FURGBOL 2010 Team Description Paper

André F. Núñez, Felipe C. Moraes, João C. Longhi, Luan Silveira, Lucas Novelo, Gisele Simas, Emanuel Estrada, and Silvia S. C. Botelho

Abstract— **The present paper describes the FURGBOL robot F-180 team. The FURGBOL RoboCup team uses inexpensive and easily extendible hardware components and a standard linux software environment. We propose a modular architecture, having three main stages: *i.* a Deliberative Stage (associated with strategy and path planning issues), *ii.* a Communication Stage and, *iii.* a Embedded Reactive Control. We describes the relevant aspects of our architecture like software, hardware and design issues.**

I. INTRODUCTION

The field of multi-robot systems has become enlarged [1]. RoboCup is a long-term effort of the academic and industrial research community to develop teams of robotic football/soccer players. Issues associated with accurated motion, team coordination, communication, embbeded systems must be treated, regarding real time restrictions. [2], [3], [4], [5].

Several important approaches propose to build sophisticated multi-robot teams through the combination of expensive and complex hardware and mechanical devices [2], [3], [4], [5].

From an educational perspective, the RoboCup Competitions is a great motivation for exposing students to design, build, manage, and maintain complex robotic systems. However, nowadays, how to participate of a RoboCup Competition with a very limited budget, bringing together recent state-of-art robotic concepts? Is it possible to implement good solutions and sophisticated design methodologies with low cost robotic and sensors platforms? The leap from theory to robotic implementation is often difficult to do, and to do well or efficiently, even more difficult.

The FURGBOL F-180 Team is an effort of the Center for Computational Science of the Universidade Federal do Rio Grande, Brazil. Our goal is to stimulate research, teaching, and applications in the fields of artificial intelligence and collaborative robotics. FURGBOL group is composed by students. Our team use inexpensive and easily extendible hardware components and a standard linux software environment. Even a very limited budget, FURGBOL has show to be a relatively successful approach; since it started, in 2001, we are six times champion of Brazilian Robocup, vice-champion of Latin American Robocup twice, and FURGBOL wins the Latin American Robocup once.

This paper describes a set of issues associated with our F-180 Robocup Team. In section 2, we introduce our architecture compose by three main stages: Embedded Reactive Control, Communication and Deliberative Stages. Next sections detail each one of these stages. Finally, we present our implemented system which illustrates the principal aspects of our contribution.

Federal University of Rio Grande - FURG - Center for Computational Science C3 - Rio Grande - RS - Brazil - email:silviacb@furg.br

II. AN OVERVIEW OF OUR TEAM

The idea is to have an omnidirectional team to play soccer. Our robots uses omnidirectional wheels, and each wheel has its own motor. In this way each motor needs an independent control and imposes a force in one from the two possible directions. The resulting force composed by the forces (from each wheel) moves the robot towards the desired direction.

The chassis consists of one laser-cut aluminium plate, having a diameter of 176mm and a height of 145mm, see figure 1.

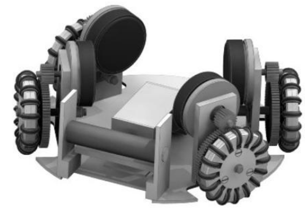

Fig. 1. The CAD chassis of FURGBOL robot.

Starting from the **Plan-Merging** Paradigm for coordinated resource utilization - and the **M+ Negotiation for Task Allocation - M+NTA** for distributed task allocation, we have developed a generic architecture for multi-robot cooperation [6]. This architecture is based on a combination of a local individual reactive control and a central coordinated decision for incremental plan

A Centralized Deliberative System is in charge of the path planning and the robot behavior. The communication system exchanges informations between robots and Central Station (CS). This stage receives the high level global information from CS, reacting to local environmental changes. Next Sections detail each one of the architecture stages.

III. A DELIBERATIVE CENTRAL STAGE

It is assumed that robots and ball are agents. A state machine is associated with each agent. A Central Deliberative System perceives the environment (agent states) and plans actions and tasks associated with each member team.

The planning module is based on a world model which models the state of each agent in the game, like in [7]. We use a set of state machines whose nodes are related to the state of the players and ball; and the transitions are given in function of the dynamics of the game.

A. A Perception Step

This step transforms position and velocity information into states associated with each agent. A set of states and transitions (actions) are defined based on the relative positions between robots and ball.

