RoboBulls 2016: RoboCup Small Size League

M. Shamsi[1], J. Waugh[1], F. Williams[2], A. Ross[2], and M. Llofriu[1,3] A. Weitzenfeld[1]

[1] Dept. of Computer Science and Engineering
[2] Dept. of Electrical Engineering, University of South Florida, Tampa, FL, USA
{muhaimen, waugh2, fwilliams3, anthonyross, mllofriualon, aweitzenfeld}@mail.usf.edu
[3] Facultad de Ingeniería, Universidad de la República, Uruguay
mllofriu@fing.edu.uy

Abstract. In this paper we present the design and implementation of our Small Sized League (SSL) RoboCup Team RoboBulls. Since this is our first participation in the RoboCup SSL, we attempt to explain every aspect of our robot hardware in as much detail as possible. Our 2015 TDP described our software architecture in full, so we summarize the changes made since then.

1 Introduction

Though we qualified for RoboCup 2015, a lack of man power and funding made it prohibitively difficult for us to participate in China. Since then our focus has been to improve the effectiveness and reliability of our robot hardware. In particular: we upgraded our motor controllers, improved our kicker design and power system, and developed a dribbler. These changes are described in detail in section 3.

Our focus on building the team was achieved at the cost of significant improvements to our software. However, we were able

2 Software

This section describes the changes made since our last TDP.

2.1 New Strategies

We refer to Strategies as high-level pieces of code that coordinate assigning Behaviors. We have implemented a new strategy for normal gameplay that can work with a variable number of robots. This is useful because the number of working robots often varies due to hardware malfunctions and development.

The strategy works as follows:

1. If the enemy team has the ball, all robots but one attempt to mark opponent robots on the friendly side of the field. The one free robot sits at mid-field, waiting for an opportunity to attack.

2. If no one has the ball, two robots keep marking the opponent robots in the friendly side of the field, and all others attempt to either take receiving positions. The robot closest to the ball attempts to intercept it.
3. If we have the ball, two robots keep marking the opponent robots in the friendly side of the field, and all others attempt to either take receiving positions. The robot with the ball prioritizes scoring, dribbling, and passing.

2.2 Improved Kick Skill

Our most used skill is called kickToPoint, which as the name suggests, is meant to automate the process of driving to the ball, aiming to a given target, and kicking the ball. One of the main problems we had was not being able to detect when the robot was close enough to the ball to actuate the kicker since we do not have a laser sensor to detect the ball like other teams.

Our previous approach was to position the robot behind the ball, and then move forward while checking the robot-to-ball distance, $\mathrm{d_{rb}}$. When the $\mathrm{d_{rb}}$ value dropped below an empirically determined threshold, the kicker was actuated.

This approach did not work consistently since $\mathrm{d_{rb}}$ can be too high to trigger the threshold if the robot approaches the ball at an offset, preventing the robot from kicking when it actually should. On the other hand, if the threshold is lowered to account for the offset, the robot kicks too early when approaching directly. This is illustrated in fig. 1 and fig. 2

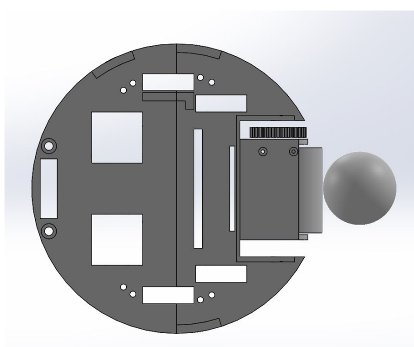

Fig. 1: Direct contact with ball

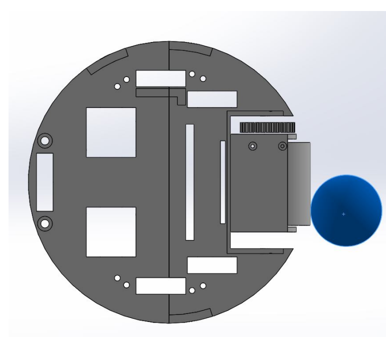

Fig. 2: Contact at an offset

Our new approach does not check $\mathrm{d_{rb}}$. Instead:

- We move to a position, p, from where we can kick the ball to the target.
- We move towards the ball, checking the robot's distance from P, $\mathrm{d_{rp}}$.
- If $\mathrm{d_{rp}}$ is above a certain empirically determined threshold, we actuate the kicker.

This approach works much better because the robot is guaranteed to kick after traveling a certain distance towards the ball, and thus is offset tolerant. Due to the frequent use of the kickToPoint skill, this improvement has been very effective in improving the flow of our games.

2.3 Improved GUI (Graphical User Interface)

Fig. 3 shows our GUI developed using the Qt framework. Latest additions include the display of the current behavior assigned to any given robot, as well as the ability to manually control multiple robots from a single client using controllers. This is useful for debugging strategies on the field, for play testing against the software, and for demonstration purposes. Future plans include the ability to mark regions on the field and behavior assignment.

