map file for ultraviolet (UV) values (color components of a pixel) based on the polygons.

The YUV image is converted into a C-plane. UNSW uses a fast algorithm to form color blobs in the C-plane. Note that they set the color bit-map files in such a way that each pixel gets classified as one color only; that is, polygons do not overlap. Using the recognized blobs, UNSW calculates sizes and centroids to form objects. There are also some sanity checks to throw away spurious data or unwanted objects.

Regarding the color calibration, a color class is defined for each of the beacon, robot marker, goal, and ball colors. Each of the 25 sample images has its 88 x 60 pixels manually classified by color. This somewhat time-consuming exercise, which needs to be repeated every time lighting conditions change, provides the training data (*y* value, *u* value, *v* value, color class) necessary to learn a more general color classification hypothesis.

First, a scatter diagram is drawn for each color from the training data showing the *u*-*v* values for different ranges of *y* values. Instead of restricting the color-class hypothesis space to *u*-*v* rectangles used by some others, nonoverlapping polygons are fitted using an iterative procedure that expands a smaller polygon to include most of the training data for each color and each range of *y* values. A polygon is a much better fit to the typically wedge-shaped color clusters evident in the scatter diagram. If the polygons for the various *y* ranges are stacked on top of each other, a three-dimensional (3D) solid emerges representing each color class in YUV space.

The learned 3D YUV array for the color classes is stored in a table on the memory stick allowing the robot to quickly classify pixels from new images by looking up which color class the pixel belongs to from its YUV value.

Localization

UNSW's localization maintains three variables: (1, 2) *x*,*y* coordinates and (3) heading of the robots. Beacons and goals are fixed. When a robot sees a beacon, it knows the heading of the beacon, and based on the size, it estimates the distance. If it sees two beacons, then it uses a triangular formula to calculate its position. If it sees only one beacon, it adjusts its position based on the heading and distance of the beacon and where it thinks it is on the field.

Locomotion

The locomotion uses a *trot gait* (diagonally opposite legs lifting simultaneously). The paws are driven in a rectangular locus calculated to give the robot a constant velocity over the ground. The orientation and size of the locus of the various legs determines whether the robot moves forward or backward, moves left or right, or turns on the spot. Head movements are driven at the same time but independently from the legs.

Strategies

When the robot is far away from the opponent's goal, the robot does not have to line up the ball and the goal to go for the ball. All it needs to do is knock the ball to the other half. However, when it's near the goal, it takes a different approach. It always tries to line up the goal and the ball and uses the dribbling skill.

The Small-Size Robot League

Small-size robot teams consist of as many as 5 robots that can each fit into an area of 180 centimeters (hence the alternative name Formula 180 or F180). The robots play on a green-carpeted table-tennis–sized field with sloping walls. The rules permit a camera to be perched above the field to be used with an off-field computer for a global vision system. This system is used to track the players, opponents, and the ball. During a game, the robots use wireless communication to receive tracking information from the off-field computer as well as commands or strategic information. No human intervention is allowed except for interpretation of the human referee's whistle.

The F180 games are exciting to watch because these robots can move quickly. The orange golf ball used as the soccer ball is propelled at speeds over three meters a second by ingenious kicking mechanisms. With the precise visual information from the global vision system, the robots themselves can move at speeds over 1 meter a second with smooth control. Nevertheless, robots moving at these speeds can and do have spectacular collisions. Intentional fouls can lead to robots being sent from the field under the shadow of a red card.

The F180 games are exciting to watch because these robots can move quickly. The orange golf ball used as the soccer ball is propelled at speeds over three meters a second by ingenious kicking mechanisms.

The need for speed and control has given the small-size league a reputation as the engineering league. Engineering disciplines including electromechanical design, applied control theory, power electronics, digital electronics, and wireless communications have been the dominating factors in success in this league over recent years. Successful teams have typically demonstrated robot speed and powerful kicking rather than elegant ball control and sophisticated team strategies.

The Competition

Sixteen teams from 9 different nations competed for the small-size champion's trophy. The early rounds of the contest demonstrated the depth of the league, with some quality teams being

Figure 3. The Small-Size–League Final Game.

eliminated during the round-robin section. In particular, the MuCows from Melbourne University, Australia, achieved remarkable performance in its first year in the contest but was unlucky to lose in a high-class group. As well as solid all-around performance, the team from Melbourne showed its engineering skill with a high-bandwidth, low-power communications system that was seemingly immune to the problems experienced by most competitors.

Three small-size teams chose not to use the global vision system; instead, these teams relied on on-board vision capture and processing to sense the environment. These teams demonstrated that it is possible to build vision hardware suitable for real-time processing within the severe size constraints of the F180 league. The ViperRoos from the University of Queensland, Australia, had the distinction of becoming the first local vision team to beat a global vision team; the score was 2–0. However, none of the local vision teams was able to reach the finals.

The eight finalists all had excellent technical merit. TEAM CRIMSON from Korea has a custom video-processing board that extracts the position of the players and the ball at the full NTSC (National Television Systems Committee) video rate of 60 hertz. It does so without ever buffering the video in random-access memory (RAM), so that the position information is delayed by only 1/60th of a second. With such a small delay in vision processing combined with highly responsive robots, TEAM CRIMSON was capable of extremely fast and controlled motion. However, because of communications problems (and some last-minute code changes!), the team was knocked out in the quarter finals.

The French team from the Universite Pierre et Marie Curie was the only team to score against the eventual champions, BIG RED from Cornell University. The French curved-path planning system allowed it to scoop the ball from in front of the opposition and make highly effective attacks on goal. The team was unlucky to be knocked out by Cornell in the quarter final.

The first semifinal match between FU-FIGHTERS from the Freie Universitat of Berlin, Germany, and the RoboRoos from the University of Queensland, Australia, showed a contrast of styles. The RoboRoos, competing for the third consecutive year, relied on smooth control and an adaptive team strategy to reach the finals, whereas the FU-FIGHTERS used fast, aggressive trajectories with an extremely powerful kicker. The FU-FIGHTERS showed clear dominance winning the match 3–0.

The second semifinal match between BIG RED and LUCKY STAR from Ngee Ann Polytechnic in Singapore was the closest match of the small-size tournament. The match was 0–0 at full time, playing through a period of sudden death extra time to come down to a penalty shootout that was decided at 4–3. LUCKY STAR combined novel electromechanical design with excellent control to achieve its result. Its robots had an extremely effective kicking mechanism that was integrated into a narrow body design. The narrow body enabled the robots to slip between defenders to get to the ball, despite the crowding of the field. Its vision and control were sufficiently good that the players would reliably kick the ball despite the small kicking face of the robot. LUCKY STAR won third place in the contest.

BIG RED went on to win the final against FU-FIGHTERS convincingly. Figure 3 is a shot from the final game. This is the second consecutive year that BIG RED has won the small-size championship and the second year that FU-FIGHTERS has come in second. Although it might seem natural to attribute their achievements to novel electromechanical design such as FU-FIGHTER's powerful kicker or BIG RED's dribbling device (described later), it is also apparent that these robots are superbly controlled. As these control issues, along with the other fundamental engineering issues, are addressed on an even scale across the competition, other factors, such as effective team strategies, will come more into play.

Cornell BIG RED: Small-Size Champion

BIG RED repeated as champion of the small-size league at RoboCup-2000. The RoboCup competition is an excellent vehicle for research in the control of complex dynamic systems. From an educational perspective, it is also a great means for exposing students to the systems engineering approach for designing, building, managing, and maintaining complex systems.

In an effort to shift the current emphasis of the competition away from simple strategies to more complicated team-based strategies, the main emphasis of this year's winning team was to play a controlled game. In other words, in a game without ball control, effective strategies essentially consist of overloading the defensive area during a defensive play (the so-called *catenaccio* in human soccer, a strategy that is very effective, even if dull and frustrating for the spectators) and shooting the ball toward open space or the goal area in the opponent's half during offensive plays. This was, in fact, the simple role-based strategy adopted by our championship team in 1999, which was shown to be extremely effective (D'Andrea et al. 2000).

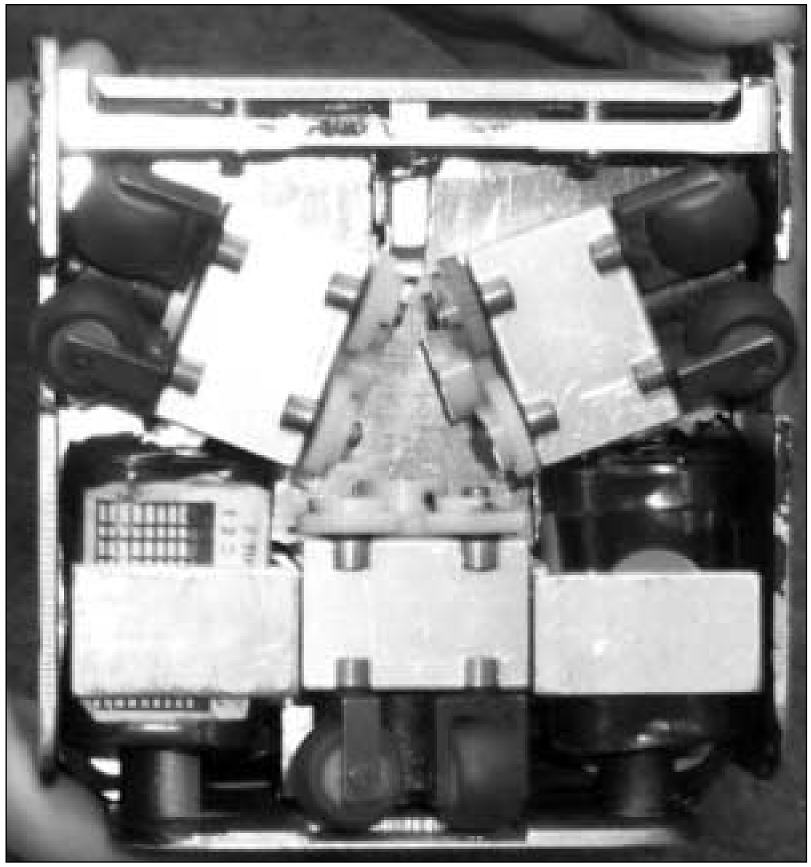


Figure 4. Bottom View, Omnidirectional Drive.

To bring control to the RoboCup competition, the Cornell team developed two electromechanical innovations and the associated control strategies to render them effective: (1) omnidirectional drive and (2) dribbling. Because of space limitations, we restrict our description to these two features, followed by the underlying feedback control strategy that allowed the Cornell team to make full use of them.

Omnidirectional Drive and Dribbling

The Cornell team implemented an effective means of position control this year. This control was achieved by placing three pairs of wheels at locations that are at the vertexes of an imaginary triangle (figure 4). Each pair of wheels has an active degree of freedom and a passive one, the active one being in the direction of the rotation of the motor and the passive one being the one perpendicular to it. Loosely speaking, because the drive directions

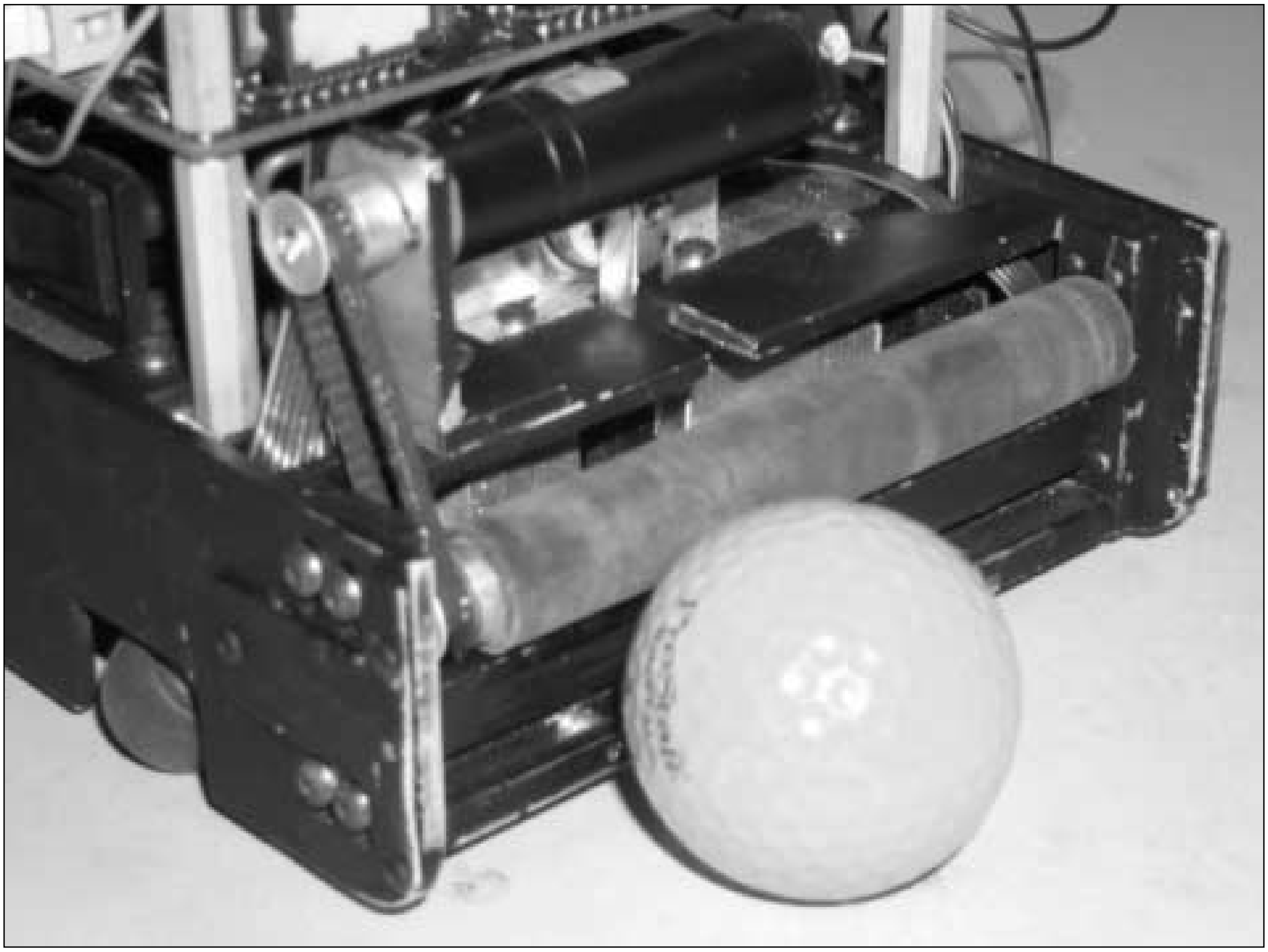


Figure 5. Angled View, Dribbling Mechanism.

are pairwise linearly independent, and the number of degrees of freedom on a two-dimensional surface is three (two translational and one rotational), one can independently control the translation and the rotation of the robot by a judicious choice of drive velocities.

