Borregos
RoboCup Standard Platform League 2010
Team Description Paper

David García, Efrén Carbajal, Carlos Quintana, Ernesto Torres, Iván Gonzalez, Carlos Bustamante, and Leonardo Garrido

Department of Computer Science, Tecnológico de Monterrey, Campus Monterrey; Monterrey, NL, México,
`daalgaor@gmail.com`, `efren.ce@gmail.com`, `caqs44@gmail.com`, `nigm2005@gmail.com`,
`skullmate@gmail.com`, `cfbh@itesm.mx`, `leonardo.garrido@itesm.mx`,
WWW home page: `http://robocup.mty.itesm.mx/`

Abstract. This paper shows the description of our team Borregos from the Tecnológico de Monterrey. Borregos team has been participating in the RoboCup competitions since 2004, starting in the 2D Simulation League and moving forward to the 3D Simulation League with spheres in 2006, the 3D Simulation League with humanoids in 2007 and the RobotStadium Simulation League since 2008. Besides this year we started a new team "Borregos-Humanoids" which participated in the 2nd. Mexican RoboCup Open. Finally, this year Borregos team has been accepted to participate in the Standard Platform League and 3D Simulation League in the RoboCup Singapore 2010 competitions. In the last years, we have been working primarily with the simulation of the SPL, the RobotStadium competition based on the Webots simulator. We have achieved very good results developing the team and in the competition. Currently, we are working on the development of our humanoids team for the RoboCup Standard Platform League with real Nao robots.

1 Introduction

RoboCup currently includes a number of different robot soccer leagues that focus on different research challenges. Soccer game is the central topic of research, a very interesting and exciting game that incite researchers to create innovative solutions that can be applied to socially significant problems and industries. One of the soccer leagues is the RoboCup Standard Platform League (SPL), in which all teams compete with identical models (Nao robots from Aldebaran Robotics) so they can focus only on software development, representing a great area of opportunity for those who want to do research about intelligent agents in realistic environments.

The goal of RoboCup is that by 2050, develop a team of fully autonomous humanoid robots that can win against the human world soccer champion team. Fulfilling this goal is not an easy task; several methodologies have to be proposed in areas such as robotics, electronics, computer science and artificial intelligence (AI). AI is a branch of computer science which studies and designs intelligent agents, where an intelligent agent is a system that perceives its environment and takes actions which maximize its chances of success. The SPL is an effort to create a realistic environment providing a standard two-legged robot for all teams in which AI techniques and algorithms can be tested.

Borregos team is part of the Intelligent Autonomous Agents research group from the Computer Science Department at Tecnológico de Monterrey, Campus Monterrey University, which objective is to develop innovative technology oriented towards distributed knowledge handling by researching about agent technologies and multiagent systems. In the RoboCup domain, the research interest is coordination and cooperation among autonomous agents.

For Borregos team, RoboCup SPL represents an excellent opportunity area to make AI research, specially in multiagent systems and dynamic environments. The team is interested in research about intelligent agents with long life, autonomy, reaction capacities, planning, learning and reasoning. The main research goal for Borregos team is to obtain emergent team behaviors through multiagent planning, coordination and decision making research. In the 3D and RobotStadium simulation leagues, our Borregos team has done a lot of work creating basic low-level behaviors like walking, turning, standing up, kicking, localization and communication. Now the team is focused on research about high-level abilities like team coordination, planning and learning.

This is the first time Borregos team participate in the RoboCup SPL competitions and also this is our first Team Description Paper (TDP). Section 2 describes the Borregos team constitution. Section 3 shows the Borregos team background. Section 4 shows our current software architecture. Sections 5, 6, 7 and 8 detail every software module. Section 9 describes the main roles of the players and their strategies. Finally, conclusions are shown.