In addition, the robots and ball will assume topological labels, called areas, that identify their localization inside the field: for x axis (that joins both goals) they might be either in defense areas, halfway or attack; for y axis (perpendicular to the x): left side line, right side line and halfway.

B. A Role Assignment Step

With the ball state and topological labels already defined, the Planning Module calculates a set of actions to be achieved by each team member. Three kinds of roles are defined: the goal-keeper, the defenser and the attacker. Each role has a own state machine.

Each one of these three basic roles supplies a target position to where each robot must move itself. In addition, the planning module decides when a robot should rotate or activate the kicker and dribbler devices. These actions happen when the robot is close enough of the ball to lead or kick it. In order to dribble or to kick, the robot must turn to position the devices in front of the ball. The robot spins if the angle between its front and the correct position is less than a constant.

C. A Path Planning Step

The Path Planning Step defines the robot motion to arrive to target position avoiding obstacles.

Thus, it is first checked whether a straight-line trajectory to the target position is possible without any collision. Then the trajectory and obstacles in the field are regarded as polygons. If one of the vertices of any polygon that represents an obstacle is inside the trajectory polygon, there will be a collision.

If the straight-line path is not possible, we apply the approximated cell decomposition method. This approach allows a robot trajectory planning without any collision.

The method divides the field in three possible cells: empty, full or mixing cells. The empty cells do not contain obstacles inside. The full ones are completely filled by obstacles. The mixing contains some part filled by obstacles and some empty part. Mixing cells are gotten dividing the main frame by backtracking in cells until it gets a minimum size of cell [1] or until it gets either an empty or full cell. If we choose a good minimum size, enough to avoid obstacle, we have a reasonable processing time.

Starting of the principle that in the end of the process, the cells that had been divided are empty or full, a graph is created connecting the empty neighboring cells. To a cell be neighbor of another one a common point is enough. Later, is executed a shortest path algorithm that uses Dynamic Programming Dijkstra [8]. In our graph, this algorithm gives the shortest path between two nodes (empty cells), giving an optimized planned trajectory for each robot, without collision.

[1]In this case, a minimum size mixing cell is gotten in a full cell.

IV. SIMULATION MODULE

To test the robots strategic decisions we develop a simulated environment, obeying the physical characteristics of the robots, see figure 2. The environment is being developed using an physics and simulation engine WEBOTS[9].

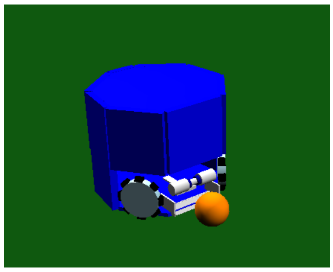

Fig. 2. The Webots Furgbol Robot.

V. THE COMMUNICATION SYSTEM

The CS broadcasts a set of packets containing the PWM levels, dribbler and kick informations and specific ID robot number. The robot owner of the packet must then extract the PWM levels from the protocol and validate it, sending this information to the Control.

The communication protocol consists a header containing the owner of the packet and the data about the PWM levels. The information about the owner of the packet is sent n times by the workstation, so if a robot does not receive this information $\lambda * n$ [2] times, it is discarded. This approach is an attempt to treat noises on the wireless link.

After validation, the Communication Module signals the Control System on the arrival of a new PWM levels. Each robot has its own Communication Module.

VI. THE EMBEDDED REACTIVE CONTROL

The Embedded Reactive Control System is responsible for the reactive behavior, receiving low level sensor signals and sending the control to the motors and actuators. This system is composed by the main processor, power stage, motors, gearbox reduction, low level sensors and kicker signals.

The control receives the PWM levels data coming from the Communication Stage, process them and activate the motors. These signals are calculated based on a pre-calculated table with the voltage curve of each motor attached to its gearbox reduction.

Each robot has a kicker and dribbler devices. We use a infrared sensor to detect the ball. This reactive stage activate this devices when some detection happens, enabling it only when the robot has the ball.

[2]Being $0 \leq \lambda \leq 1$.

VII. IMPLEMENTATION AND RESULTS

Building a robot team to play soccer is a big challenge in different fields. The range of technologies spans AI, robotic research and embedded system design. Therefore robots are ideal demonstrators for a number of research activities since they offer opportunities to evaluate various strategy-theories, software algorithms, hardware-architectures and design techniques.