The dribbling mechanism is a rotating bar with a latex cover placed just above the kicking mechanism (figure 5). On contact with the ball, the rotation of the bar imparts a backward spin on the ball; the bar is strategically placed such that the net component of the force on the ball is always toward the robot, which is achieved without violating the 20-percent convexity rule. (The convexity rule states that no more than 20 percent of the ball along any dimension can be within the convex hull of the robot.)

The omnidirectional drive, coupled with the dribbling mechanism, greatly increases the potential capabilities of the robots (we stress the word potential because it is not obvious that a real-time control strategy can be developed to fully use these features). The main capability that is rendered possible by this combination is the effective receiving of passes, which must be central to any sophisticated team-based strategy.

Trajectory Generation and Control

The overall trajectory generation and control scheme for one robot is depicted in figure 6. Starting from the vision block, the calculated position and orientation of the robot is fed to a prediction block, which calculates the best estimate of the position and orientation of the robot in the robot temporal frame based on the vision data and the history of the commanded velocities. The trajectory generation block solves a relaxation of an optimal control problem to calculate the future robot velocity profile required to reach the prescribed final position and final velocity in either the shortest possible time or in a prescribed amount of time using the least amount of control effort; a similar step occurs for the robot orientation. The dynamics of the motors and the robots are taken into account to ensure that the generated

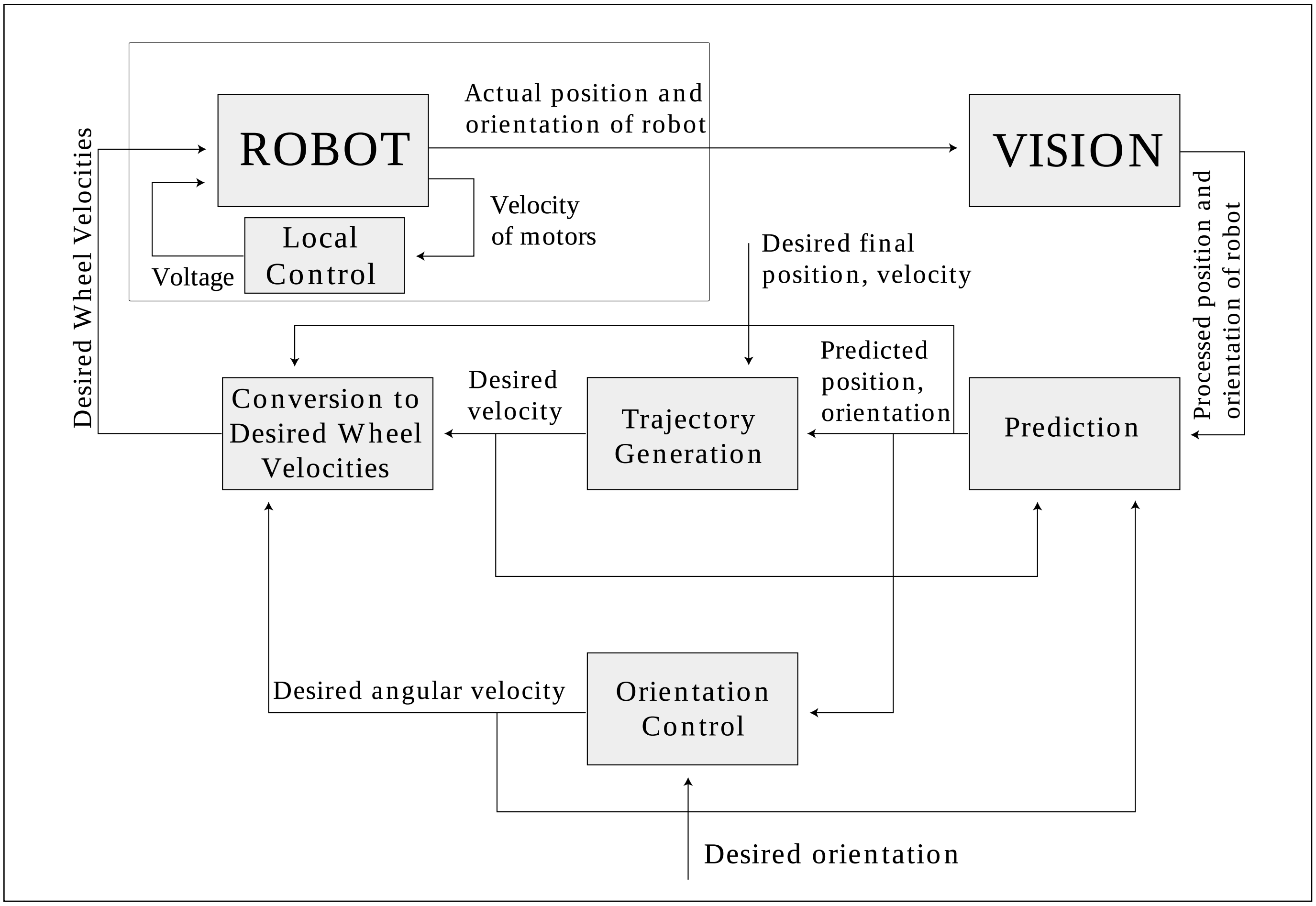


Figure 6. Block Diagram Representation of Trajectory Generation.

velocity profiles are feasible. After converting these data to wheel velocities, this information is fed to the robots by wireless communication. A local control loop on the robot regulates the actual wheel velocities about the desired wheel velocities.

Obstacle Avoidance

The Cornell team displayed advanced obstacle avoidance, in large part because of the hierarchical decomposition of overall robot control into trajectory generation and the higher-level algorithms used to determine where to send the robots. In the simplest case, the algorithm is based on determining if a collision will take place, followed by drawing tangents to the first obstacle known to be in the path to the destination. Once the tangent points are known, trajectories are generated for the destinations with the tangent points as via points, and the feasible path with either the shortest time or the least amount of control effort is followed. A refinement of the previous algorithm is used when there are multiple obstacles and when obstacles are sufficiently close to each other.

Observations

The proposed increase in the field size (at all RoboCup competitions so far, the field has been the size of a table-tennis table—the field might be widened for RoboCup-2001) will greatly reward teams that implement effective team play. We strongly feel that the dribbling mechanism will greatly improve the quality of the game and allow teams to effectively use sophisticated team-based strategies. The main benefit of the omnidirectional drive mechanism is a simplification of the resulting control problem, which greatly reduces the computation required for generating nearly optimal trajectories and, thus, frees up computational resources for higher-level control decisions; we do not feel, however, that it will be a necessary feature of future competitive teams. It is clear, however, that successful future teams must seriously address dynamics and control issues, such as estimation, coping with system latency, robustness, and optimal control; only by doing so can the full benefits of team play and cooperation be achieved.

The Middle-Size Robot League

The RoboCup F2000 League, commonly known also as the middle-size robot league, poses a unique combination of research problems, which has drawn the attention of well over 30 research groups worldwide.

Environment and Robots

The playing environment is designed such that the perceptual and locomotion problems to be solved are reasonably simple but still challenging enough to ignite interesting research. The field size is currently 9 meters x 5 meters. The goals have colored walls in the back (yellow) and on the sides (blue). The field is surrounded by white walls (50 centimeters high) that carry a few extra markings (squared black markers of 10-centimeter size plus black-and-white logos of sponsors in large letters). A special corner design is used and marked with two green lines. The goal lines, goal area, center line, and center circle are all marked with white lines. The ball is dark orange. Illumination of the field is constrained to 500 to 1500 luxes. Matches are played with teams of four robots, including the goalie.

The robots must have a black body and carry color tags for team identification (light blue and magenta). Quite elaborate constraints exist for robot size, weight, and shape. Roughly, a robot body can be as large as about 50 centimeters in diameter and be as high as 80 centimeters; must weigh less than 80 kilograms; and must have no concavities large enough to take up more than one-third of the ball's diameter. The robots must carry all sensors and actuators on board; no global sensing system is allowed. Wireless communication is permitted both between robots and between robots and outside computers.

During the penalty shootout, CS FREIBURG first scored three of five penalty kicks. Then, it was GOLEM's turn, and it scored the first penalty kick. Excitement was at its peak when it missed the next two. CS FREIBURG defended the next one as well and became the RoboCup-2000 Middle-Size League champion.

Research Challenges

The most notable difference from the F180 league is that global vision is not permitted. In a global camera view, all the robots and the ball move, and the goals, the walls, and the markings of the field remain fixed. If the moving objects can be tracked sufficiently fast in the video stream, all the positions and orientations are known, and a global world model is available. The situation is completely different in the F2000 league, where the cameras on top of the robots are moving through the environment. All the usual directional cameras, and most omnidirectional cameras, can perceive only a small part of the environment, which greatly complicates tasks such as finding the ball, self-localizing on the field, locating teammates and opponents, and creating and updating a world model. In addition, the vast majority of F2000 robots are completely autonomous, carrying all sensors and computational equipment on board, which makes them much larger and heavier. Fast movements are much more difficult to control. These are two of the main reasons why F2000 robots play at much slower speeds than F180 robots.

The difficulties described here exert a strong force on new teams to think about robot design, and repeatedly, new teams with new hardware designs have displayed stunning first-time appearances at RoboCup tournaments. This year, we had another two examples: CMU HAMMERHEADS from the United States and GOLEM from Italy, each of which introduced a new mobile base into the middle-size league. The HAMMERHEADS use a modified version of the commercially available CYE robot, a differential-drive base with a trailer attached to it. The GOLEM robots feature a triangular omnidirectional drive design based on mechanum wheels, which provided for the best combination of maneuverability and speed that the F2000 league has seen so far. The drive design of the GOLEM robots was complemented by the use of only a single sensor: an omnidirectional camera with a custom-made mirror design, which provided the robot with a complete view of the field from virtually every position. The clever combination of these two key design decisions allowed the GOLEM team to apply much simpler techniques for localization and world modeling as well as action selection, which significantly reduced development time.

RoboCup-2000 Tournament

Fifteen teams participated in the RoboCup-2000 middle-size–league tournament. The rules for the middle-size robot league were only marginally changed from last year, which gave teams the opportunity to focus on software improvements rather than the design of new hardware. The play schedule was designed to give all teams ample opportunity to gain practical playing experience, with a total of 57 games. Each team was assigned to one of two groups, with seven and eight teams, respectively, for the qualification rounds. Each group played a single round-robin schedule, such that each team played at least six or seven games. The four top teams in each group went to the playoff quarterfinals.

In this year's tournament, we had more exciting matches than ever, with quite a number of surprising performances. Most teams had previous tournament experience and showed significant progress over previous play levels. In addition, the two remarkable newcomers this year, the HAMMERHEADS and GOLEM, both made it to the quarterfinals, a remarkable success, especially for new teams. The other teams reaching the quarterfinals were last year's champion SHARIF CE from Iran; RMIT UNITED from Melbourne, Australia; the Osaka University TRACKIES from Japan; and the three German teams, GMD ROBOTS from Bonn, AGILO ROBOCUPPERS from Munich, and CS FREIBURG. GOLEM, SHARIF CE, TRACKIES, and CS FREIBURG qualified for the semifinals. The semifinals and finals matches were the most exciting games in middle-size–league history, watched by a crowd of more than 1000 enthusiastic spectators. Both the third-place game and the final game took penalty shootouts to determine the winners. Last year's champion, SHARIF CE, finished third after tying the TRACKIES 1–1 at full time and winning the penalty kicks 3–2. The final game between CS FREIBURG and GOLEM was tied 3–3 at full time. During the penalty shootout, CS FREIBURG first scored three of five penalty kicks. Then, it was GOLEM's turn, and it scored the first penalty kick. Excitement was at its peak when it missed the next two. CS FREIBURG defended the next one as well and became the RoboCup-2000 Middle-Size League champion.

Lessons Learned and Future Developments

When newcomer SHARIF CE won last year, many observers attributed its superior performance largely to the new hardware design, which gave it more speed and more maneuverability than most other teams. With the GOLEM team from Italy, we had yet another team with a new mobile platform making it to the finals. Many AI people were concerned that the focus in F2000 would shift mainly to new mechanical designs and hardware work. However, this year, CS FREIBURG won the championship because of its superior software capabilities; except for slightly redesigned kickers, the hardware design has remained almost the same since the team started out in 1998. Many teams have much faster, more maneuverable robots than CS FREIBURG.

After a year of keeping the rules virtually unchanged, it is now time to think about modifications that promote research, particularly in two directions:

Making robots more robust and reliable—Comparatively small changes in the environment often disturb the robots' performances significantly. Reducing the dependency on environmental color coding and developing fast and robust algorithms for perceptual tasks such as object detection, object localization, and object tracking are essential goals for future research.

Enhancing playing skills—Most robots push or kick the ball with a simple device; only few robots could demonstrate dribbling capabilities, such as taking the ball around an opponent in a controlled manner. Playing skills can be improved through the application of learning techniques. In addition, we need to relax some of our constraints on the robot's form and shape to promote the design of innovative ball-manipulation devices.

Rule changes to foster research in these directions can be expected for future tournaments.