2 Team constitution

Currently, our team is mainly formed by students in our M.Sc. in Intelligent Systems and a Ph.D. in Artificial Intelligence here at Tecnológico de Monterrey, Campus Monterrey. David García holds a B.S. in Computer Science and is enrolled in our Master program. He has experience programming in Java and C++ and has worked in the development of the team for the RobotStadium competition and 3D League using the simulated Nao robot. Efrén Carbajal holds a B.S. in Industrial and Systems Engineering and is also enrolled in the Master program. He has been working in the development of the team for RobotStadium competition. Carlos Bustamante has already graduated from our Master program but is still collaborating with us. His thesis [1] explains a layered approach for developing agents for the RoboCup 3D simulation league, from the physics models up to the probabilistic localization and situation evaluation. Iván Gonzalez holds a B.S. in Mechatronical Engineering and has experience in mechanisms, robotics and programming languages. Carlos Quintana and Ernesto Torres hold a B.S. in Electronic Systems, both have specialization in electronics devices and are skilled in several programming languages. They are new team members and started their participation developing the humanoids Borregos team based on the RoboNova robot from Hitec Robotics. Ph.D. Leonardo Garrido is the leader of our Borregos team and head of the Intelligent Autonomous Agents Research Group of the Computer Science Department at Tecnológico de Monterrey, Campus Monterrey. He has several publications related to probabilistic learning, case based reasoning, multiagent systems, robotics and computer simulations.

3 Team Background

Our team Borregos was created back in 2004 and participated in the RoboCup Portugal competitions in that year in the Soccer Simulation 2D category with M.Sc. Emmanuel Martinez as the team leader, under the supervision of Ph.D. Ramon Brena [7]. Emmanuel, in his master's thesis, proposed a new technique using a search algorithm based in game theory. His method consists of selecting the best action in step T of the simulation, based on the evaluation of the sequence of possible actions in step $T + 1$, creating a decision tree over which the search is performed.

In 2005, M.Sc. Carlos Bustamante started developing a new team for the simulation 3D league with spherical agents and was classified to participate in the RoboCup Germany international competitions in the city of Bremen, under the supervision of Ph.D. Leonardo Garrido. A fuzzy bayesian approach for decision making in RoboCup Simulation 3D was presented in the RoboCup Symposium [6]. Later, a comparison between fuzzy bayesian classifiers and gaussian bayes classifiers was published in [5]. Another student, Cesar Flores, contributed developing the physics models which were used in the goto and dribble behaviors of the agents. This approach was published in [2] and later in a book chapter [4].

Afterwards, a hybrid monte carlo localization with Kalman filter sensor fusion approach was used for diminishing the effect of noise and uncertainty in the agent self localization process, and was published in [3]. With this approach, Borregos3D participated in the RoboCup Brazil Open 2006 competitions and won the third place.

In July 2007, we were TOP 16 in the first humanoid simulation competition in the RoboCup USA international competitions in the city of Atlanta, in which humanoid robot models (based on the Fujitsu HOAP-2 robot) were used for the first time. In November 2007, we were the organizers of the 3D simulation league in the 3rd. RoboCup Latin American Open celebrated in Monterrey, México.

In 2008, Borregos team participated in the RoboCup China international competitions in the city of Suzhou, being one of the TOP 8 teams of the simulation 3D league and joining the first RobotStadium online robot soccer competition, a new simulation league using the Webots simulator by Cyberbotics/Gostai, and the URBI (Universal Real-time Behavior Interface) programming language for the scripting of controllers.

During the first half of 2009, Borregos team participated in the RobotStadium competition with a refactored controller in Java, remaining in the first place for much of the competition and finishing as fourth in the last round. During this competition a lot of work was done and the team became one of the strongest in the RobotStadium League. The main features of the team were: good basic behaviors (walk, turn, stand up, shoot), robot coordination through communication, landmark based auto-localization and a high quality vision module for detecting the ball and the goals.

In 2009, the team participated in the 3D simulation league in RoboCup Austria competitions. The team finished at the middle of the ranking. Borregos team had good performance in terms of basic skills and its main characteristic for this competition was the adaptation of agents to the new restricted vision rule in the league.

Also in 2009, in order to participate in the 2nd. Mexican RoboCup Open 2009, within the category of humanoid robots, we started to develop our Borregos-Humanoids team using the RoboNova robot. The humanoids team was developed with basic behaviors like walking, turning, shooting, standing up as well as searching and tracking the ball.

Now, our next research goal is to complement all our research work in the simulation field with the development of a humanoids team in a more realistic domain such as the RoboCup SPL with real Nao robots.