Fig. 3: GUI showing various information from our GameModel

3 Robots

We plan to attend RoboCup 2016 with at least five robots with kickers and dribblers. We do not have the space to implement chip-kickers because the motor gearboxes are too large. Four of these robots have been built and tested so far, and we are currently building the fifth robot. This section describes the current robot design and the pending changes planned for the competition. Fig. 4 shows four second generation RoboBulls SSL robots. They are currently equipped with a kickers and dribblers, but a chip-kicker implementation was beyond on the scope of this years improvements because of limited space on the chassis.

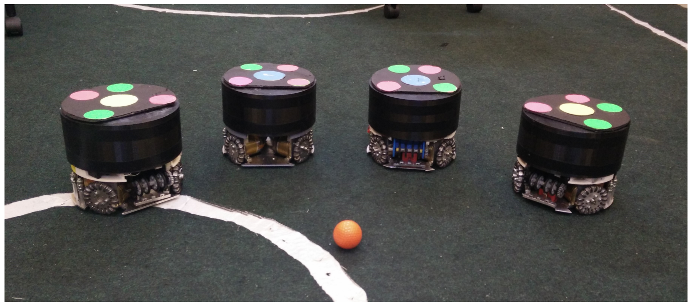

Fig. 4: First generation RoboBulls SSL robots

3.1 Components

The current robots are identical in their design. They have been constructed with custom aluminum plates and brackets made to firmly attach the motors. The electrical components are organized in a housing compartment designed in SOLIDWORKS and printed by MakerBot 3D printers at USF facilities. This ensures our electrical components are positioned safely and accessibly inside the robot.

Table 1: List of electrical components per robot

Component	**Number**	**Function**
Maxon EC45 Brushless DC Motors with Spur Gearhead	4	Spin omni-wheels to propel the robot in any direction
Arduino Mega 2650	1	Receive commands over Xbee radio and generate various actuator signals
4S Lipo 20C 3000mAh	1	Supply various components with power
Voltage Alarm	1	Sound alarm if battery voltage is too low
DC-DC Step-Up Converter	1	Provide 250VDC source to charge Capacitor
Solenoid And Plunger	1	Kick the ball along the ground
Dribbler Frame, Drum, and 12V DC Motor	1	Hold the ball close to the robot while moving
250 VDC 2200uF Capacitor	1	Provide high-power discharge to solenoid
Voltage Regulator	2	Provide appropriate voltage for Arduino and dribbler
PCB	1	Routes power to the dribbler, solenoid, and capacitor - controlled by the Arduino Mega

3.2 Drive System

We removed the optical encoders which were attached to the back of our motors since they were made redundant by new controllers from Maxon Motors ESCON 36/3 servo controllers (part number 336287). These are able to accept Hall Sensor feedback from the motor in order to determine the speed, and come with built-in closed loop speed control ability. They are also capable of auto-tuning through a free program provided by Maxon Motors called ESCON Studio. Overall this improved the motion of our robots because the controllers provide much higher starting torque compared to the optical-encoder method. This is because the Hall-effect sensors allow the controllers to read the position of the rotor so that the angle between the rotor flux and the stator flux can be kept as close to 90 as possible [1]. The controllers have built-in over-current and over-voltage protection to prevent damage to the motors during stalls. The placement of the motors is shown in fig. 5.

3.3 Kicker

The kicker consists of a hand-wound solenoid mounted onto the base of the robot. It is powered by a 250V 2200uF capacitor, which is charged by a step-up converter at 200V. Two 5V relays controlled by the Arduino Mega act as switches to control the charging and discharging of the apacitor. A kick is actuated by a 15ms pulse of current at 200V from the capacitor to the solenoid. This allows for a maximum kick range of approximately 15 meters. The software limits the kicker to 1 kick per 6 seconds to prevent the solenoid from overheating, and a rubber-band hooked to the back of the arm retracts it after each kick.

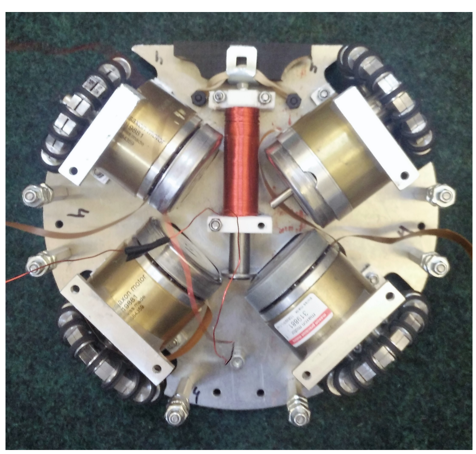

Fig. 5: Motors and kicker attached to chassis

3.4 Dribbler

The dribbler consists of a 3D printed frame which houses a 12V DC motor (1030rpm free-run, 3.2 kg-cm torque) and roller that makes contact with the ball. It is attached to the body of the robot by a free moving hinge so that the frame can rotate backwards by a maximum of 4 degrees. The back of the frame is padded with compressible material to absorb impacts from the ball.

