RoboIME: on the road to RoboCup 2016

Jan L. L. Segre, Lucas O. de Lima, Naum A. F. Barreira, Victor Bramigk, and Paulo F. F. Rosa

Instituto Militar de Engenharia (IME)
Praca General Tiburcio, 80 - Praia Vermelha, Urca
Rio de Janeiro - RJ - Brazil
roboime@googlegroups.com

Abstract. This paper describes the electronic, mechanical and software designs developed by RoboIME Team in order to join RoboCup 2016. All designs are in agreement with the rules of Small Size League 2016. This is the third RoboIME participation in a world level RoboCup event, although the team has already been challenged three times in competitions in Brazil and Latin America.

1 Introduction

RoboIME is a small-size league soccer robot team from IME, Rio de Janeiro, Brazil. This is only the 5th time the team is taking part in competitions. The biggest result achieved was in 2012 when the team achieved second place in Latin American Robotics Competition.

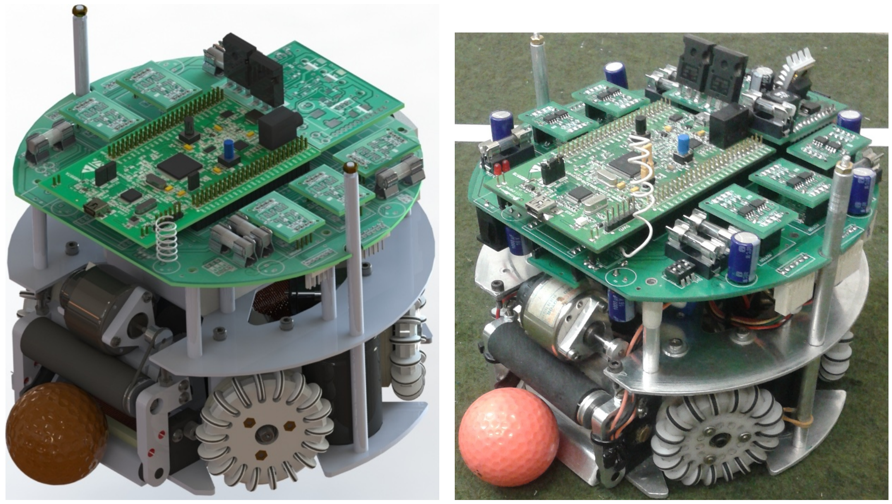
All students that work in this project are members of the Laboratório de Robótica e Inteligência Computacional at IME. Previous studies [1][5] provided the basis for the current structure of software and hardware teams. This paper describes the computer, electronic and mechanical designs.

This work is organized as follows. The mechanical design of RoboIME robots is presented in section 2. Then the firmware and electrical project is presented in section 3. The software system is presented in section 4. Finally, discussion and future work are described in section 5.

2 Mechanical Design

This robot was designed and built using CAD (Computer Aided Design) and CAM (Computer Aided Manufacturing) software. Moreover, extensive testing was done to validate the current project.

Most of our robot parts were CNC machined and made out of 7075 aluminium and high density polyoxymethylene (POM). The POM have some excellent properties such as high rigidity, good impact resistance, a non-stick characteristic and beng a highly machinable material. In this way, some parts of the robots, like the plunger stopping body, are more suitable to be made out of POM than aluminium. For example, the dribbler arm is a pivot-rotating mechanism, and using POM eliminates the need for a bearing within the assembly.



(a) (b)

Fig. 1. 3D model 1a and real robot 1b views.

2.1 Dimensional Constraints

In compliance with the SSL rules, the height of the robot is 149 mm and the maximum projection of the robot on the ground is 180 mm.

Using CAD software we were able to measure the percentage of the ball area that was covered by the robot. The maximum percentage of ball coverage found was 19,8%, in accordance to the 20/80 rule of the league. The height of the dribbler cylinder is also adjustable, so we are able to find the optimum point for the best ball control.

2.2 Transmission System

A system of internal gears was made to transfer the power of the motors to the wheels. This system has several advantages compared to traditional gear meshing, such as avoiding debris entering the motors, creating a cavity to apply grease for lubrication of gears and an overall smaller size.

However there are some difficulties in the manufacturing of this part, mainly due to the small size of the teeth needed to mesh with standard motor gear (the motor being used is the Hsiang Neng DC brushed motor type HN-GH35GMB). At this motor the distance between two consecutive teeth is less than 1 mm, thus it was not feasible to machine the internal gear. So it was decided for 3D printing in ABS plastic as the manufacturing process.

