UaiSoccer 2014 Team Description

Bruno de P. O. Paiva, Gabriel S. Ramos, Luiz F. S. Lemes,
Rodrigo A. Chaves, Rodrigo R. Caputo, and Wilson R. L. Júnior

Universidade Federal de São João del-Rei
Praça Frei Orlando, 170, Centro
São João del-Rei - MG - Brazil
uaisoccer@gmail.com

Abstract. This paper presents the UaiSoccer Small Size League team
with the intention to participate in RoboCup 2014. Describe the 3 main
modules: mechanics, electronics and software, showing our solutions to
the challenges that the category provides. We will present the new robot
developed in the last year, which allowed us a big jump in performance
and results for our young team.

1 Introduction

UaiSoccer is a Brazilian soccer robot team developed and builded by students
of the Federal University of São João del-Rei. It was founded in 2009 and the
following year attended the Latin American Robotics Competition for the Small
Size League, getting 4th place and becoming the best rookie in the category that
year. In 2013 the new robot was finalized and won 3rd place in CBR (Brasilian
Robotics Competition).

In nearly five years we have acquired a significant experience that has allowed
us to overcome many barriers and achieve some important goals. Today we have
a SSL robot that can be used not only for the competition category, but also for
research in related areas.

Initially, it was described the mechanical structure giving an overview of the
construction of items such as our wheel and kick. Then we use the electron-
ics embedded in our robot will be presented, microcontrollers, motors, drivers,
in addition to communication modules, among others. Finally we expose our
software system along with the algorithms and techniques used.

2 Mechanical Design

Each robot has 178 mm diameter and 148 mm high, within the rules of the
competition and are produced in large part by materials of high mechanical
strength and low density, such as aluminum alloy and alloy steels to guarantee
agility on the same drive without sacrificing the structural issues.

For the movement of robots are used four omnidirectional wheels, drive them
together generates the movement of the robot according to the desired.[2] This

drive system is very efficient because the robot does not need to be re-oriented to move, so widely used in the design of the teams participating in the Smallsize category.

The wheel is the most important component of the omnidirectional drive system, because their design should be done so that she can move in the direction perpendicular to the axis of the engine without her having to drag, which decreases friction. For this to occur, the wheel has pulleys all around, they are involved by o-rings which aims to facilitate the movement. The wheels are driven by EC 45 flat of 30 Watt brushless motors and the transmission is made by straight cylindrical gears machined in nylon 6.6 due its properties and system requirements.

Fig. 1. Exploded view

The angle of disposition of the wheels in an omnidirectional system is relevant in some parameters such as the effort required in each gear motor. Some settings are used for various project templates, our model was designed in partnership with SOARES[1], . They are configured to 33, 147, 225 and 315 degrees. Taken into account space restrictions due to the positioning of the kicking and dribbling systems.

The kicker consists of a set of large diameter-Solenoid interlock shaft driven by an electronic board, the movement of the axis is based on the laws of electromagnetism, which propels him forward with great energy. The dribbling system is responsible for maintaining the robot with the possession of the ball during the match, the silicone rubber coated shaft is driven by a brushless EC 20 flat of 5 Watt motor and his quick movement of rotation creates a backspin force, which directs the ball against the robot, and keeps them in constant contact. In our current project, the dribbling system involves approximately 16% of the ball.

3   Electronics

The Department of Electronics is responsible for the development of systems that control the activation and operation of the various necessary for the proper functioning of the robot electronics.
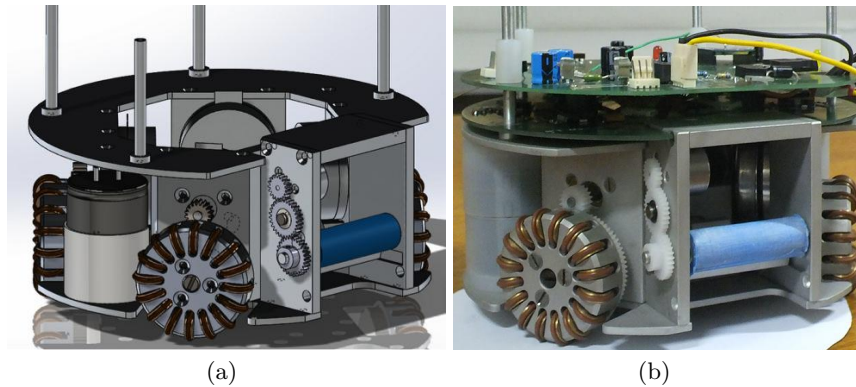
Fig. 2. (a) UaiSoccer's Robot's 2013 3D CAD (b) Real robot