CS FREIBURG: Middle-Size Champions

After winning RoboCup in 1998 and coming in third in 1999, CS FREIBURG again won the championship at RoboCup-2000. One of the reasons for this success is most probably the accurate and reliable self-localization method based on laser range finders (LRFs) (Gutmann, Weigel, and Nebel 1999). However, although this method was basically enough to win the competition in 1998, it was necessary for it to work on a number of different problem areas to stay competitive. Since 1998, the CS FREIBURG team has worked on improving the basic ball-handling skills, improving the action-selection mechanism, improving team play, and improving sensor data gathering and interpretation. These points are described in much more detail in Weigel et al. (2001). In particular, the first point implied some redesign of the hardware and software. Figure 7 shows one of the redesigned CS FREIBURG robots with the new kicking device and movable fingers. However, a new kicker and a new way of steering the ball is not enough. It is also necessary to develop basic behaviors that exploit the new hardware and develop a mechanism for selecting the appropriate behavior in a given situation.

New Tactical Skills: Dribbling and Rebound Shots

For the 2000 competition, the CS FREIBURG team put a lot of effort into developing a new set of basic skills to respond to a large number of different game situations. Some of the most important skills are described in the following paragraphs.

To get hold of the ball, a player moves to a position behind the ball following a collision-free trajectory generated by a path-planning system that constantly (re)plans paths based on the player's perception of the world (GoToBall). The system is based on potential fields and uses A* search for finding its way out of local minima. If close to the ball, a player

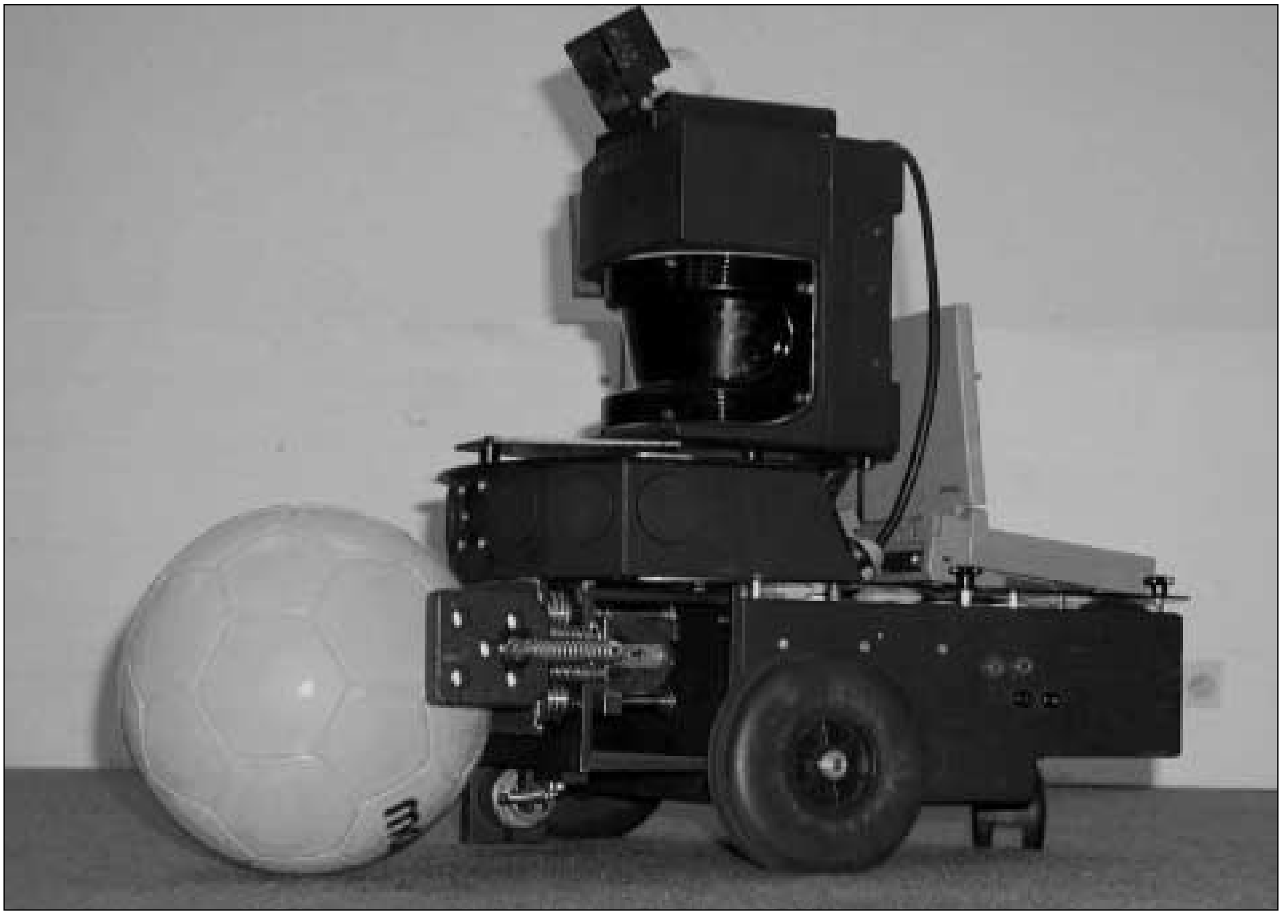


Figure 7. CS FREIBURG Player Mounting SICK LRF, Color Camera, Libretto Laptop, WAVELAN Wireless Ethernet, and Custom-Made New Kicking Device.

approaches the ball in a reactive manner to get it precisely between the fingers while it is still avoiding obstacles (ApproachBall). Once in ball possession, a player turns and moves the ball carefully until facing in a direction that allows for an attack (TurnBall). If the player is right in front of the opponent's goal, it kicks the ball in a direction where no obstacles block the direct way to the goal (ShootGoal). Otherwise, it first heads toward a clear area in the goal and turns sharply just before kicking in case the opponent goalkeeper moved in its way (MoveShootFeint). However, if obstacles are in the way to the goal, the player tries to dribble around them (DribbleBall) unless there is not enough room. In this case, the ball is kicked to a position close to the opponent's goal by also considering rebound shots using the walls. In the event of being too close to an opponent or to the field border, the ball is propelled away by turning quickly in an appropriate direction (TurnAwayBall). If a player gets stuck close to an obstacle, it tries to free itself by first moving away slowly and (if this doesn't help) then trying random moves (FreeFromStall).

Players fulfilling strategic tasks position themselves following collision-free paths (GoToPos) to dynamically determined positions. From these positions, the players either search the ball if not visible (SearchBall) by rotating constantly or observe it by turning until facing it (ObserveBall).

In RoboCup-2000, the CS FREIBURG team seemed to be one of the few teams capable of effectively dribbling the ball and the only one that deliberately exploited the possibility of rebound shots using the walls. Therefore, these two skills will be described in more detail.

Figure 8a shows a screen shot of a player's local view while it is dribbling. In every cycle, potential continuations of the current play are considered. Such continuations are lines to points closer to the opponent's goal within a certain angle range around the robot's heading. All the possible lines are evaluated, and the direction of the best line sample is taken as the new desired heading of the robot.

A line is evaluated by assigning it a value that is higher the further it is away from objects, the less turning is necessary for the player, and the

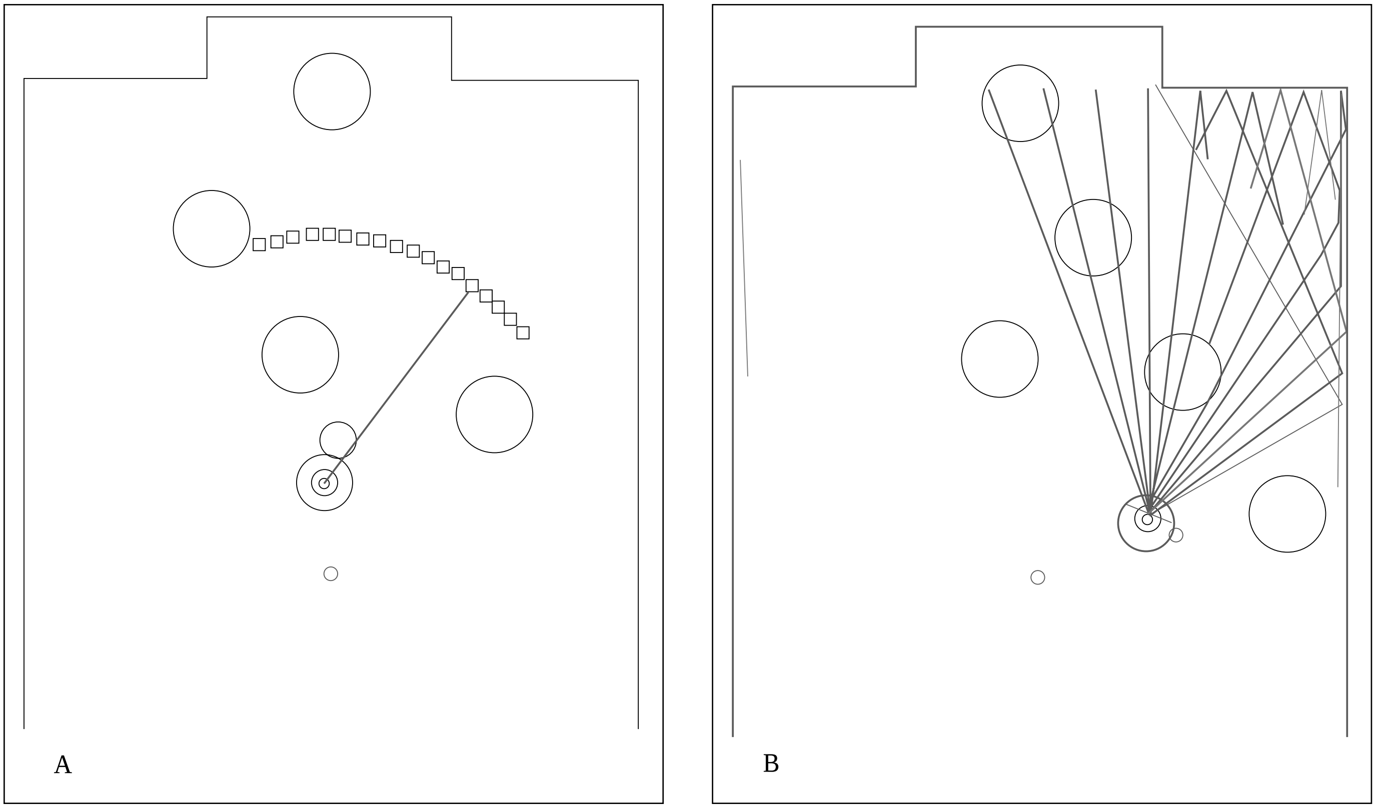


Figure 8. A CS FREIBURG Player's View of the World.

A. Dribbling. B. Ball passing. Circles denote other robots, and the small circle in front of the player corresponds to the ball. Lines almost parallel to the field borders are perceived by the laser range finder. The other lines leading away from the player are evaluated by the skills.

closer its heading is to the opponent's goal center. Determining the robot's heading this way and adjusting the wheel velocities appropriately in every cycle lets the robot smoothly and safely dribble around obstacles without losing the ball. The CS FREIBURG team scored some beautiful goals in this year's tournament after a player had dribbled the ball over the field around opponents along an *S*-like trajectory. Of course, all this sophisticated motion only works because the ball-steering mechanism allows for tight control of the ball.

Figure 8b shows a screen shot of a player during ball passing. For this skill, the lines are reflected at the walls and are evaluated only once to find the best direction in which to kick the ball. A line's value is higher the further it is from obstacles, the closer its end point is to the opponent's goal, and the less turning it requires for the player to face in the same direction. Using the passing skill, the players on the CS FREIBURG team were able to play the ball effectively to favorable positions and even score goals directly

Action Selection: Extended Behavior Networks

One of the critical components in a robotic soccer agent is the action-selection mechanism. As the outline of actions for playing the ball shows, a lot of different basic skills have to be taken into account, and for each situation, an appropriate action has to be chosen. During the development of the CS FREIBURG team, a number of different methods have been tried, none of them completely satisfactory. In 1999, extended behavior networks (Dorer 1999) were adapted to the needs of the CS FREIBURG team. This formalism, which had been used by RoboCup-99's runner-up in the simulation league, was developed based on Maes's (1990) proposal. It modifies Maes's proposal in particular by changing the activation mechanism in a way that makes the action selection appear to be closer to decision-theoretic planning.

This formalism allows for modular and flexible specification of behaviors and their interactions. In addition, it is possible to adjust tactics to opponents by supporting more defensive or offensive play.

Strategy: Roles and Placement

The CS FREIBURG players organize themselves in roles, namely, active, support, and strategic. Although the active player always tries to get and play the ball, the supporting player attempts to assist by positioning itself appropriately. The strategic player always occupies a good defensive position.

Each player constantly calculates its utility to pursue a certain role and communicates the result to its teammates. Based on its own and the received utilities, a player decides which role it wants to take. This approach is similar to the one taken by the ART team (Castelpietra et al. 2001); however, the CS FREIBURG players additionally communicate to their teammates which role they are currently pursuing and which role they want to take. A role can only be taken from another player if its utility for this role is the best of all players, and the robot currently pursuing the role also wants to change its role. Following this strategy makes it less likely that two or more players are pursuing the same role at the same time than assigning rules based on utility values only.

The target positions of the players are determined like the SPAR method of the CMUNITED team in the small-size league (Stone, Veloso, and Riley 1999). From the current situation observed by the robots, a potential field is constructed that includes repulsive forces arising from opponent players and attracting ones from desirable positions, for example, positions from where the ball is visible. Positions are then selected based on the robot's current role; for example, the position of the active player is set close to the ball, the supporting player is placed to the side and behind the active one, and the strategic player takes a defensive position that is about half way between its own goal and the ball but behind all opponent players.

Observations

The success of the CS FREIBURG team this year can clearly be attributed to the effective team play and the rich set of basic ball-handling skills. Always being present at strategically important positions compensated for the comparatively slow robots of the CS FREIBURG team. The basic skills enable the robots to move quickly to the ball and offer a variety of different ball-handling actions exploiting the new powerful kicking and ball-steering mechanism. As demonstrated, for example, in the game against CS SHARIF, the CS FREIBURG players did extremely well in getting to the ball and blocking the opponent before it could actually become dangerous.

One of the experiences was that tuning the parameters of the basic skills by hand was very time consuming. Therefore, some future work will concentrate on learning methods for parameter adjustment of some of the basic skills.