The traditional fused filament deposition method for 3D printing, in geometries smaller than the filament itself, create cavernous structures that weaken the piece. Applying the stereolithography 3D printing process (when a laser beam cure a liquid resin layer by layer), we manage to achieve a higher resolution.

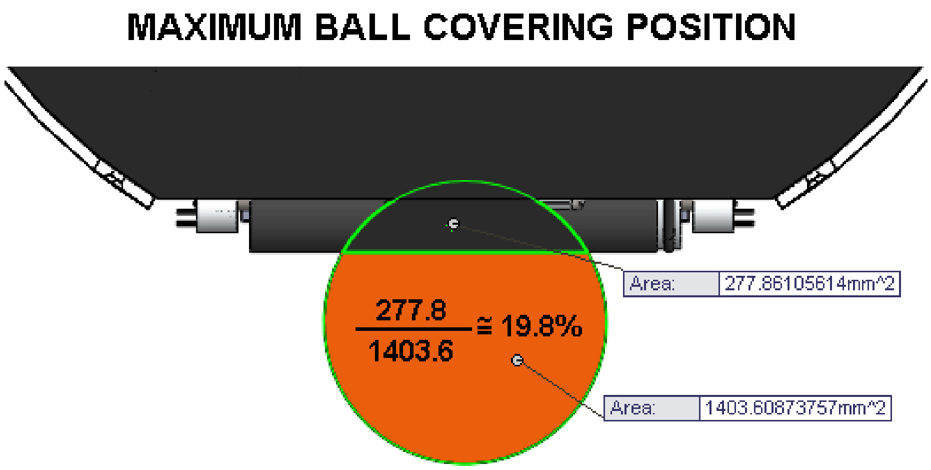


Fig. 2. Maximun covered area of the ball.

This way we can precisely print the teeth profile, avoiding failures due to empty spots and achieving a more solid and precise component.

2.3 Chip Kick

The chip kick is based on a flat solenoid, which is mounted in a slot at the chassis (close to the ground). When activated the core of mild steel is accelerated against the rear of the chip, which revolves around its axis and makes the ball rise. Due to the limited space, complex construction and details, we also have chosen the 3D printing as the manufacturing process.

The flat solenoid is assembled in a way that works as a guide rail for the kick plunger as well. We are using rubber bands to pull back the chipper and kicker plungers, keeping the mechanism simple. The final dribbling/kicking mechanism is a very neat assembly and can be easily adapted for any other chassis.

3 Embedded System

RoboIME electronics consist of nine boards: (a) the Main Board, responsible for communication between the other boards; (b) the Stamp Board,responsible for the embedded computations; (c) the Kicker Board, responsible for maintaining high voltage and activate the kickers; (d) five Motor Controller Boards which are responsible for the robot's motion control and the dribbling device; (e) the Transceiver Board, which is responsible for the link between the robot and the main computer. All these boards are describes below in this section.

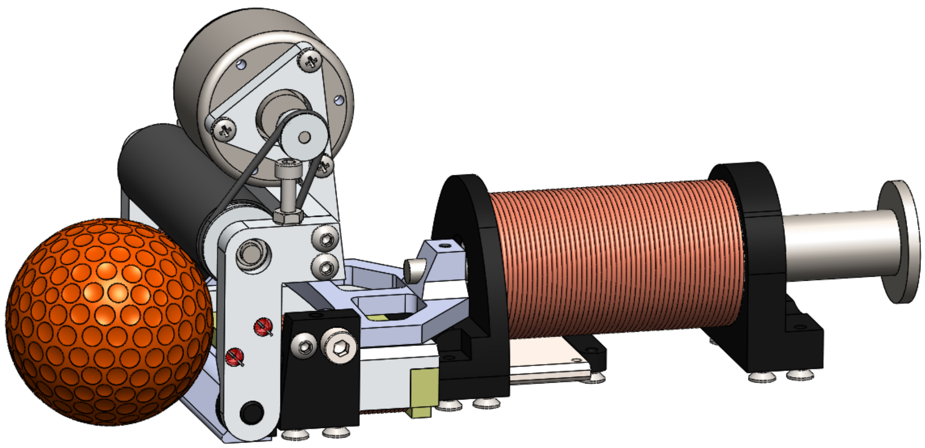


Fig. 3. Dribbler, chipper and kicker assembly.