The UaiSoccer Project 1.0 - Small Size is made up primarily in its electronic part of two printed circuit boards, which are called Mother_Board and BRD_Motores. These are necessary and sufficient for the correct operation of systems like motor control, wireless communication, triggering the kick, the ball detection, among other key to the robot that will be cited as more detail below.
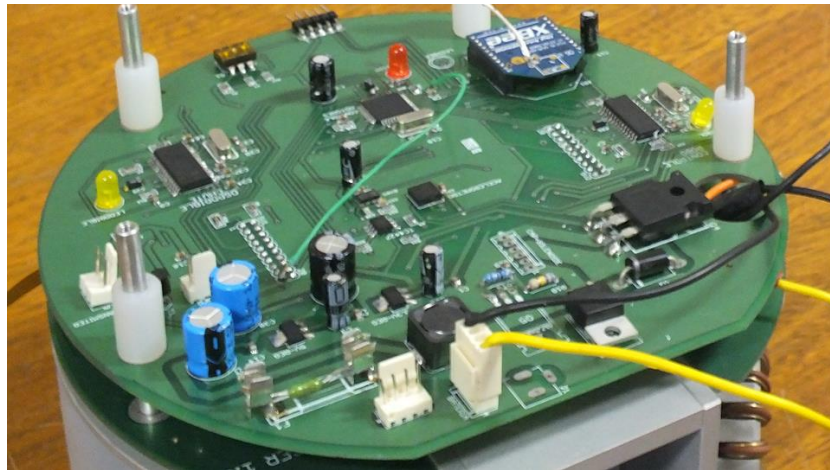

Fig. 3. Mother_Board and BRD_Motores in the robot

3.1   Microcontrollers

The UaiSoccer Project 1.0 - Small Size in your electronics, can be seen as highly complex, just when it was analyzed which type of microcontroller that would

use, from the consideration of issues such as available funds and also the need for component with high processing performance, reached the conclusion that the best option would microcontrollers dedicated to signal processing DSP's.

The DSP's acquired were as follows: a unit of DSPIC33EP64GP504 and two units of DSPIC33EP64MC202, where one has to his first gets its name from CONTROLE_MOTORES because its function is just driving the motors that are responsible for the motion of the robot.

The other two microcontrollers, one calling CONTROLE_ROBO and other DSPIC_CHUTE have different and specific functions. The DSP CONTROLE_ROBO has the function to receive information from sensors such as accelerometer and gyroscope that allow the control of the movement of the robot as a whole in a more robust and precise, while the other DSP calling DSPIC_CHUTE has the control function and monitoring of the following items dribbling, shooting, infrared sensor and battery level.

An interesting observation is that all connect to wireless communication module via UART communication and DSP's CONTROLE_ROBO CONTROLE_MOTORES and communications features including UART, SPI, I2C aimed at facilitating the maximum control actions to drive.

3.2 Motor and drive

The motors used on the wheels are the Maxon EC-45 Brushless Flat, that after searching the literature and with other teams category were seen by all members who participated in the project as the best option for the function because its switching and control for through the hall effect sensor and the ability to reach high speeds at low voltages were fundamental to the robot in a game that requires rapid acceleration and deceleration prerequisites.

However for the correct operation of these engines use a driver that is triggering these through a command from the DSP to indicate the desired speed is needed. An Overview of the drive system of the wheels is as follows: the wheels are driven by the engine, which, in turn, are driven by integrated circuits that are L6235PD our drivers, read the rotor position through the hall effect sensors and make logical drive motor.

Important information is that for the speed control of motors, one of the hall effect sensors for each motor is connected to the microcontroller (DSPIC33EP64GP504). Aware that this type of engine has eight poles, and that this results sixteen pulses per spin is possible to calculate the motor speed through time between a pulse and other hall effect sensor.