The Simulation League

The RoboCup-2000 competition was the most exciting and most interesting simulation competition to date. As in past years, the competition was run using the publicly available SOCCER SERVER system (Noda et al. 1998). Thirty-four teams from 14 countries met in a round-robin competition followed by a double-elimination final series. Although most of the teams had competed in previous competitions, there were several notable new entries, including the eventual champion, FC PORTUGAL, which had an exciting 1–0 final with the Karlsruhe BRAINSTORMERS. The high standard of the competition made for many exciting matches throughout the competition; nearly 25 of the final-round games went into overtime, 1 eventually having to be decided by a coin toss after scoreless overtime lasted the length of 2 normal matches.

The RoboCup SOCCER SERVER

The RoboCup SOCCER SERVER provides a standard platform for research into multiagent systems. The SOCCER SERVER simulates the players and field for a two-dimensional (2D) soccer match. Twenty-two clients (11 for each team) connect to the server, each client controlling a single player. Every 100 microseconds, the SOCCER SERVER accepts commands, by socket communication, from each client. The client sends low-level commands (dash, turn, or kick) to be executed (imperfectly) by the simulated player it is controlling. Clients can only communicate with each other using an unreliable, low-bandwidth communication channel built into the SOCCER SERVER. The soccer server simulates the (imperfect) sensing of the players, sending an abstracted (objects, for example, players and ball, with direction, distance, and relative velocity) interpretation to the clients every 150 microseconds. The field of view of the clients is limited to only a part of the whole field. The SOCCER SERVER enforces most of the basic rules of (human) soccer, including offsides, corner kicks, and goal kicks, and simulates some basic limitations on players such as maximum running speed, kicking power, and stamina limitations.

An extra client on each team can connect as a coach, who can see the whole field and send strategic information to clients when the play is stopped, for example, for a free kick.

The soccer monitor connects to the SOCCER SERVER as another client and provides a 2D visualization of the game for a human audience (figure 9). Other clients can connect in the same way to do things such as 3D visualization, automated commentary, and statistical analysis.

Figure 9. A Screen Shot of the SOCCER SERVER *Monitor Augmented with* FC PORTUGAL*'s Debugging Tools.*

Research Themes

Many of the research challenges addressed by teams in 2000 came out of problems observed by teams from previous competitions. Two research themes were especially prominent: (1) learning and (2) multiagent coordination. Other research areas included improving situational awareness given incomplete and uncertain sensing and high-level team specification by human designers.

The first research theme, learning, was especially common among the successful teams. Teams adapted techniques such as simulated annealing, genetic programming, or neural nets to the problem of creating optimized low-level skills such as dribbling (Riedmiller et al. 2001). Experience has shown that although advanced skills were an essential component of a successful team, building such skills by hand is difficult and time consuming. The skills developed with learning techniques were in some cases superior to the hand-developed skills of previous years. Hence, RoboCup has provided a useful, objective example where learning produced a better outcome than labor-intensive programming.

Not all learning research was focused on low-level skills; several teams addressed the problem of how to learn high-level strategies. RoboCup provides an interesting domain to investigate such issues because although there is a clearly defined objective function, that is, win the game, the huge state space, unpredictable opponent, uncertainty, and so on, make the problem challenging. Most approaches learning at a high level layered the learning in some way (a successful approach in the 1999 competition), although the specifics of the learning algorithms varied greatly, from neural networks to evolutionary algorithms.

The second major research theme was multiagent coordination. Although in previous competitions, a highly skilled team might do reasonably well with kiddie soccer tactics, for example, dribbling directly to goal, so many teams this year had high-quality skills that more sophisticated team strategies were required to win games. Conversely, the high-quality skills triggered more interest in team strategies because players had the ability to carry them out with some consistency. As well as the learning approach to developing high-level strategies, a variety of human engineered approaches were used (for example, Murray, Obst, and Stolzenburg [2001]). A key to many of the approaches was the online coach. The coach was commonly used to analyze the opposition and determine appropriate changes to the team strategy (Esaki et al. 2001). Other teams developed tools or techniques aimed at empowering human designers to easily specify strategies, yet other teams relied on carefully engineered emergent team behavior (for example, Prokopenko, Butler, and Howard [2001]) or dynamic team planning to achieve the desired team behavior.

RoboCup simulation teams are increasingly complex pieces of software usually consisting of tens of thousands of lines of code with specialized components working together in real time.

The Competition

RoboCup simulation teams are increasingly complex pieces of software usually consisting of tens of thousands of lines of code with specialized components working together in real time. Handling the complexity is forcing researchers to look critically at agent paradigms not only in terms of the resultant agent behavior but also at the ease with which very complex teams can be developed within the paradigm (and how it should be done).

However, the rapidly increasing complexity of RoboCup simulation agents should not deter new researchers from starting to work with RoboCup. An online team repository currently contains source code or binaries for 29 of the teams that competed in the 1999 World Cup plus many more from previous years. The repository allows new RoboCup participants to quickly get a team going. In fact, a number of the top teams in 2000 were developed on top of the freely available code of the 1999 champions, CMUNITED-99 (Stone, Riley, and Veloso 1999). The growing code base provides code for interaction with the SOCCER SERVER, skills, strategies, debugging tools, and so on, in a variety of programming languages and paradigms.

The reigning champion team, CMUNITED-99, was reentered—unchanged—in the 2000 competition to assess the advances made during the year. In 2000, CMUNITED-99 finished fourth. In 1999, CMUNITED-99's aggregate goals for and against tally was 110–0, but in 2000, the tally was far more competitive, 25–7 (including a 13–0 win). Also interesting was that four of six of CMUNITED-99's elimination-round games went into overtime (resulting in three wins and one loss). CMUNITED-99's record in 2000 shows two things: First, although they finished fourth, several teams were nearly as good and, perhaps, unlucky to lose to them, and second, the competition was extremely tight. It also indicates just how good CMUNITED-99 was in 1999.

As well as the main competition, there were extensive evaluation sessions designed to compare the ability of teams to handle increased sensor and effector uncertainty. The sensor test was a repeat of the test from last year and involved changing the average magnitude of the error in the simulated visual information players received. The effector test was a surprise to the teams and involved changing the average magnitude of the difference between what command was sent by a player and what was actually executed. The evaluation session provides a unique opportunity to test a wide variety of agent implementations under identical conditions. Extensive evaluation log files, providing a large amount of high-quality data, are available for analysis.

Despite the advances made in 2000, the RoboCup simulator is far from a solved problem. Although high-level learning has progressed significantly, learned high-level strategies were generally inferior to hand-coded ones; a challenge for 2001 is to have learned strategies outperform hand-coded ones. Using RoboCup simulation as a platform for research into high-level multiagent issues is only just starting to emerge, for example, with the use of the online coach. Additionally, as the standard of play gets higher, there is both increased interest and use for opponent-modeling techniques that can counter complex, previously unseen team strategies. The rapidly increasing complexity of RoboCup software challenges us to continue improving our methods for handling complexity. The advances made and the research areas opened up in 2000 bode well for yet another interesting, exciting competition in 2001.

FC PORTUGAL: Simulator League Champion

FC PORTUGAL is the result of a cooperative project that started in February 2000 between the University of Aveiro and the University of Porto in Portugal. FC PORTUGAL won both the

RoboCup 2000 Simulation League European and World championships, scoring a total of 180 goals and conceding none.

CMUNITED-99 source code (Stone, Riley, and Veloso 1999) was used as a starting point, allowing the development effort to focus on more interesting research issues. We pursued a variety of research threads covering all aspects of RoboCup team development, with the overriding themes being multiagent cooperation and coordination. At the team level, FC PORTUGAL introduces the concept of tactic and incorporates novel algorithms for using flexible, dynamic team formations, including the ability for players to dynamically change positionings and roles. Intelligent communication provides all players with an accurate picture of the world despite the uncertainty and limited field of view enforced by the simulation. At the individual level, interesting aspects of FC PORTUGAL include intelligent perception, qualitative reasoning about action selection through integration of real soccer knowledge, and the use of online optimization techniques for low-level skills. Several important development tools were used, including a visual debugger (figure 9), advanced replay facilities for human developers, and a world-state error analyzer. Because of space constraints, it is not possible to fully explain all the FC PORTUGAL advances in detail. We describe selected features here and refer the interested reader to Reis and Lau (2001) for more details.

Team Strategy Definition and Situation-Based Strategic Positioning

CMUNITED brought the concepts of formation and positioning to robot soccer (Stone 1998; Stone, Riley, and Veloso 1999) and used dynamic switching of formations as well. FC PORTUGAL extends this concept and introduces the concepts of tactics and player types. FC PORTUGAL's team strategy is based on a set of tactics to be used in different game situations and a set of player types. Tactics include several formations used for different game-specific situations (defense, attack, goalie free kick, scoring opportunity, and so on). Formations are composed of 11 positionings that assign each player a given player type and a base strategic position on the field.

One of the most significant features is the clear distinction between strategic situations (when the agent believes that it is not going to use an active behavior soon) and active situations (ball recovery and ball possession). In strategic situations, players use a situation-based strategic positioning (SBSP) mechanism that adjusts its base strategic position according to the ball-position and -velocity and player-type strategic information. The result is the best strategic position in the field for each player in each situation. Because at each time, only a few players are usually using active behaviors, SBSP enables the team to move like a real soccer team, keeping the ball well covered while it remains strategically distributed around the field. For active situations—ball possession, ball recovery, and stopped game—decision mechanisms based on the integration of real soccer knowledge are used.

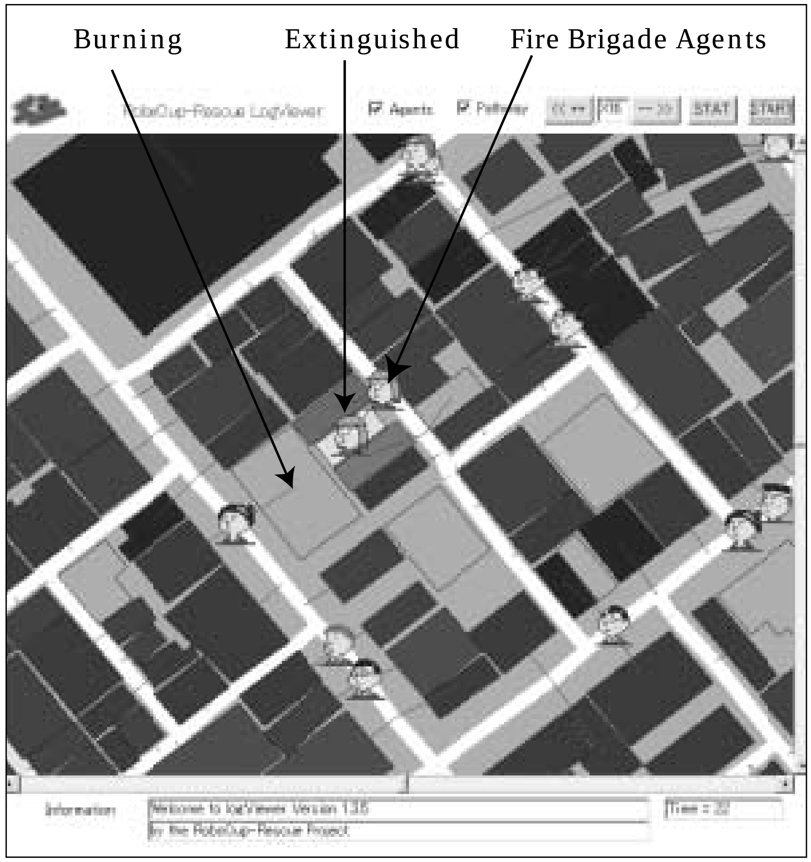


Figure 10. Two-Dimensional Viewer Image of RoboCup Rescue Prototype Simulator.

Intelligent Perception and Communication

In a complex domain such as soccer, the multiple sensors of an agent must be coordinated and used in an intelligent way to give the agent the most accurate understanding of the current state of the world possible. Players receive world information by their vision system, close-range touch sensors, and shouted information from teammates. The SOCCER SERVER limits the agent's viewing distance and viewing angle. Hence, at any time, there are large parts of the field that the player cannot see. However, the player has a neck that can be turned

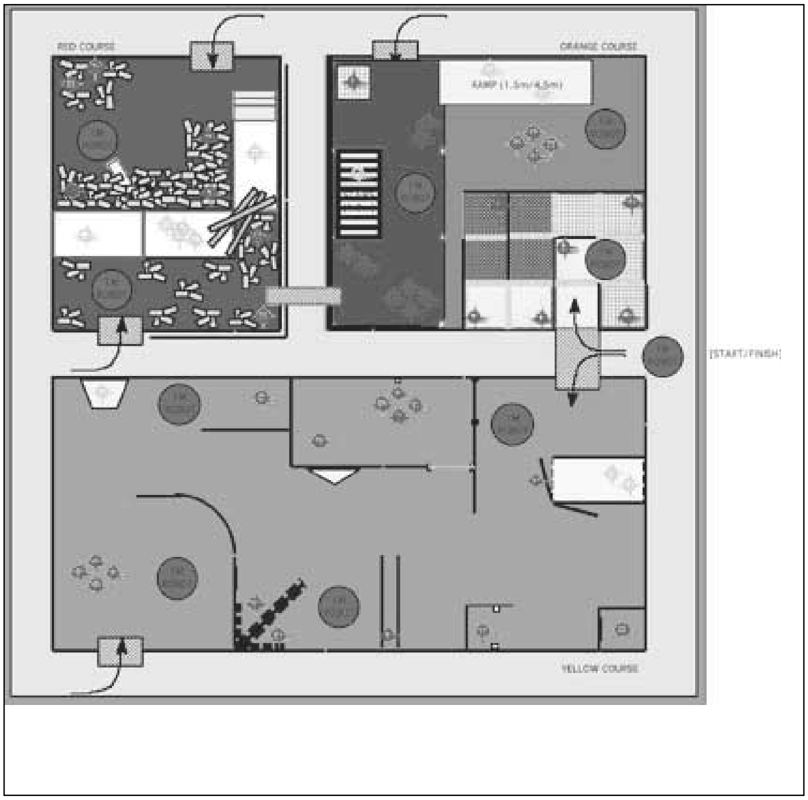


Figure 11. Urban–Search-and-Rescue Contest Field.

independently of it's body (within some limits), so it need not be looking in the direction it is moving. In different situations, different aspects of the world are more or less important to the player. The server also limits communication between teammates; namely, messages are broadcast over a limited range around the talking player, and players can only hear one message from their teammates each two simulation cycles.