The roller is made from Lego parts since the Lego wheels provide a smooth, rubber surface for contact with the ball. We use 4 Lego wheels with gaps between them for the ball to move into, which prevents lateral motion. The rotation is transferred between the motor and the roller by 2 spur gears; one is a small Lego gear and the other is a 3D-printed (ABS material) gear sized appropriately to fit the frame. It is shown in fig. 6.

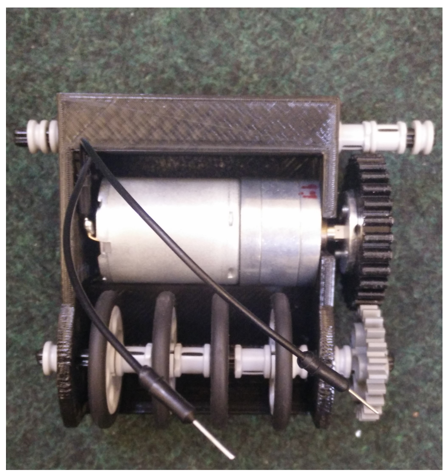

Fig. 6: Dribbler frame with motor and Lego roller

3.5 Power

Each robot is powered by a pack of 4S 20C LiPo (Lithium Polymer) batteries of 3000mAh capacity. The 15V pack powers the Arduino, ESCON Servo controllers, the dribbler, and the step-up converter in parallel. One parallel connection is regulated down to 8V for the Arduino Mega 2650 and another is regulated down to 12V for the dribbler motor.

3.6 Planned Improvements

Development of smaller gearboxes Our current motors come with gearboxes that occupy a majority of the space on the chassis. This prevents us from housing a chip kicker since there is barely enough space to fit a normal kicker. We plan to replace the gearboxes with a transmission mechanism built into the omni-wheels. We decided to forego this for the 2016 competitions due to a lack of time and manufacturing ability.

Increased kick power While the current kicker design is able to propel the ball down the length of the field, the velocity of the ball is too slow to prevent the ball from being intercepted by opponent robots over distances greater than 1.5m. We plan to improve this by experimenting with the duration of the impulse given to the solenoid, using higher voltages, and adding more capacitors for more current. In order to implement any of these changes, we first have to resolve issues related to the power relays burning after a certain period of use.

Chip-kicker We are in the process of designing a chip kicker that will fit on the chassis once the large gearboxes have been removed. This will prove essential for clearing the ball and making passes when blocked by opponents.

4 Involvement

The USF RoboBulls participate annually at the USF College of Engineering Expo. In 2015 we hosted several hundred K-12 students at our lab where they could build Lego NXT robots to play soccer and run it under our SSL framework. This very successful and there were a surprising number of designs that work very well. The children were able to get hands-on experience in building robots and received encouragement and help from our team members.

For the 2016 Expo we plan on having the students remotely control our SSL robots for 2v2 games of robot soccer. This will help us test the stability of our robots for the 2016 RoboCup tournaments.

5 Conclusion

We presented detailed hardware overview of the RoboBulls SSL team. Using our system we able to achieve fast holonomic omni-directional motion, ball handling, and kicking. The control software has been developed and tested in full games in addition to using the grSim simulator [2]. All layers involved in the processing of one vision frame; namely network communications, world modeling, high-level strategies, low-level behaviors and skills, communication, kicking and motor control have been implemented, tested, and shown to work.

Short-term future work consists of improving the current robots for the 2016 competition, such as improving the kicking power. Long-term future work will consist of redesigning the whole fleet with gearboxes built into the omni-wheels and developing a chip-kicker.

More information can be found at www.usfrobobulls.org

Our Qualification Video can be found at:
https://www.youtube.com/watch?v=F4DgH5gKi4s

6 Acknowledgments

This work is supported in part by the USF Student Government, the Bio-Robotics Lab at USF and the NSF grant #1117303 entitled "Investigations of the Role of Dorsal versus Ventral Place and Grid Cells during Multi-Scale Spatial Navigation in Rats and Robots". We would like deeply thank Juan Calderon and the STOx's small size league team of Santo Tomas University for their help and advice.

References

1. Gamazo-Real, Jos Carlos, Ernesto Vzquez-Snchez, and Jaime Gmez-Gil. "Position and speed control of brushless DC motors using sensorless techniques and application trends." Sensors 10.7 (2010): 6901-6947.
2. Monajjemi, Valiallah, Ali Koochakzadeh, and Saeed Shiry Ghidary. "grsimrobocup small size robot soccer simulator." RoboCup 2011: Robot Soccer World Cup XV. Springer Berlin Heidelberg, 2012. 450-460.