3.3 Kick System

When we thought about kicking system had a considerable problem that was how it would be possible to generate considerable strength to throw the ball, and the robot only power of approximately 12 V DC.

To solve the issue of tension after literature research has concluded that the use of DC Chopper called Boost converter would be able to work around this

limitation we had before. Through this converter was possible that a voltage of 12 V DC applied at its input to become 180V DC that are used in charging two capacitors. The parameters of the switching converter is 42 KHz with duty cycle of 70%.
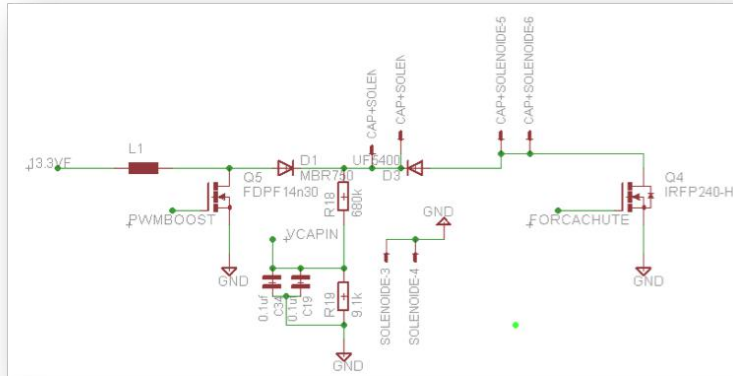

Fig. 4. Circuit of Boost Converter

With the two capacitors charged, it is possible realization of the kick. Which is driven and controlled through a MOSFET in saturation mode, which functions as a closed switch between the capacitor and the solenoid, and the longer closed, the larger key is the current flow on the solenoid and the greater the force being applied on the ball.

For security reasons the output voltage of the boost converter is monitored by DSPIC_CHUTE via a voltage divider that relates the maximum permitted output of the boost converter with the maximum voltage supported by DSPIC voltage.

3.4 Communication

For communication design makes use of XBEE 802.15.4 (Series 1) that is capable of receiving and transmitting data via a wireless communication module. Its interface with the microcontroller is via serial communication (RS-232).

The process of transmission of information is done by the base station that has the function to send a broadcast to all the information to the robots on the field who are able to identify which packet received this information belongs to you and discards the data that are not destined to it.

3.5 Driver Motor Dribble

The engine is dribbling the EC-20 Brushless Sensorless flat, ie, it does not have hall effect sensor. Therefore it was necessary for the chosen drive the driver had the same ability to control motor speed without speed measurement sensor. The TDA5145T integrated circuit is able to perform this task, due to the artifice of "the WEF engine" to identify the position of the engine and make the switch logic.

4 Software

4.1 Software Project

The software area has the purpose of analyze the game and create strategies for the coordinate of the robots. The system was developed in C++ using the QT Creator Framework, wich is executed in an Ubuntu Linux OS platform.

The process that covers the software can be defined as seen in Figure 5 below. Some parameters are necessary for decision making in the system, and those parameters can be obtained from available tools of the RoboCup: the Referee and the SSLVision.
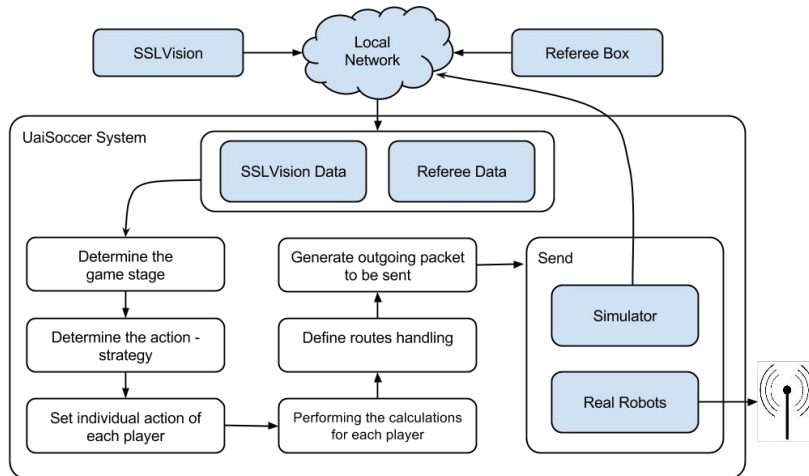