FC PORTUGAL's strategic looking mechanism (SLM) intelligently determines the direction the player should turn its neck based on its current information availability and requirements. SLM decides on a direction to look by calculating the utility of each possible direction the agent could look and selecting the direction with the highest utility. The utility is calculated by assessing the areas of the ground where important information is likely to be sensed and for which the agent does not already have the appropriate information. For example, in an attacking situation, high utility might be ascribed to looking in the direction of the goal because this information could help determine whether a shot on goal was a good option. FC PORTUGAL agents use communication to maintain agents' world states that are updated by sharing individual knowledge and increase team coordination by communicating useful events (for example, position swap). Players evaluate the utility of talking based on the comparison of assumed teammate knowledge (from received messages) with their own knowledge. It should be noted that hearing a teammate message prevents hearing other messages that might be more useful. FC PORTUGAL agents talk only when they believe that the utility of their communication is higher than that of their teammates.

Kick Optimization

The ability of RoboCup players to kick the ball powerfully and accurately is a valuable asset. However, producing such kicks is not an easy task. To kick, the player issues a command indicating the direction and force with which the ball should be kicked. The resulting ball velocity depends on the position of the ball relative to the player, its previous velocity, and the direction of the kick. Acceleration of the ball to a high speed can take several kicks, for example, to first position the ball appropriately, then multiple kicks to accelerate it.

FC PORTUGAL used optimization techniques to create a good kicking ability based on a succession of basic kicks. The optimization process has two steps that are performed online during the game each time players want to kick the ball powerfully. First, random search and simple heuristics are used to generate kick sequences for the given situation (that is, initial ball position-velocity and desired kick angle-velocity). Kick sequences are evaluated based on final speed, number of basic kicks, and possible opponent interference. Second, hill-climbing search tries to improve the best kick sequence found by random search. This method resulted in flexible, fast kicking skills that provided FC PORTUGAL players a solid basis for executing higher-level strategies.

RoboCup Rescue

The RoboCup Rescue Project was newly launched by the RoboCup Federation in 1999. Its objective is as follows: (1) development and application of advanced technologies of intelligent robotics and AI for emergency response and disaster mitigation for the safer social system; (2) the introduction of new practical problems with social importance as a challenge of robotics and AI, indicating a valuable direction of research; (3) proposal of future infrastructure systems based on advanced robotics and AI; and (4) acceleration of rescue research and development by the RoboCup competition mechanism. A simulation project is running at present, and a robotics and infrastructure project will soon start.

In Melbourne, a simulator prototype targeting earthquake disaster was open to the public to start international cooperative research. A real rescue robot competition was proposed to start a new league in 2001.

Simulation Project

Distributed simulation technology combines the following heterogeneous systems to make a virtual disaster field: (1) disaster simulators model the collapse of buildings, blockage of streets, spread of fire, traffic flow, and their mutual effects; (2) autonomous agents represent fire brigades, policemen, and rescue parties, all of which act autonomously in the virtual disaster; (3) the simulation kernel manages state values and networking of and between the systems; (4) the geographic information system gives spatial information to the whole system; and (5) simulation viewers show 2D and 3D images of simulation results in real time, as shown in figure 10.

The RoboCup Rescue simulation competition will start in 2001. The details are described in Kaneda et al. (2001), Kuwata and Shinjoh (2001), Ohta (2001), Takahashi et al. (2001), Tadokoro and Kitano (2000), and Tadokoro et al. (2000). The simulator prototype can be downloaded from robomec.cs.kobe-u.ac.jp/robocup-rescue/.

AAAI-RoboCup Rescue Robot Competition

A rescue robot competition will start in 2001 in cooperation with the American Association for Artificial Intelligence (AAAI). The target is search and rescue of confined people from collapsed buildings, such as in earthquake and explosion disasters. In Melbourne, Robin Murphy (University of South Florida) demonstrated two robots that have been developed for real operations.

The large-scale arena of the AAAI Urban–Search-and-Rescue (USAR) Contest (figure 11) will be used. It consists of three buildings simulating various situations. The easiest building has a flat floor with minimal debris, but the most difficult building includes a 3D maze structure consisting of stairs, debris, and so on, with narrow spaces. The details are described on the AAAI USAR web page (www.aic.nrl.navy.mil/~schultz/aaai2000/).

More than other RoboCup competitions, the rules of the 2001 rescue competition will focus on direct technology transfer, specifically to real disaster problems on the basis of the 2000 AAAI USAR contest. For example, practical semiautonomy with human assistance and information collection for realistic operation are potential competition components.

Figure 12. Images of the RoboCup Jr. Events.

RoboCup Junior

RoboCup Jr. is the educational branch of RoboCup, and it puts emphasis on teaching young people about research and technology by giving them hands-on experience. RoboCup Jr. development was initiated in 1997, and the first public show was at RoboCup-98 in Paris with a demonstration of LEGO MINDSTORMS robots playing soccer in a big LEGO stadium with rolling commercials; LEGO spectators making the wave, stadium lights, and so on (Lund et al. 1999); and children playing with other LEGO robot models. In 1999, during RoboCup-99 in Stockholm, children were allowed to program their own LEGO MINDSTORMS robots in the morning, and then play tournaments in the afternoon (Lund and Pagliarini 2000). The fast development of complex robot behaviors was

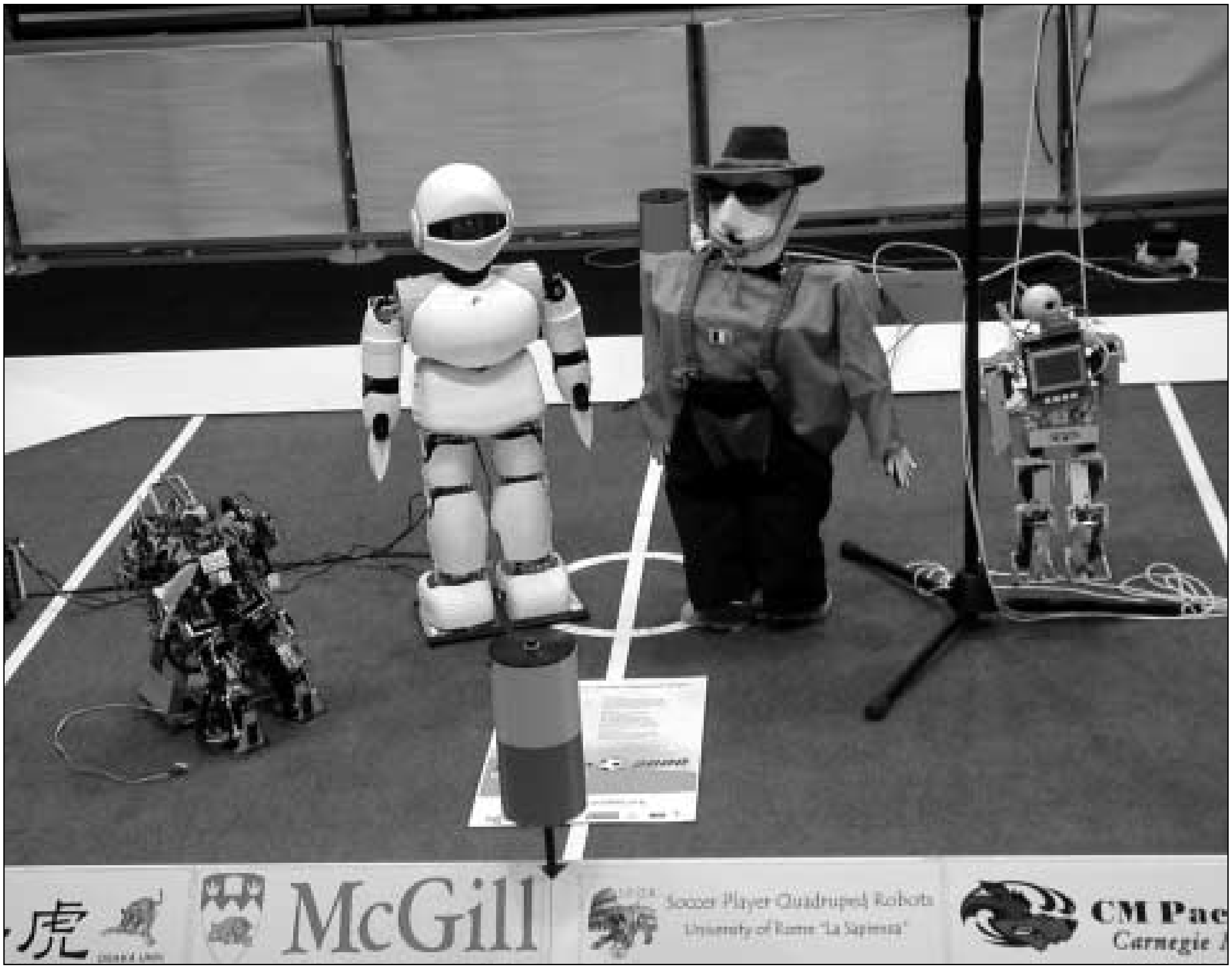


Figure 13. Four Humanoids Demonstrated at RoboCup-2000 (left to right): MARK-IV, PINO, ADAM, *and* JACK DANIEL.

achieved with the interactive LEGO football set-up based on a user-guided approach to behavior-based robotics. This activity was refined for RoboCup-Euro-2000 in Amsterdam (Kröse, Bogaard, and Hietbrink 2000), where 10 Dutch and 2 German school groups participated in a one-day tournament.

The RoboCup Jr. 2000 activity in Melbourne, in which a total of 40 groups of children participated, differed from the previous activities in several aspects: (1) children were both building and programming their robots; (2) the development took place during the six to eight weeks prior to the competition; (3) in most cases, the work was done as part of a teaching project in schools; and (4) there was a robot sumo competition and a robot dance performance in addition to the soccer competition.

During previous events, children had no opportunity to build the robots. However, educational approaches such as construction (Lund 1999) suggest that the construction of an artifact is important to understand the artifact; so, RoboCup Jr. 2000 allowed children to both build and program the robots. This endeavor was facilitated by the use of LEGO MINDSTORMS robots, partly because this tool allows for easy assembly of robots and partly because most children are familiar with LEGO. The tasks were designed so that the simple sensors and actuators are sufficient, but a few children from the more advanced technical classes made their own sensors and integrated them with the LEGO MINDSTORMS control unit.

There were three different events during RoboCup Jr. 2000: (1) the Dance-Performance Event for students to 12 years (primary school), (2) the Converging Robot Race (Sumo) for students to 14 years (grades 7 and 8), and (3) RoboCup Jr. 2000 Soccer for students 14 to 18 (grades 7–12). We put special emphasis on broadening RoboCup Jr. from being a purely competitive event to include the cooperative event of a robot dance-parade. In previous years, and during RoboCup Jr. 2000, we found the competitive robot soccer event to result in a gender bias toward boys. This bias is not surprising because the robot soccer event promotes soccer, technology, vehicles, and competition, and we often find that boys are more enthusiastic about these subjects than girls. We did not perform any rigorous scientific gender studies, but our experience from many events gave a clear picture of a gender bias. We therefore introduced the dance-parade to address other issues, such as cooperation, context construction, and performance. Indeed, more than 50 of the participants who signed up for the robot dance-performance event were girls.

Each participating team had three minutes for the robot dance-performance. The teams designed the robots, designed the environment in which the robots danced, programmed the robots to perform, and made a music cassette with the appropriate music for the performance. Many of the teams also designed their own clothes to match the robots and the environment, and many teams designed clothes for the robots. There was no limitation to the hardware (any robot could be used), but during RoboCup Jr. 2000, all participating teams chose to use LEGO MINDSTORMS. The performing robots included a Madonna look-alike, a disco-vampire, a dragon on the beach, and four feather-dressed dancers. Ten teams participated in the dance-performance event, and prizes were given for best dressed robot, best programming, best choreography, most entertaining (best smile value), best team T-shirt design, best oral presentation by participants to judges, and creativity of entry.

The RoboCup Jr. soccer game had 20 participating teams. Each team built one or two robots (in all cases from LEGO MINDSTORMS) to play on a field of approximately 150 centimeters x 90 centimeters. The floor of the field is a gradient from black to white, which allows the robots to detect position along one of the axes by measuring reflection from the floor with a simple light sensor. The ball used in the finals was an electronic ball produced by EK Japan (Lund and Pagliarini 2000). The ball emits infrared light that can be detected with simple, off-the-shelf LEGO sensors. Bellarine Secondary College won the final event by drawing 3–3 and winning on golden goal after being down 3–0 at half time.

The success of the RoboCup Jr. 2000 event was to a large degree because of the involvement of enthusiastic local teachers and toy-hardware providers, who promoted and designed the event in collaboration with the researchers. The local teachers were able to incorporate the RoboCup Jr. project into their curricula. Involvement of local teachers seems crucial for the success of such events. In the future, RoboCup Jr. will make an effort to promote national and local competitions, apart from the big events at the yearly RoboCup. Figure 12 shows images from this year's event.

Humanoid Robot Demonstration

The RoboCup humanoid league will start in 2002 toward the final goal of RoboCup, which is to beat the human World Cup soccer champion team with a team of 11 humanoid robots by 2050. This league will be much more challenging than the existing ones because the dynamic stability of robots walking and running will need to be handled.

The main steps of such development will be (1) building an autonomous biped able to walk alone on the field; (2) locomotion of this biped, including straight-line movement, curved movement, and in-place turns; (3) identification of the ball, the teammates, and the opponents; (4) kicking, passing, shooting, intercepting, and throwing the ball; (5) acquisition of cooperative behavior (coordination of basic behaviors such as passing and shooting); and (6) acquisition of team strategy.

Although items 3 to 6 are already addressed in the existing leagues, the humanoid league has its own challenges related to handling the ball with feet and hands.