4 Software Architecture

Our team has an extensive experience in the simulation leagues of RoboCup. We have developed a fully featured team for the simulated Nao robots in the RobotStadium online competition as well as for the RoboCup 3D Simulation League. Therefore we already have many algorithms developed in Java for the basic behaviors. Our main features are: basic behaviors, auto-localization, vision module and also some high-level abilities like coordination through communication.

Our main goal is to translate all the algorithms generated in simulation to the SPL mainly to experiment in a more realistic environment and also to generate new robotic research about intelligent agents, complementing it with our simulation research. With our experience on the 3D and RobotStadium simulation leagues, we are adapting our code to the real Nao robots in order to create a high competitive SPL team.

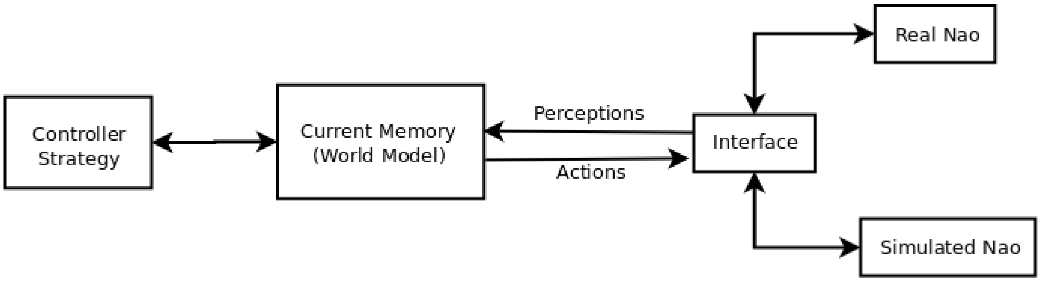

Fig. 1. Borregos robot controller.

Currently we are translating all our RobotStadium modules in Java language to naoqi modules in C/C++ language for the real Nao robot. However, our intention is to develop a unique C/C++ robot controller for both the SPL and the RobotStadium simulation (see figure 1). Our controller will be able to interact with a real robot and with a simulated Nao robot trough a interface. In this way the logic of the controller remains for both platforms and also is more robust in terms of software updates, since just the interface should be adapted to new versions of the simulator and the Aldebaran SDK.

The controller is composed by 4 main modules: vision, motion, localization and communication. Figure 2 shows the overall software architecture of the Borregos team. Controller has an abstract module called Player which interacts with the 4 main modules. Player module is responsible for the synchronization between the controllers and the running platform (Nao or Webots). Also Player module implements the main controller cycle: Sense, Think and Act. The two main role modules implemented in the team, the goal keeper and the field player are inheriting this abstract module. These roles implement the strategies for playing soccer. They contain the decision making algorithms and control the behavior of the robots.

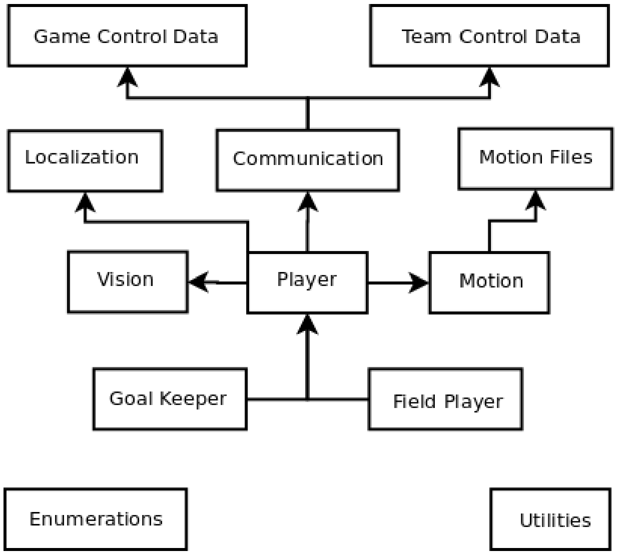

Fig. 2. Borregos software architecture.