Fig. 5. UaiSoccer System Diagram

4.2 Referee and SSLVision

The referee is a software that sends data to the robots about the game state. These information are used with the purpose of making the robots follow the

rules of the game and respond to signals like stop, fault, among others. The SSLVision is a system of capture and processing images from the field. This software, even as referee, sends information through UDP packages in a network that connect all the computers associated with the game. These packages hold information as field width, field length and more advanced informations, like positioning and directioning of each robot in field.

4.3 Data receive

To maintain the data reception always updated, was necessary a module of reception. This module is composed by two subsystems responsables for receive data from the referee and the SSLVision, as seen in Fig.5. Noise filters, as Kalman [4], are aplyied for these modules, in order to make them less suscetible to server transmission and cameras catchment errors.

4.4 Finite State Machine

For the software make decisions in game, like start attack mode or defense mode, we choose to use a finite state machine (FSM). A FSM is a mathematical model used to represent behavior changes in the software, depending on the parameters. In our software, we use the information provided by referee and the SSLVision about the state of the game, to decide the actions that each robot will take.

4.5 Tasks designation

After the state of the game be decided, is necessary to assign tasks for each one of the robots. For each state of the FSM, we have a priority queue that define what position that each robot will assume, as we can see in Fig.5. Thus, we choose the best robot for execute the functionalitys that we find in the first position of the queue. Then, we repeat this process until all the robots have their functions in game.

4.6 Players activitys designation

After the designation of tasks, the computation of the new positions for the robots is realized. This computation take into account the construction of the players. Every action to be executed is connected with a specific function writed in the code, that determines the position, orientation and actions that each robot must have.

4.7 Navegation

With te players ativitys designated, we apply an algorithm of navegation to define the path for a new position. This algorithm is based in shortest path problem in graphs. Thus, each vertice of the graph will be the robots, and the

edges are represnted by the connection of two robots that do not have any barrier between them.

From this model, we apply the Dijkstra algorithm[5] to define the best and shortest path to the robot.

4.8   Shipping information

With the path setted, we define the velocitys in each one of the wheels to execute the path, then we create a package with all the information necessary for the robot. We send this package to the responsable module, that can send the package to the robots or to the GrSim[6] simulator.

If we are shipping information to the real robots, is utilized the protocol of communication of the XBee transmitter. Otherwise, for send the information to the simulator, we transfer the packages using the UDP protocol in a local network. Thus each iteration of the system will generate new data that will be sended to the right place.

References

1. SOARES, S. J. P. Controle de um Robô Móvel utilizando linearização por realimentação. 2011. 68 f. Tese (Mestrado em Engenharia elétrica) – Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de São João Del Rei, São João Del Rei. 2011
2. INIESTA, E. D. (2006): Diseño e Implementación de Los Robots F180 Del ITAM. Instituto técnologico autônomo do México, 2006.
3. Fialho, A. S., Búrigo, A. C., Jurkovski, B., de Azevedo, C. K., Adolfo, C., Dalbem, C. M., de Melo Leonardi, E., da Fontoura Beltrão, F., Chies, F. A., Silva, G. H., Schmidt, I. A., Lopes, I. F., de Mello Bones da Rocha, J. P., da Silveira, K. S., Barreto, L. C. M., Zawacki, L. F., Cavinato, M. V., e Barone, D. A. C. RoboPET team description paper. Proceedings of 14th International RoboCup Symposium, Singapore, 2010.
4. G. Welch and G. Bishop. An introduction to the kalman filter. Department of Computer Science, University of North Carolina, Tech. Rep. TR 95-041, 2001.
5. Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford (2001). "Section 24.3: Dijkstra's algorithm". Introduction to Algorithms (Second ed.). MIT Press and McGraw–Hill. pp. 595–601.
6. GrSim – RoboCup Small Size Robot Soccer Simulator; V. Monajjemi, A. Koochakzadeh, S. S. Ghidary - 2012;