At RoboCup-2000, the humanoid demonstration was held with four characteristic humanoids. Figure 13 shows these four humanoids, pictured from left to right. MARK-V is from K. Tomiyama's group at Aoyama Gakuin University. MARK-V showed its ability to walk and kick a ball into a goal. PINO is from the Kitano Symbio Project, Japan. It demonstrated walking and waving its hand to say good-bye! ADAM, from LRP, France, walked 100 centimeters in a straight line autonomously and was also controlled by an off-board computer. JACK DANIEL, from Western Australia University, demonstrated a walking motion while it was suspended in the air.

These humanoids are still under development. At RoboCup-2001 we expect to see more humanoids with improved walking and running and also some new capabilities.

RoboCup Workshop and Challenge Awards

There is no doubt that RoboCup is an exciting event: The matches are thrilling to watch and the robots and programs are fun to design and

RoboCup is fundamentally a scientific event. It provides a motivating and an easy-to-understand domain for serious multiagent research. Accordingly, the RoboCup Workshop, which is held each year in conjunction with the Robot Soccer World Cup, solicits the best work from participating researchers for presentation.

build. Even so, RoboCup is fundamentally a scientific event. It provides a motivating and an easy-to-understand domain for serious multiagent research. Accordingly, the RoboCup Workshop, which is held each year in conjunction with the Robot Soccer World Cup, solicits the best work from participating researchers for presentation.

The RoboCup-2000 Workshop was held in Melbourne, adjacent to the exhibition hall where the competitions were staged. Twenty papers were selected for full presentation, and an additional 20 were selected for poster presentation from over 60 submissions. Paper topics ranged from automated intelligent sportscaster agents to motion planners and vision systems. The workshop was attended by more than 200 international participants.

The number of high-quality submissions to the RoboCup Workshop continues to grow steadily. To highlight the importance of the scientific aspects of RoboCup, and to recognize the very best papers, the workshop organizers nominated four papers as challenge award finalists. The challenge awards are distinctions that are given annually to the RoboCup-related research that shows the most potential to advance their respective fields. The finalists were a localization method for a soccer robot using a vision-based omnidirectional sensor by Carlos Marques and Pedro Lima (2001); behavior classification with self-organizing maps by Michael Wünstel, Daniel Polani, Thomas Uthmann, and Jürgen Perl (2001); communication and coordination among heterogeneous midsize players, ART99, by Claudio Castelpietra, Luca Iocchi, Daniele Nardi, Maurizio Piaggio, Alessandro Scalso, and Antonio Sgorbissa (2001); and adaptive path planner for highly dynamic environments by Jacky Baltes and Nicholas Hildreth (2001).

These presentations were evaluated by a panel of judges who attended the presentations based on the papers themselves as well as the oral and poster presentations at the workshop. This year two awards were given: The Scientific Challenge Award was given to Wünstel et al. for their work on applying self-organizing maps to the task of classifying spatial agent behavior patterns, and the Engineering Challenge Award was given to Marques and Lima for their contribution to sensing and localization.

We expect that the workshop will continue to grow. In future years, we might move to parallel tracks so that more presentations will be possible.

The RoboCup-2000 Scientific Challenge Award went to Michael Wünstel, Daniel Polani, Thomas Uthmann, and Jürgen Perl (2001) for their method for using self-organizing maps (SOMs) to classify and structure spatio-temporal data.

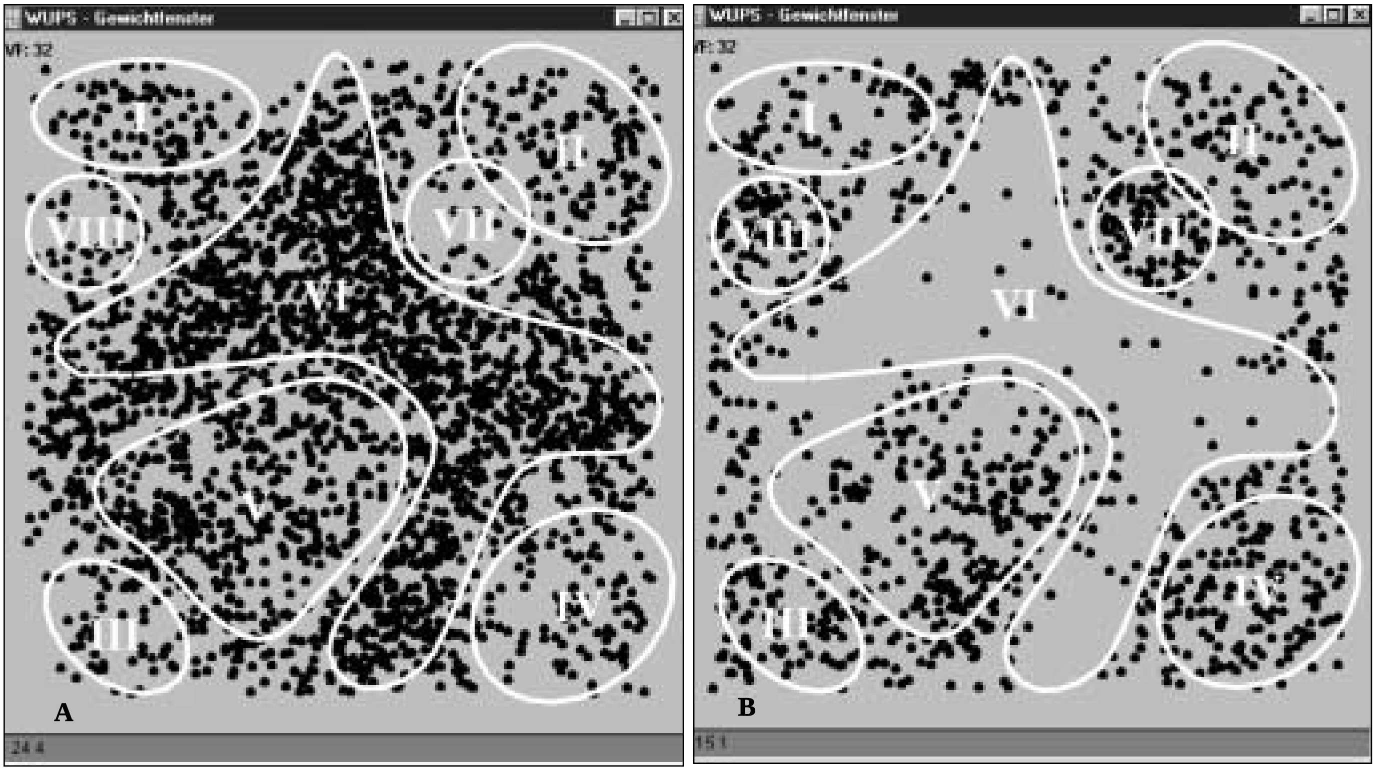
Scientific Challenge Award

The RoboCup-2000 Scientific Challenge Award went to Michael Wünstel, Daniel Polani, Thomas Uthmann, and Jürgen Perl (2001) for their method for using self-organizing maps (SOMs) to classify and structure spatiotemporal data. The goal of the work was to develop a method to detect characteristic features of trajectories. It was used to analyze the behaviors exhibited by RoboCup players during tournament games. The games took place in simulation using the SOCCER SERVER, so the complete game data were available as log files. The actions of RoboCup players were then analyzed on a purely behavioral level; that is, no knowledge about implementation or inner states of the agents was used. Here, an outline of the method is given. For details of the method, the reader is referred to the original paper.

The model is based on Teuvo Kohonen's (1989) SOM. The SOM is a data-analysis method inspired by the structure of certain cortex types in the mammal brain. It is able to identify clusters in high-dimensional data and project (map) these data onto a 2D grid respecting their topology, that is, their neighborhood structure that allows an intuitive visualization. The mapping and the visualization capability are an advantage of the SOM over standard statistical methods; in particular, the SOM is not just useful for separating different clusters, but it also resolves the inner structure of the clusters. Mathematically, it is related to the principal surface models for data distributions known from statistics (Ritter, Martinetz, and Schulten 1992), although unlike the SOM, these surface models are not designed to handle data sets decomposing into different clusters.

In the SOM model, two spaces exist: (1) the (typically) high-dimensional data space and (2) the space of SOM units, having a function similar to the code-book vectors of vector quantization. Unlike in vector quantization, however, the SOM units are typically organized in a two-dimensional grid, representing a topological, that is, neighborhood, relation between the units.

For a trained SOM, each data point from the high-dimensional space is projected onto an element of the two-dimensional SOM grid. Every SOM unit represents a vector in the high-dimensional data space, such that the SOM can be viewed as an embedding of the two-dimensional SOM grid into the high-dimensional data space. Neighboring units typically represent neighboring regions in the original data space. In turn, a data point from the high-dimensional space can be projected onto that



unit in the low-dimensional grid whose code vector is closest to the original point.

One of the questions addressed in the paper is the representation of trajectories of individual players and of players interacting with the ball. There are different approaches to represent trajectories adequately to be able to analyze them with SOMs. One method is to project every state of the original trajectory to the corresponding SOM unit and examine the resulting trajectory on the SOM grid in a fashion similar to the trace of an elementary particle in a cloud chamber (James and Miikkulainen 1995). One can then, for example, compare the order of activated units with some reference order. This method has also been used successfully for example in Carpinterio (1999) to recognize instances of a theme in a piece of music by J. S. Bach. It has also been used in speech recognition to trace the order of phonemes (Kangas 1994; Mehler 1994). These are static SOM models of trajectories in the sense that the SOM units only represent certain states, and the data-space trajectories are transformed into a trajectory on the SOM grid. A different representation was used by Chappell and Taylor (1993) who used leaky integrator units. These units can store the units' activations for a while and therefore are able to represent temporal information. The method presented by Wünstel et al. (2001), however, adopts a dynamic view of the trajectory representation. It does not attribute just a single state to an SOM unit; instead, each unit stores a trajectory slice containing several successive states. Therefore, in this model, the mapping performed by the SOM training does not just project the static state space but the space of trajectory slices onto the SOM grid and, thus, pro-

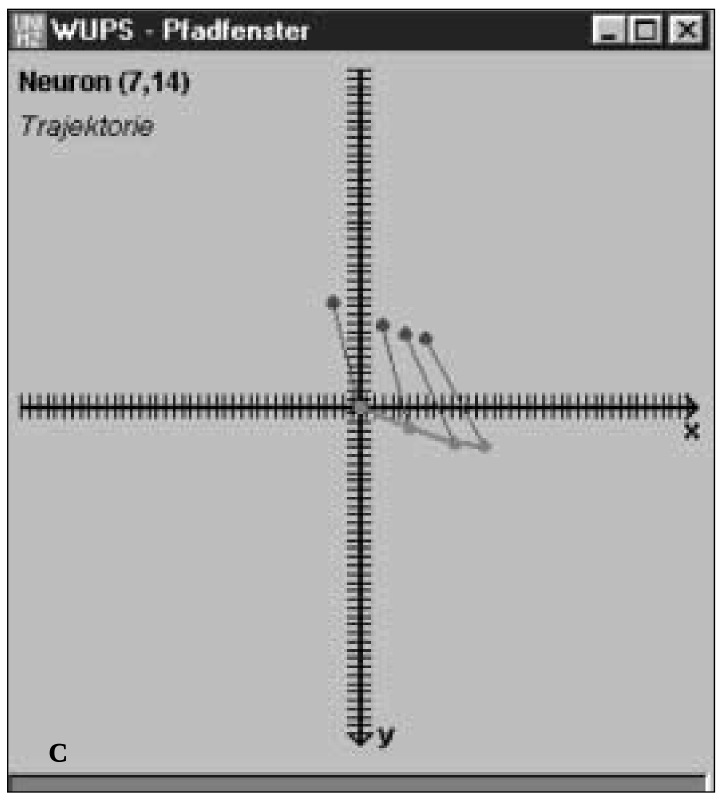


Figure 14. Combined Player-Ball Behavior.

A, B. A classification of different types of combined player-ball behavior, as found by the self-organizing map. C. A typical representative of such combined behavior from region VI of the left and middle plots (ball is dribbled alongside the player).

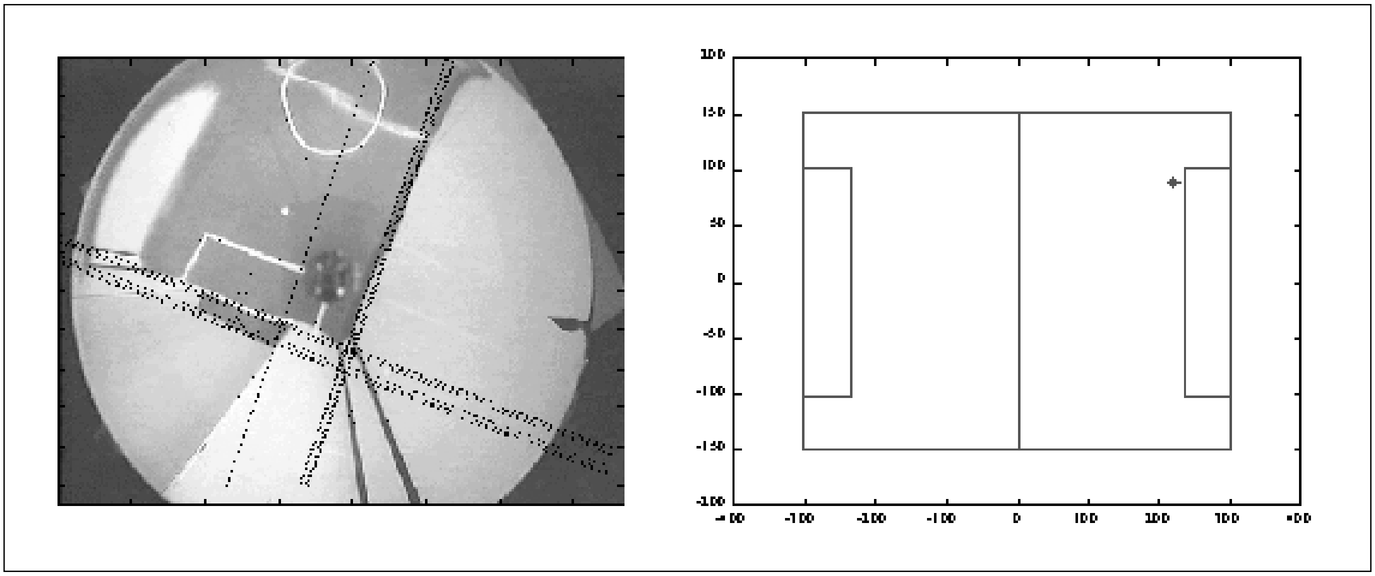


Figure 15. Self-Localization Results.