The Furgbol system was developed in a computer with Intel Core i7 2.8GHZ processor and 6GB of RAM. The Furgbol software has been developed using GNU/Linux operational system and C++ programming language with the QtDesigner development tool.

A. The Communication Stage

The wireless communication is implemented with the LaipacTech's TRF-2.4G modules, at 2.4GHz to 2.5GHz frequency range, divided into 125 channels with center frequencies spaced 1Mhz. The workstation broadcasts the packets information about the PWM levels, with a bandwidth of 1200 bps. For instance, the CS sends two times the information about the owner of the packet. Each robot has also its own Communication Module, composed by the TRF-2.4G Transceiver. Currently the communication is one-way only. We use a 19200bps rate to transmit data.

B. The Embedded Reactive Control

The onboard processing is made by a 32 bits RISC microcontroller from the ARM7TDMI family, running at 16MHz. The ARM family of microcontrollers has a wide range of applications to assist on the programming process. In our project the C programming language was chosen, using the Keil Elektronik's uVision environment to generate the assembly code.

The board is divided in three distinct stages: Communication Stage (detailed earlier in this section), Power Stage and Control Stage. The power circuitry consists of L6235 drivers.

Power is supplied by two Li-Po batteries, each one able to deliver 7,4V/2800mAh. The Control Stage is responsible for the Actuation Step, which are implemented in the microcontroller program. Nowadays, we use four omni directional wheels in a 60 and -60 degrees disposition from robot's front line.

We have design and build chassis, wheels, dribbler and kick mechanic devices. The reduction gearboxes are able to rotate the wheels at 750 RPM with Brushless DC 12V motors, using omni-directional wheels. These wheels have a diameter of 50mm that makes possible to develop a maximum linear speed of 2m/s. In order to support the modifications over the old model, like new gearboxes, motors and wheels, a new chassis made out of aluminum was constructed. All stages and algorithms run online.

VIII. CONCLUSIONS

RoboCup contest is an important test-bed for several areas of the Robotic and Computer Science and Engineering. In this paper, we have described a low cost model underlying the FURGBOL Brazilian autonomous robot F-180 team, its implementation and our experiences with it. From a set of theories and algorithms, we have designed and implemented a real team of robots. We have proposed an architecture composed by three main modules proposed: *i*. a Deliberative Stage, *ii*. a Communication Stage and, *iii*. a Embedded Reactive Control. Our architecture was implemented using inexpensive and easily extendible hardware components and a standard software environment. FURGBOL has show to be a relatively successful approach; we are six times champion of Brazilian Robocup and champion of Latin American Robocup, and vice-champion of this same tournament twice.

IX. ACKNOWLEDGMENTS

We thank the Brazilian Council for Scientific and Technological Development (CNPq) and National Council for the Improvement of Higher Education (CAPES).

REFERENCES

[1] L. Parker, F. Schneider, and A. Schultz, *Multi-Robot Systems. From Swarms to Intelligent Automata*. Springer, 2005.

[2] S. Shimizu and H. Nagahashi, T. Fujiyoshi, "Robust and accurate detection of object orientation and id without color segmentation," in *RoboCup 2005*, 2005.

[3] A. Egorova, A. Gloye, A. Liers, R. Rojas, M. Schreiber, M. Simon, and F. Tenchio, O. Wiesel, "Fu-fighters 2003 (global vision)," in *RoboCup 2003*, 2003.

[4] R. D'Andrea, "Robocup2003 cornell robocup documentation, mechanical group final documentation," in *RoboCup 2003*. Springer, 2003.

[5] J. Loomis, J. Palmer, and P. Pandit, "Performance development of a real-time vision system," in *RoboCup 2003*. Springer, 2003.

[6] S. S. C. Botelho and R. Alami, "M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement," in *ICRA2000*, 2000.

[7] T. Laue, A. Burchardt, K. Cierpka, S. Fritsch, N. Gode, K. Huhn, N., T. Kirilov, B. Lassen, M. Miezal, E. Lyatif, M. Schwarting, A. Seekircher, and R. Stein, "B-smart team description for robocup 2007," in *RoboCup 2007*, 2007.

[8] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. McGraw Hill/MIT Press, 2001.

[9] Webots, "http://www.cyberbotics.com," commercial Mobile Robot Simulation Software. [Online]. Available: http://www.cyberbotics.com