Similarly, the Motion module interacts with all the motion files (section 6) and the Communication module interacts with the module for team communication and the module for supervisor communication (section 8). Finally, there are 2 global modules: Utilities and Enumerations. The enumerations module contains data structures with static information like filed dimensions, global coordinates, etc. The utilities module contains methods that facilitate data processing. Following sections detail the 4 main modules.

5 Vision

Ball, goals and robots of each team have a different color, to ease their recognition. However, the light conditions are continuously adjusted which influence objects way to shines. Besides vision extract information of a very simple camera: 160x120 pixels color (RGB) camera 25 fps.

Vision module performs recognition of objects like ball and goals, which allowed robots to gain precision to positioning in direction to enemy goal, and broaden internal state. Our vision module follows the process showed in figure 3. The process starts obtaining a raw image from the camera device. Then a RGB matrix is extracted from the input image and the color classification process is performed. As output of color classification we have an object map which is a matrix of the same size as the RGB matrix, but instead of RGB values it contains object labels. Immediately the pattern recognition process takes the object map and tries to find flags and then construct a vector for every recognized flag. Vector contains distance, vertical angle and horizontal angle to flags. Finally vectors are available to other modules through an interface (public getter methods).

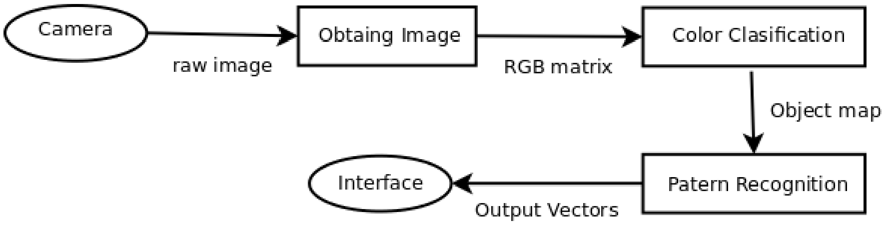

Fig. 3. Borregos Vision module process.

5.1 Color classification

It is essential in the vision system to obtain information of colors to be able to identify objects. Color classification is the process that maps pixel values of an image to a color label that corresponds to a region of color space pre-defined in a look up table (LUT). A hard color classification is used, due to different object are indicated by different colors, even when objects color perception change because of light and shadows. We do not have to classify every pixel object of image, instead we classified those as unknown color, in this way, we do not exclude any indispensable color and, at the same time, keep away from overlapping color space problem.

5.2 Pattern recognition

Basically patterns are used to obtain more information about the environment, particularly in this project, to know more regarding goals. The way to proceed is, once all relevant pixels are classified as objects, a second analysis of this new array of objects is needed to be done, in order to classify patterns. Within the Vision module, the main flags identified are the goals corners.

Goals colors are either yellow or sky blue, then goals are first detected by color, whenever a robot recognize those colors, it knows there is a goal somewhere in front, and we mean for somewhere, that distance and direction is unknown so far. Depending if robots are seeing or not goal post bases (recognizing them), could know this information. Our way to address this issue is that, once color is detected, a scan is run looking for post bases.

5.3 Vector Calculations

The raw values of ball distance, direction and elevation are calculated based on the supposition that ball is always over the floor, which is generally true. This is done by calculating the centroid (average x and y positions) of all pixels corresponding to the ball in the image and dividing them over the image width, then multiplying the resulting calculation by the fill of view (FOV) of the camera. Just as is shown in the following equations:

$$ballDirectionAngle = (xball/numBallPixels/width - 0.5) * horizontalfov$$
$$ballElevationAngle = (yball/numBallPixels/height - 0.5) * verticalfov$$

Now, depending of the estimation required and knowing the ball and Nao measures, and the position of yaw and pitch servos, it is possible to compute distance and direction and elevation angles as showed in following equations:

$ballDirectionAngle = ballDirectionAngle - yaw$
$ballElevationAngle = ballElevationAngle - pitch - cameraPos(high - low)$
$ballDistance = (robot_height - ball_radius)/Math.tan(-ballElev)$

In case of goals calculation, a very similar approach is used.

6 Motion

Nao robot is build with servo motors which can be set to a particular angle that can be measured in degrees or radians. Each servo in the robot will add 1- degree of freedom to the robot. Our current Nao model includes 22 DOFs: Head (2) + Shoulder (4) + Elbow (4) + Hip (6) + Knee (2) + Ankle (4).