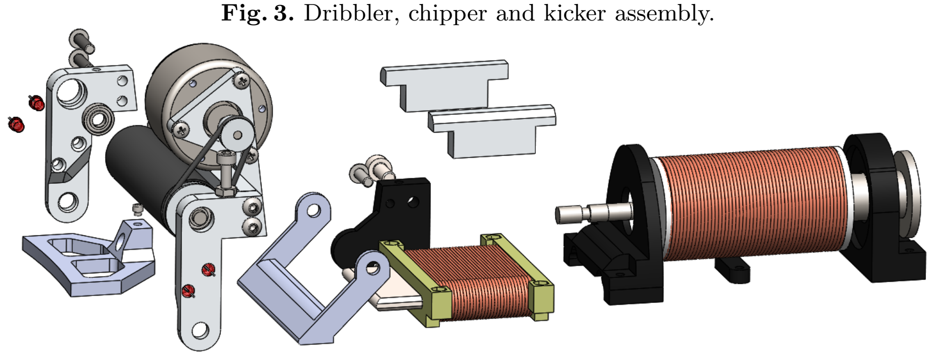


Fig. 4. Exploded view.

3.1 Main Board

The Main Board features a socket to plug the boards in: the kicker's sensor, a optical sensor used to detect if the robot has the ball possession; dribbler motor, which makes possible to the robot to spin and move backward without losing the ball; four quadrature encoders and the power supply with safety devices.

3.2 Stamp Board

This board is responsible for performing all the logical function. Serving as a brain for the electronic system there is an embedded STM32F407VG microcontroller, with an ARM Cortex M4 as main CPU, 1 MB Flash, 192 KB RAM memory working at 168 MHz, that was programmed in C using CoIDE and Eclipse IDEs. The main function of the embedded system is to receive data from the AI and convert into movement. For do so, there is a Proportional Derivative Integrative Control sampling the real wheel's velocity, comparing with the desired and outputting the appropriate voltage to the motor. That control has fundamental importance in looking for the correct velocity of the wheel. There is also a current control that avoid the Motor Controller Board to burn out.

3.3 Kicker Board

This board is responsible for produce the high voltage used to activate the two coils, controlling the kick strength and discharge almost instantly all the power stored on the coils. There are two kinds of kick, the forward kick and the high kick. There are two steps in this board: charge and discharge. The first one has the unique function of keep a constant output of 180V DC from an input of 7/8V DC. A DC-DC step-up power supply controlled by the MC34063 IC and two electrolytic capacitors of 2200F, 200V are used for this task. The second one is to drive the kickers. In this part are used one TC4427 MOSFET Driver IC and two IRFP4868PBF Power MOSFETs that are responsible for close the high voltage circuit of the first step with the ground through the coil, converting electrical into mechanical energy. A precise control of the actuation time ensures that the kick will occur with the right velocity.

3.4 Motor Controller Board

The idea of the RoboIME electronic is to modularize the electronic project. For this, there is one controller module board for each wheel motor. If one of them burns out, it is possible to exchange it quickly. Each board has two TC4427 (MOSFET driver) and two IRF7319 (complementary half H bridge). These ICs create an H-bridge, allowing the velocity control in both directions through a Pulse-Width Modulation, converting a digital signal input into an analog output.

3.5 Transceiver Board

The Transceiver Board is responsible for the link between the microcontroller and the main computer through a nRF24L01 wireless chip. The first is done using a own protocol, operating in the 2.4GHz band, simplex, fully compliant with FCC and ETSI regulations. The second is accomplished through the Serial Peripheral Interface Bus (SPI), a standard.

4 Software Solutions

The software systems consist of three projects: pyroboime (AI), ssl-webclient (graphical client), grSim (simulator).

4.1 Artificial Intelligence

The AI, pyroboime, is based on the STP (Skill-Tactic-Play) architecture and implemented in python. It has the following components: interface with the ssl-vision, ssl-refbox and grSim, it also has a built-in communication module for the radio transmitter system.

The STP is a three tier architecture where the lowest level, skills, enables the low level manipulations on a single robot. The middle layer, tactics, makes use of the skill layer to execute higher level behaviour, possibly enabling coopration, but still acting on a single robot. The upper layer, plays, coordinates the tactics associated to each robot in order to maximize performance, each play is implemented to behave according to specific states: stop, halt, indirect kicks and normal play. A higher level layer implmented as a play switches between other plays based on the current referee state. This architecture is better depicted in figure 5.