Left: The actual image. *Right:* The determined position in the field geometric model.

vides a mapping of the dynamic structure of the original trajectories.

In the paper discussed here, we used a method labeled spatially focused representation (SFR). We consider a trajectory as a simple sequence of spatial data: The time intervals are fixed and discrete, and the spatial data are continuous. An agent trajectory in a RoboCup simulation consists of a sequence of agent positions on the two-dimensional soccer field, one for each discrete simulation time step. The trajectory is split into windows of short length (six time steps in the current case). The difference vectors of the agent positions in two successive time steps are calculated (giving five difference vectors in the current example). This sequence of difference vectors is now concatenated, resulting in a combined vector (10-dimensional in the example). A complete player trajectory is thus transformed into a set of trajectory slices represented as a sequence of position difference vectors, which, in turn, is combined into a single larger vector (the SFR vector) for every slice. The SOMs are then trained in the standard fashion using the SFR vectors as training data.

After the SOM is trained, each SOM unit represents a trajectory slice (similar to a code-book vector of vector quantization); because of the topology preservation property of the SOMs, units representing similar trajectory slices denoting similar agent microbehaviors are typically grouped together on the SOM grid. This grouping yields coarse clusters indicating distinct fundamental trajectory patterns (which belong to different behavior types); however, the inner structure of these pattern clusters is also mapped to the SOM grid respecting the neighborhood structure as far as possible. It is then possible to study the trajectory projections onto the SOM grid that are generated by given players. Doing so for players from different RoboCup teams allows us to reveal significant differences in the microbehaviors of the respective players.

The paper extends the method to handle interactions between a player and the ball (a special case of a more general two-agent or two-object interaction). The SFR vectors have to be extended to the so-called enhanced spatially focused representation (ESFR) vectors, which combine two simultaneous trajectories (one player and the ball) and, thus, examine the player-ball interactions. The data representation is similar to SFR. In addition to the differences between the agent positions at successive time steps in SFR, ESFR includes the difference vectors between player and ball at the different time steps in the vector representation. The ESFR vectors are used to train the SOM. The resulting SOMs then create a map of the short-term (micro-)interaction between player and ball that is able, for example, to display significant differences in ball-handling behavior by players from different teams.

As an example, figure 14a and figure 14b show the classification for different types of combined player-ball behavior, as found by the SOM, the dots representing behavior patterns found. For example, region VI (the large central region) represents dribbling behavior where the ball is carried alongside the player. A typical representative of such behavior is shown in figure 14c, where the ball (lower path) is led by a player (upper path) to the right toward the opponent goal.

Applied to concrete players, the results for a CMUNITED-99 player are shown in figure 14b and for a MAINZ ROLLING BRAINS 1999 player in figure 14c. The different handling behaviors are clearly reflected in the significantly distinct dot patterns. The alongside dribbling (region VI) is mostly carried out by the CMUNITED-99 players, but a different type of dribbling, carrying the ball in front of the player (region VII), is predominantly performed by the MAINZ ROLLING BRAINS players. This example clearly shows one way that the SOM is able to resolve different playing styles. For further details and results of the behavior-classification method, the interested reader is referred to Wünstel et al. (2001).

Engineering Challenge Award

The navigation system is perhaps the most important subsystem of a mobile robot. In many applications, especially those concerning indoor well-structured environments, one important feature of the navigation system concerns the ability of the robot to self-localize, that is, autonomously determine its position and orientation (posture). Once a robot knows its posture, it is capable of following a preplanned virtual path or stabilizing its posture smoothly (de Wit, Siciliano, and Bastin 1996). If the robot is part of a cooperative multirobot team, it can also exchange the posture information with its teammates so that appropriate relational and organizational behaviors are established (Lima et al. 2000). In robotic soccer, these issues are crucial. If a robot knows its posture, it can move toward a desired posture (for example, facing the goal with the ball in between). It can also know its teammates' postures and prepare a pass or evaluate the game state from the team locations.

An increasing number of teams participating in RoboCup's middle-size league are approaching the self-localization problem. The proposed solutions are mainly distinguished by the types of sensor used: LRFs, vision-based omnidirectional sensors, or a single frontal camera. LRFs require walls surrounding the soccer field to acquire the field border lines and correlate them with the field rectangular shape to determine the team postures. Should the walls be removed, the method becomes inapplicable.

The winner of the Engineering Challenge Award (Marques and Lima 2001) describes an algorithm that determines the posture of a middle-size–league robot, with respect to a given coordinate system, from the observation of natural landmarks of the soccer field, such as the field lines and goals, as well as from a priori knowledge of the field geometry. Even though the intersection between the field and the walls is also currently used, the wall replacement by the corresponding field lines would not change the algorithm. The algorithm is a particular implementation of a general method applicable to other well-structured environments, also introduced in Marques and Lima (2001).

The landmarks are processed from an image taken by an omnidirectional vision system, based on a camera plus a convex mirror designed to obtain (by hardware) the soccer field bird's eye view, thus preserving the field geometry in the image. This mirror, although developed independently, was first introduced in Hicks and Bajcsy (1999). The image green-white-green color transitions over a predetermined number of circles centered with the robot are collected as the set of transition pixels. The Hough transform is applied to the set of transition pixels in a given image, using the normal representation of a line (Gonzalez and Woods 1992)

$$p = x_i^t \cdot \cos(\phi) + y_i^t \cdot \sin(\phi) \qquad (1)$$

where

$$x_i^t, y_i^t$$

are the image coordinates of transition pixel p^t and ρ, ϕ the line parameters. The q (q = 6 in this application) straight lines $(\rho_1, \phi_1), \ldots, (\rho_q, \phi_q)$ corresponding to the top q accumulator cells in Hough space are picked and, for all pairs $\{(\rho_j, \phi_j), (\rho_k, \phi_k), j, k = 1, \ldots, q, j \neq k\}$ made out of the those q straight lines, the following distances in Hough space are computed:

$$\Delta\phi = \left|\phi_j - \phi_k\right| \qquad (2)$$

$$\Delta\rho = \left|\rho_j - \rho_k\right| \qquad (3)$$

Note that a small $\Delta\phi$ denotes almost parallel straight lines, and $\Delta\rho$ is the distance between two parallel lines. The $\Delta\phi$ and $\Delta\rho$ values are subsequently classified by relevance functions, which, based on the knowledge of the field geometry, will filter out lines whose relative orientation and/or distances do not match the actual field relative orientation and/or distances. The remaining lines are correlated in Hough space with the geometric field model to obtain the robot posture estimate. An additional step must be taken to disambiguate the robot orientation. In the application to a soccer robot, the ambiguity is the result of the soccer field symmetry. The goal colors are used to remove such ambiguity.

Currently, the algorithm has been implemented in C and runs on a PENTIUM 233 megahertz with 64 megabytes of RAM in less than 0.5 seconds. It is used by each of the ISOCROB team robots to obtain their self-localization during a game after either a predetermined timeout has expired, or more than a predetermined number of bumps were sensed by the robot (figure 15). The algorithm is part of each robot's navigation system, but it is also used by the robot to share information with its teammates regarding team postures and ball location. The navigation system includes a guidance-control algorithm that relies on odometry most of the time, but odometry is reset whenever the self-localization algorithm runs.

A similar method was proposed by Iocchi and Iocchi and Nardi (2000) for soccer robots too. Their method also matches the observed field lines with a 2D field model in the Hough space. However, because only a single frontal camera is used, their approach considers lines detected locally, rather than globally, and requires odometry to remove ambiguities. The AGILO team (Schmitt et al. 2000) also proposes a vision-based approach to the self-localization problem. A single frontal camera is used to match a 3D geometric model of the field with the border lines and goals line segments in the acquired image. Only a partial field view is used in this method. Several teams use a vision-based omnidirectional hardware system but only for tracking the ball and the markings on opposing robots (Machado et al. 2000; Nakamura et al. 2000; Pagello et al. 2000).

Conclusion

RoboCup-2000 showed many advances, both in the existing competition leagues and in the introduction of several new events. The participation and attendance were greater than ever, with about 500 participants and more than 5000 spectators.

RoboCup-2001 is going to be held in the United States for the first time. It will run from 2 August through 10 August 2001 in Seattle, colocated with the Seventeenth International Joint Conference on Artificial Intelligence

(IJCAI-2001). RoboCup-2001 will include a two-day research forum with presentations of technical papers and all competition leagues: soccer simulation, RoboCup rescue simulation (for the first time), small-size robot (F180), middle-size robot (F2000), four-legged robot, and RoboCup rescue robot in conjunction with the AAAI robot competition (for the first time). It will also include a RoboCup Jr. symposium, including one-on-one robot soccer and robot dancing competitions, and other educational events for middle-school and high-school children. Finally, RoboCup-2001 will include an exhibition of humanoid robots.

For more information, please visit www.robocup.org.

Acknowledgments

RoboCup-2000 was sponsored by Sony, SGI, FujiXerox, Australon, and RMIT University. The sections of this article reflect the work of the indicated authors along with Claude Sammut, Son Bao Pham, and Darren Ibbotson (UNSW team); Tamas Nagy and Pritam Ganguly (Cornell team); Steffen Gutmann and Bernhard Nebel (CS FREIBURG team); Luis Paulo Reis (FC PORTUGAL team); Tomoichi Takahashi and the RoboCup Rescue team (rescue); Dominique Duhaut (humanoid); Michael Wünstel, Thomas Uthmann, and Jürgen Perl (scientific challenge); and Carlos Marques (engineering challenge).

References

Asada, M.; Veloso, M. M.; Tambe, M.; Noda, I.; Kitano, H.; and Kraetzschmar, G. K. 2000. Overview of RoboCup-98. *AI Magazine* 21(1): 9–19.

Augusto, O., and Carpinteiro, S. 1999. A Hierarchical Self-Organizing Map Model for Sequence Recognition. *Neural Processing Letters* 9(3): 209–220.

Baltes, J., and Hildreth, N. 2001. Adaptive Path Planner for Highly Dynamic Environments. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Castelpietra, C.; Iocchi, L.; Nardi, D.; Piaggio, M.; Scalso, A.; and Sgorbissa, A. 2001. Communication and Coordination among Heterogeneous Mid-Size Players: ART99. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Chappell, G. J., and Taylor, J. G. 1993. The Temporal Kohonen Map. *Neural Networks* 6(3): 441–445.

Coradeschi, S.; Karlsson, L.; Stone, P.; Balch, T.; Kraetzschmar, G.; and Asada, M. 2000. Overview of RoboCup-99. *AI Magazine* 21(3): 11–18.

D'Andrea, R.; Lee, J.-W.; Hoffman, A.; Samad-Yahaja, A.; Creman, L.; and Karpati, K. 2000. BIG RED: The Cornell Small-League Robot Soccer Team. In *RoboCup-99: Robot Soccer World Cup III*, 49–60. eds. M. Veloso, E. Pagello, and H. Kitano. Berlin: Springer Verlag. Forthcoming.

de Wit, C. C.; Siciliano, B.; and Bastin, G., eds. 1996. *Theory of Robot Control.* Communications and Control Engineering Series. New York: Kluwer.

Dorer, K. 1999. Behavior Networks for Continuous Domains Using Situation-Dependent Motivations. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'99), 1233–1238. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Esaki, T.; Sakushima, T.; Futamase, S.; Ito, N; Takahashi, T.; Chen, W.; and Wada, K. 2001. Kakitsubata Team Description. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Fujita, M.; Veloso, M.; Uther, W.; Asada, M.; Kitano, H.; Hugel, V.; Bonnin, P.; Bouramoue, J. C.; and Blazevic, P. 2000. Vision, Strategy, and Localization Using the Sony Legged Robots at Robocup-98. *AI Magazine* 21(1): 45–56.

Gonzalez, R., and Woods, R. 1992. *Digital Image Processing.* Reading, Mass.: Addison-Wesley.

Gutmann, J.-S.; Weigel, T.; and Nebel, B. 1999. Fast, Accurate, and Robust Self-Localization in Polygonal Environments. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS'99).* Washington, D.C.: IEEE Computer Society.

Hicks, R. A., and Bajcsy, R. 1999. Reflective Surfaces as Computational Sensors. Paper presented at the CVPR-99 Workshop on Perception for Mobile Agents, 26 June, Fort Collins, Colorado.

Iocchi, L., and Nardi, D. 2000. Self-Localization in the RoboCup Environment. In *RoboCup-99: Robot Soccer World Cup III*, eds. M. Veloso, E. Pagello, and H. Kitano, 318–330. Berlin: Springer Verlag.

James, D., and Miikkulainen, R. 1995. SARDNET: A Self-Organizing Feature Map for Sequences. In *Proceedings of the Advances in Neural Information Processing Systems, Volume 7*, eds. G. Tesauro, D. S. Touretzky, and T. K. Leen, 577–584. San Francisco, Calif.: Morgan Kaufmann.

Kaneda, T.; Matsuno, F.; Takahashi, H.; Matsui, T.; Atsumi, M.; Hatayama, M.; Tayama, K.; Chiba, R.; and Takeuchi, K. 2001. Simulator Complex for RoboCup-Rescue Simulation Project as a Test-Bed for Multi-Agent Organizational Behavior in Emergency Case of Large-Scale Disaster. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Kangas, J. 1994. On the Analysis of Pattern Sequences by Self-Organizing Maps. Ph.D. thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Kohonen, T. 1989. *Self-Organization and Associative Memory, Volume 8.* 3d. ed. Springer Series in Information Sciences. Berlin: Springer-Verlag.

Kröse, B.; Bogaard, R.; and Hietbrink, N. 2000. Programming Robots Is Fun: RoboCup Jr. 2000. Paper presented at the Belgium Netherlands AI Conference 2000, 1–2 November, Brussels, Belgium.