There are two ways for building motions, the first one is using hard coded sequences and the other one is directly controlling via an API the movement of each of the servo motors. Both methods include advantages and disadvantages. Hard coded sequences can be programmed by recording positions from the robot and later on call this generated motion whenever is required. Also a step by step construction can be used, given a time line, describe the robot position at each of the time intervals.

By using hard coded sequences we have built the basic motions that allow our robot to move around the environment. We have put a lot of effort on improving the speed and the quality of basic behaviors, these include: turning left or right, kicking the ball, stand up motions depending whether falling on its back or its front, circling the ball to align the robot for a clean shot at the goals, and finally, a fast and dynamically stable walking. Although this motions are hard coded sequences they have been proved resourceful for our robot by giving it the ability to move properly around the environment.

Directly controlling sequences via an API gives extra flexibility by allowing to change dynamically the position of the robot according to its environment. This type of motion controlling seams to be the best way but has the drawback of being more difficult to implement and requires a higher computational time that should be taken in consideration when used in conjunction with other heavy processing algorithms. Currently we don't use this type of sequence control, but we plan to implement it in a near future.

We are currently researching in machine learning algorithms focuses on continuous domains. This will allow smarted and optimized sequences to be generated.

7 Localization

Self-localization is necessary for tasks like positioning, finding the ball, finding the goal and aiming. Also it is very important to identify teammate positions and locate the opponents. This is a hard problem because it requires good quality data from the vision module and some complex calculus. The localization module must provide a 3D vector with the robot's global position on the field. Localization module uses the output vectors from the vision module corresponding to the field flags like the goal posts and corners. These vectors contain relative localization information and by calculus we determine vectors for the not seen flags (see figure 4). Finally, given the relative vectors to the field flags, the localization module generates the translation vector and the rotation matrix of the robot.

8 Communication

Our communication module is in charge of managing the sending and receiving messages with both the supervisor and with other robots of the team. Every time robot controllers send own information to their teammates. Every robot share its localization information for strategy proposes. We have implemented two main type of messages: messages from goal keeper and messages from field player.

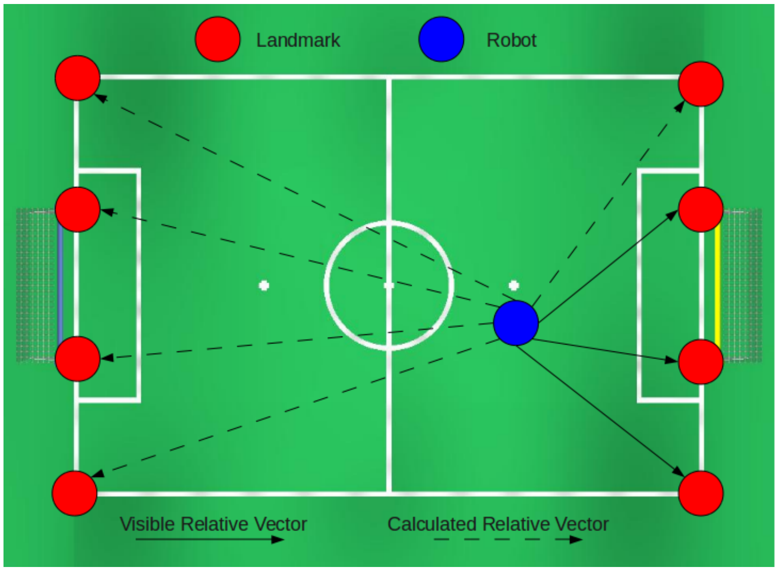

Fig. 4. Borregos Localization module process.

Moreover, every time the controller checks for available messages in the queue of the receptor device. Then it is verified if the message comes from the supervisor or a teammate through the reading of a message header. If the message has no a valid header it is discarded. For messages from the supervisor, the controller updates its internal state with the RoboCup game control data (current game state, score, time, etc). For messages from teammates the controller checks if the info comes from the goal keeper or a field player and then updates its internal state with the localization info from the teammates.