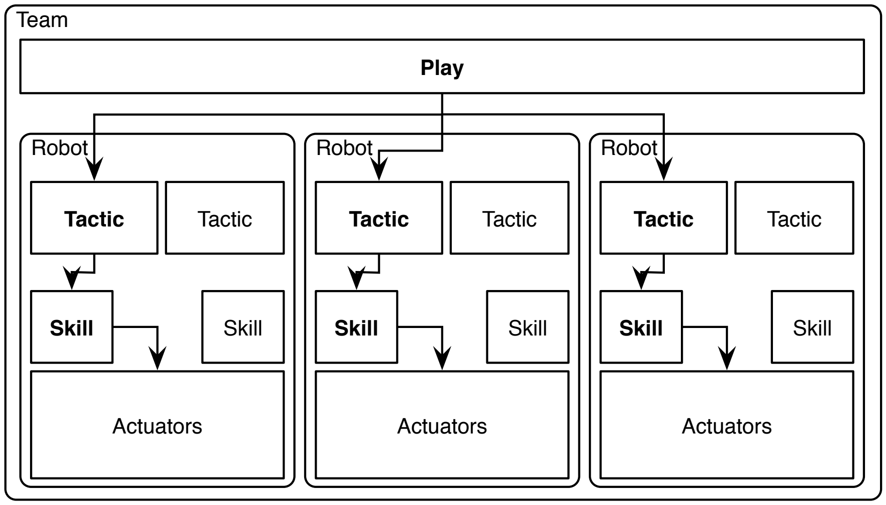


Fig. 5. STP Diagram

There is also a skill implemented to redirect inputs from a joystick such that it is possible to test the robot with little effort.

The interface is structured as a filter stack which connects the AI to a flexible collection of updaters (which receive state information) and commanders (which deliver commands to the robots, in both the simulated and real environments). It abstracts the external environment where the game is played from the AI. Among the filters in said stack is a Kalman filter that reduces the noise comming from the updater data.

The built-in radio trasmitter system interfaces with libusb to control the transmitter hardware, which is connected via USB.

4.2 Support systems

The graphical client, ssl-webclient, is a Web interface implemented in nodejs using WebSockets, HTML5, SVG and ZeroMQ. It has the following functionality: displaying and altering the AI state, broadcasting games through the internet and playing log files.

Lastly the simulator, grSim, originally developed by Parsian Robotic, was customized to fit our needs. It provides the following functionality: simulating the game environment and exposing an ssl-vision compliant protocol.

Fig. 6. Snapshot of the simulator

4.3 Source

All of the projects above have been open sourced with GPL-like license, with the main difference being that a derived work used on a competition must have its source released by the next edition of that competition. The sources of those projects are available on the team's github page: http://github.com/roboime/.

5 Discussion and Future Works

For the this competition, following goals are being sought: validate the reestructure of the IA system; stabilize the new transmission system and the new parts in the mechanical design.

Acknowledgements

This research was partially supported by Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro - FAPERJ(grant E-26/111.362/2012); Fundação Ricardo Franco (FRF) and Fábrica de Material de Comunicação e Eletrônica (FMCE/IMBEL). The team also acknowledges the assistance of Mr. Carlos Beckhauser from FMCE. Special thanks to Diego Felix de Almeida, Luis Renault L. Rodrigues, Stefano H. Rodrigues, Thiago A. Navarro do Amaral and Victor L. Horta Ferreira, former team members that made this project possible.

References

1. Alexandre Tadeu Rossini da Silva: Comportamento social cooperativo na realização de tarefas em ambientes dinâmicos e competitivos. Instituto Militar de Engenharia, Rio de Janeiro (2006)
2. Madeira, B. E., de Almeida, D. F., de C. Maia Jr., E., Rodrigues, L. R. L., Rosa, P. F. F., Rodrigues, S. H., and do Amaral, T. A. N.: RoboIME: Team Description Paper for RoboCup 2012.
3. B. Browning and J. Bruce and M. Bowling and M. Veloso: STP: Skills, tactics and plays for multi-robot control in adversarial environments Carnegie Mellon University, Pittsburgh, PA (2004)
4. Khatib, O.: Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. In International Journal of Robotics Research, vol. 5, no. 1, p. 90-98 (1986)
5. Marco Antonio Firmino de Sousa: Uma Plataforma para Cooperação Autónoma de Múltiplos Robôs Instituto Militar de Engenharia, Rio de Janeiro (2008)
6. Russell, Stuart J.; Norvig, Peter: Artificial Intelligence: A Modern Approach (2nd ed.) Upper Saddle River, New Jersey: Prentice Hall, pp. 163-171 (2003)
7. Stefan Zickler: Physics-Based Robot Motion Planning in Dynamic Multi-Body Environments Carnegie Mellon University, Pittsburgh, PA (2010)
8. Welch, G. and Bishop, G.: An introduction to the Kalman filter. Technical Report TR 95-041, Department of Computer Science, University of North Carolina (2001)