Kuwata, Y., and Shinjoh, A. 2001. Design of RoboCup-Rescue Viewers—Toward a Real-World Emergency System. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Lima, P.; Ventura, R.; Aparício, P.; and Custódio, L. 2000. A Functional Architecture for a Team of Fully Autonomous Cooperative Robots. In *RoboCup-99: Robot Soccer World Cup III*, eds. M. Veloso, E. Pagello, and H. Kitano, 378–389. Berlin: Springer Verlag, Berlin.

Lund, H. H. 1999. Robot Soccer in Education. *Advanced Robotics Journal* 13(8): 737–752.

Lund, H. H.; Arendt, J. A.; Fredslund, J.; and Pagliarini, L. 1999. Ola: What Goes Up, Must Fall Down. *Journal of Artificial Life and Robotics* 4(1).

Lund, H. H., and Pagliarini, L. 2000. RoboCup Jr. with Lego MINDSTORMS. In Proceedings of the International Conference on Robotics and Automation (ICRA2000). Washington, D.C.: IEEE Computer Society.

Machado, C.; Costa, I.; Sampaio, S.; and Ribeiro, F. 2000. Robot Football Team from Minho University. In RoboCup-99: Robot Soccer World Cup III, eds. M. Veloso, E. Pagello, and H. Kitano, 731–734. Berlin: Springer Verlag.

Maes, P. 1990. Situated Agents Can Have Goals. In *Designing Autonomous Agents: Theory and Practice from Biology to Engineering* *and Back*, ed. P. Maes, 49–70. Cambridge, Mass.: MIT Press.

Marques, C., and Lima, P. 2001. A Localization Method for a Soccer Robot Using a Vision-Based Omni-Directional Sensor. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Mehler, F. 1994. Selbstorganisierende Karten in Spracherkennungssystemen. Dissertation, Institut fur Informatik, Johannes Gutenberg-Universitt Mainz.

Murray, J.; Obst, O.; and Stolzenburg, F. 2001. Toward a Logical Approach for Soccer Agents Engineering. In RoboCup-2000: Robot Soccer WorldCup IV, eds. G. Kraetzschmar, P. Stone, and T. Balch. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Nakamura, T.; Terada, K.; Takeda, H.; Ebina, A.; and Fujiwara, A. 2000. Team Description of the RoboCup NAIST. In *RoboCup-99: Robot Soccer World Cup III,* eds. M. Veloso, E. Pagello, and H. Kitano, 727–730. Berlin: Springer Verlag.

Noda, I.; Matsubara, H.; Hiraki, K.; and Frank, I. 1998. SOCCER SERVER: A Tool for Research on Multiagent Systems. *Applied Artificial Intelligence* 12(3): 233–250.

Noda, I.; Suzuki, S.; Matsubara, H.; Asada, M.; and Kitano, H. 1998. RoboCup-97: The First Robot World Cup Soccer Games and Conferences. *AI Magazine* 19(3): 49–59.

Ohta, M. 2001. RoboCup Rescue Simulation: In Case of Fire Fighting Planning. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Pagello, E.; Piaggio, M.; Chella, A.; Clemente, G.; Cassinis, R.; Bonarini, A.; Adorni, G.; and Nardi, D. 2000. Azzurra Robot Team—ART. In *RoboCup-99: Robot Soccer World Cup III,* eds. M. Veloso, E. Pagello, and H. Kitano, 695–698. Berlin: Springer Verlag.

Prokopenko, M.; Butler, M.; and Howard, T. 2001. On Emergence of Scalable Tactical and Strategic Behavior. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetszchmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Reis, L. P., and Lau, N. 2001. FC PORTUGAL Team Description: RoboCup-2000 Simulation-League Champion. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Riedmiller, M.; Merke, A.; Meier, D.; Hoffmann, A.; Sinner, A.; Thate, O.; and Ehrmann, R. 2001. Karlsruhe BRAINSTORMERS—A Reinforcement Learning Approach to Robotic Soccer. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Ritter, H.; Martinetz, T.; and Schulten, K. 1992. *Neural Computation and Self-Organizing Maps: An Introduction.* Reading, Mass.: Addison-Wesley.

Scmitt, T.; Klupsch, M.; Hanek, R.; and Bandlow, T. 2000. AGILO ROBOCUPPERS: RoboCup Team Description. In *RoboCup-99: Robot Soccer World Cup III,* eds. M. Veloso, E. Pagello, and H. Kitano, 691–694. Berlin: Springer Verlag.

Stone, P. 1998. Layered Learning in Multiagent systems. Ph.D. dissertation, Computer Science Department, Carnegie Mellon University.

Stone, P., and Veloso, M. 1999. Task Decomposition, Dynamic Role Assignment, and Low-Bandwidth Communication for Real-Time Strategic Teamwork. *Artificial Intelligence* 110(2): 241–273.

Stone, P.; Riley, P.; and Veloso, M. 1999. CMUNITED-99 Source Code. Available at www.cs.cmu.edu/ pstone/RoboCup/CMUnited99-sim.html.

Stone, P.; Riley, P.; and Veloso, M. 1999. The CMUNITED-98 Champion Simulator Team. In *RoboCup-98: Robot Soccer World Cup II,* eds. M. Asada and H. Kitano, 61–76. Lecture Notes in Artificial Intelligence. Berlin: Springer Verlag.

Tadokoro, S., and Kitano, H., eds. 2000. *The RoboCup Rescue: A Challenge for Emergency Search and Rescue at Large-Scale Disasters.* Tokyo: Kyoritsu.

Tadokoro, S.; Kitano, H.; Takahashi, T.; Noda, I.; Matsubara, H.; Shinjoh, A.; Koto, T.; Takeuchi, I.; Takahashi, H.; Matsuno, F.; Hatayama, M.; Nobe, J.; and Shimada, S. 2000. The RoboCup Rescue Project: A Robotic Approach to the Disaster-Mitigation Problem. In Proceedings of the IEEE International Conference on Robotics and Automation. Washington, D.C.: IEEE Computer Society.

Takahashi, T.; Takeuchi, I.; Koto, T.; Tadokoro, S.; and Noda, I. 2001. RoboCup Rescue Disaster Simulator Architecture. In RoboCup-2000: Robot Soccer World Cup IV, eds. P. Stone, T. Balch, and G. Kraetzschmar. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Weigel, T.; Dietl, M.; Dümler, B.; Gutmann, J.-S.; Marko, K.; Müller, K.; Nebel, B.; and Thiel, M. 2001. The CS FREIBURG Team. In RoboCup-2000: Robot Soccer World Cup IV, eds. T. Balch, P. Stone, G. Kraetzschmar, and H. Kitano. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

Wünstel, M.; Polani, D.; Uthmann, T.; and Perl, J. 2001. Behavior Classification with Self-Organizing Maps. In RoboCup-2000: Robot Soccer World Cup IV, eds. T. Balch, P. Stone, G. Kraetzschmar, and H. Kitano. Lecture Notes in AI 2019. Berlin: Springer Verlag. Forthcoming.

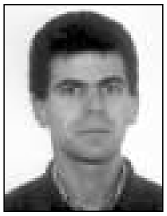
Yamamoto, T., and Fujita, M. 2000. A Quadruped Robot Platform with Basic Software for RoboCup-99 Legged-Robot League. Paper presented at the International Conference on Intelligent Robots and Systems 2000 (IROS-00), 30 October–5 November, Takamatsu, Japan.

Peter Stone is a senior technical staff member in the Artificial Intelligence Principles Research Department at AT&T Labs Research. He received his Ph.D. in 1998 and his M.S. in 1995 from Carnegie Mellon University, both in computer science. He received his B.S. in mathematics from the University of Chicago in 1993. Stone's research interests include planning and machine learning, particularly in multiagent systems. His doctoral thesis research contributed a flexible multiagent team structure and multiagent machine learning techniques for teams operating in real-time noisy environments in the presence of both teammates and adversaries. He is currently continuing his investigation of multiagent learning at AT&T Labs.

Thilo Weigel is the team leader of the CS FREIBURG team. He received a diploma in computer science from the University of Freiburg, Germany, in 1999. Currently, he is a Ph.D. student in the Artificial Intelligence Research Group at the University of Freiburg. His research interests include mobile robot self-localization, multiagent systems, and robotic soccer. His e-mail address is weigel@informativeness-freiburg.de.

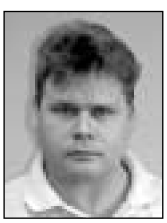
Daniel Polani obtained his diploma in physics and his Ph.D. in computer science from the University of Mainz, Germany. Currently, he is a senior researcher at the Institute for Neuro- and Bioinformatics at the University of Lübeck, Germany. His research interests are multiagent systems, neural networks, and genetic algorithms.

Nuno Lau is an assistant researcher at

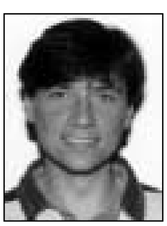


Aveiro University, Portugal, where he is currently a Ph.D. student in electrical engineering. He received his engineering degree from Oporto University in 1993 and a DEA degree in biomedical engineering from Claude Barnyard University, Lyon, France, in 1994. His research interests include reconfigurable systems, robotics, AI, and computer science.

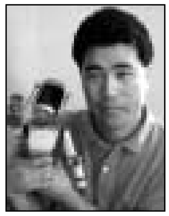
Minoru Asada received a B.E., an M.Sc., and a Ph.D. in control engineering from Osaka University, Osaka, Japan, in 1977, 1979, and 1982, respectively. Since April 1997, he has been a professor in the Department of Adaptive Machine Systems at the same university. He received the 1992 Best Paper Award at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS92) and the 1996 Best Paper Award of the Robotics Society of Japan (RSJ). He was a general chair of IEEE/RSJ 1996 International Conference on Intelligent Robots and Systems (IROS96). Since early 1990, he has been involved in RoboCup activities, and his team was the cochampion with the team from the University of Southern California in the middle-size league at the first RoboCup held in conjunction with IJCAI-97 in Nagoya, Japan.



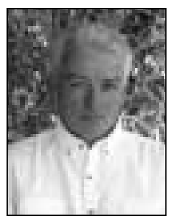
Tucker Balch builds robots at the Robotics Institute of Carnegie Mellon University. He received a B.S. and a Ph.D. in computer science from the Georgia Institute of Technology in 1984 and 1998, respectively. From 1988 to 1995 he flew F-15s in the United States Air Force. Balch's research focuses on the challenges of building and understanding large-scale multiagent systems.



Raffaello D'Andrea received a B.A.Sc. in engineering physics from the University of Toronto and an M.S. and a Ph.D. in electrical engineering from the California Institute of Technology. He has been with the Department of Mechanical and Aerospace Engineering at Cornell University since 1997, where he is an assistant professor. He is also a member of the Applied Mathematics, Electrical Engineering, and Theoretical and Applied Mechanics departments at Cornell University. In addition to several teaching awards at Cornell University, he is a recipient of the 1995 American Control Council O. Hugo Schuck Best Paper Award, the 1996 IEEE Conference on Decision and Control Best Student Paper Award, and a 2000 National Science Foundation CAREER award. His research interests include the development of computational tools for the robust control of complex interconnected systems.



Masahiro Fujita is a system architect and a chief research scientist at Digital Creatures Laboratory, Sony Corporation. He received a B.A. in electronics and communications from Waseda University, Tokyo, in 1981 and an M.S. in electrical engineering from the University of California at Irvine in 1989. His research interests include modular robotics, agent architecture, visual and audio perception, neural networks, and evolutionary systems.



Bernhard Hengst's research interests are machine learning and autonomous agents. He is currently completing his Ph.D. in hierarchical reinforcement learning at the University of New South Wales, School of Computer Science and Engineering, Australia. For the last 25 years, he has worked in private industry, initially in operations research in Australia and the United Kingdom and later as general manager with ACI Australia, Ferntree Computer Corporation, and GE Capital.

Gerhard Kraetzschmar is a senior research scientist and assistant professor at the Neural Information Processing Department at the University of Ulm. His research interests interests include a wide range of fields in robotics, neural and evolutionary computation, and (distributed) AI, including autonomous mobile robots, neurosymbolic integration, learning and memory, and robot control and agent architectures. He is a member of AAAI, ACM, IEEE CS, IEEE RAS, and GI.

Pedro Lima received his Licenciatura (5 years) and his M.Sc. in 1984 and 1989, respectively, both in electrical and computer engineering, from the Instituto Superior Técnico (IST), Lisbon Technical University, Portugal, and his Ph.D. (1994) in electrical engineering from the Rensselaer Polytechnic Institute. Since 1994, he has been an assistant professor at IST. He is also a member of the Institute for Systems and Robotics, a Portuguese private research institution, where he is the coordinator of the Intelligent Control Group and a member of the Scientific Council. In 2001, Lima will be the chair of the RoboCup-2001 Middle-Size League.

Henrik Hautop Lund is a professor in sensor and actuator technology at the Maersk Institute at the University of Southern Denmark, Odense. He holds a Ph.D. in computer systems engineering, an MSc in computer science, and a BSc in mathematics. In 1997, Lund founded the LEGO Lab, and he is occupied with development of LEGO MINDSTORMS robots and how to put AI into our daily lives. His research interests include biologically inspired robotics, evolutionary robotics, user-guided adaptive behavior, and the relationship between robot behavior and robot morphology.

Paul Scerri is a graduate student at Linköpings Universitet in Sweden. His research interests include adjustable autonomy, engineering intelligent agents, and real-time systems.

Satoshi Tadokoro received a B.E. in precision machinery engineering in 1982, an M.E. in precision machinery engineering in 1984 from the University of Tokyo, and a D.E. in system science in 1991 from Kobe University. From 1984 to 1993, he was a research associate in the Department of Instrumentation Engineering, Kobe University, and has been an associate professor in the Department of Computer and Systems Engineering since 1993. His research interest is in rescue robotics and molluscan robotics. He received the Hydraulic and Pneumatic Technology Foundation Award in 1993, the Japan Society of Artificial Intelligence Incentive Award in 2000, and the Robotics Symposia Best Paper Award in 2000.