9 Strategy

Our controller has two main roles, each one with a specific game strategy. For the goal keeper the strategy is very simple. The following is the main cycle strategy for the goal keeper:

1. Search ball.
2. Calculate direction and distance to ball
3. If ball is far, then go to the center of goal.
4. If ball is close and straight, then wait.
5. If ball is close, not moving and straight, then go to kick it.
6. If ball is close and not straight, then walk sideways.

For the filed player the main cycle strategy is:

1. Search ball.
2. Calculate direction and distance to ball
3. Compare localization information with teammate localization.
4. Negotiate if should go to the ball or wait.
5. If angle to ball is big, then turn to the ball direction.
6. If distance to ball is big, walk to the ball.
7. If angle an distance to ball are small, then surround ball while angle to goal is big.
8. If angle an distance to ball are small and angle to goal is small, then shoot.

Also both strategies are constantly tracking the ball and checking for a stand up position. The described strategies are for the normal play game state, but with a few changes also work for the penalty kick game state.

Currently, our controller use the shared teammate information for synchronization proposes. We use that information to prevent our robots to collide between them when they are going to the ball. Also we are trying a better positioning of the robots within the field.

Nonetheless, we are currently working on a emergent dynamic coordination mechanism, a Swarm Intelligence technique called Division of labor, which will allow construct an adaptive team behavior. Also we are researching about connectionist reinforcement learning in order to achieve automatic learning of basic behaviors and also learning policies for decision making. Finally we have the intention to extend the reinforcement learning approach to a multiagent learning environment in order to achieve team abilities like passes.

The individual skills of robots are among the top factors to focus on when developing a competitive team, that team behavior is not quite advance. Therefore, this is an aspect in which we are constantly working on.

10 Conclusions

Borregos team has been actively participating in the RoboCup competitions since 2004. We have several publications related to RoboCup simulation and we have experience working with the simulated Nao robot in the RoboCup Soccer Simulation League and the RobotStadium competition. Our goal is to develop a very competitive team for the RoboCup Standard Platform League, in order to migrate all our research results on the simulation domain to a more realistic one, including probabilistic localization and fuzzy bayesian decision making. We also want to try reinforcement learning methods in continuous environments, efficient vision algorithms which cannot be tested with good accuracy on simulations and coordination and cooperation of multiple robots using communication and swarm intelligence techniques.

References

1. Carlos Bustamante. Probabilistic agent localization and fuzzy bayesian pass evaluation for the robocup simulation 3d league. Master's thesis, Tecnológico de Monterrey, Monterrey, Mexico, December 2007.
2. Carlos Bustamante, Cesar Flores, and Leonardo Garrido. A physics model for the robocup 3d soccer simulation. In *Proceedings of Agent-Directed Simulation Symposium (ADS 07)*, Norfolk, VA, USA, March 2007.
3. Carlos Bustamante and Leonardo Garrido. Monte-carlo localization with kalman filter sensor fusion. In *Proceedings of the 4th IEEE Latin American Robotic Symposium*, Monterrey, Mexico, November 2007.
4. Carlos Bustamante and Leonardo Garrido. *Soccer Robotics*, chapter Probabilistic and Statistical Layered Approach for High-Level Decision Making in Soccer Simulation Robotics. I-Tech Education and Publishing, Austria, 2007.
5. Carlos Bustamante, Leonardo Garrido, and Rogelio Soto. Comparing fuzzy naive bayes and gaussian naive bayes for decision making in robocup 3d. In *Proceedings of the 5th. Mexican International Conference on Artificial Intelligence (MICAI 06)*, pages 237–247, Apizaco, Tlaxcala, Mexico, November 2006. ISBN 3540490264.
6. Carlos Bustamante, Leonardo Garrido, and Rogelio Soto. Fuzzy naive bayesian classification in robosoccer 3d: A hybrid approach to decision making. In *Proceedings of the RoboCup International Symposium*, Bremen, Germany, 2006.
7. Emmanuel Martínez and Ramón Brena. Maximizing future options: An on-line real-time planning method. In *Proceedings of the 4th. Mexican International Conference on Artificial Intelligence (MICAI 05)*, Lecture Notes on Artificial Intelligence, Monterrey, NL, Mexico, November 2005. Springer-Verlag.