CRoNE 2019

Conference Proceedings

Mechanical, hardware and firmware considerations for a Robocup SSL robot

Pablo Reyes Robles[1*]**, Maximiliano Aubel**[1*]**, Nicolás Hernández Peña**[1*]**, Jorge Alvarez**[1]

[1]Innovación y Robótica Estudiantil, Universidad Técnica Federico Santa María

*For correspondence:
pablo.reyesr@alumnos.usm.cl
(P.R.R.); maximiliano.aubel@
sansano.usm.cl (M.A.); nicolas.
hernandez@alumnos.usm.cl
(N.H.P)

Abstract   Sysmic, formerly AIS, is a team of students that develops robots capable of playing football autonomously to participate in the Small Size League category of the Robocup Soccer league. One of the main purposes of the competition is to collaborate and share new features with other teams, implying an accelerated rate of innovation and complexity of the robotics systems. Sysmic participated in the 2018 edition of the Robocup in Montréal, Canada, identifying what are the main issues to fix or change to reach the current standard of development of major teams. This article describes new features that the team is currently working on, showing significant improvements in hardware and mechanical fields. The main conclusion is that the current path of development is resulting in substantial enhancements to our former robotics systems and, future development plans are presented.

Introduction

Sysmic Robotics, previously Artificial Intelligence Soccer (AIS), is a team of Federico Santa María Technical University students who design a system of multi-agent robots capable of performing in football matches autonomously and intelligently. The team was formed in 2011 to generate mobile platforms and experience the drive of brushless direct current (BLDC) motors. From this process, the team decides to participate in the Robocup competition in the Small Size League (SSL) category, whose system corresponds to the confrontation between teams of 6 robots in highly dynamic environments, with a hybrid centralized/distributed system consisting of a set of cameras on the playing field whose image is processed by software, which allows to detect the robots of both teams and the ball used. Depending on the state of the game, said software determines the set of actions to be carried out by each of the agents and is sent through a wireless communication system of each team's preference, all of which are following the rules of the competition.

Currently, the team has participations in the Latin American and Robotics Competition (LARC) in 2013 and in the 2014 and 2018 editions of the Robocup. In the latter, the team obtained a sixth place in the table in Division B of the category. From these instances, the team is developing the fourth generation of robots, whose main technical features are presented in this article.

First, the redesign of the mechanical structure of the robot is presented, highlighting the efficient redistribution of the internal components with respect to the previous generation resulting in a decrease in the height and center of gravity, which translates into greater stability. Subsequently, the redesign of the hardware system is presented, showing schematically the changes to a new architecture that have the centralization of processes in a single high-performance MCU, the increase in transmission rate through wireless communication after the renewal of the transceiver

Table 1. Summary of main changes from former robot generation.

Robot generation	2018	2019
Microcontroller	PIC32MX440F256H x5	STM32F767BI6
Drive motors	Maxon EC-45 30W	Maxon EC-45 50W
Encoder	Hall sensors only	Maxon MILE 2048 CPT
Dribbler motor	Maxon EC16 30W	Maxon EC22 25W with Hall sensors
Communication module	APC220 433[MHz]	NRF24L01+ 2.4[GHz]
Dimension	18 cm diam x 15 cm	18 cm diam x 11 cm
Gear reduction	1:2	1:3
Wheel diameter	55 mm	50 mm
Small Wheel per Drive train	15	18

module and the addition of high-resolution encoders for speed sensing of their motors. Finally, current software development is presented as future work, due to its early state.

Mechanical and electrical system

General Structure

The robot is shaped like a cylinder and according to the rules of the competition, it can have a diameter of at most 18 [cm] and a height of 15 [cm]. Our previous design has the limit height of 15 [cm] with a lot of space available inside due to the inefficient distribution of its components. For this new version, the height was reduced to 11 [cm], considerably lowering the center of mass of the robot. Also, our previous design considered an 18 [cm] PVC tube as the case that, due to its rigidity, transfers much of the impacts it experiences to the internal parts. Currently, the case is made of hard cardboard because it only has to protect the upper section of the robot that receives very few impacts, but it is evaluated to be made of carbon fiber. To optimize the space, the supports that join the main PCB to the robot as part of the case are joined, giving easy access to the parts most prone to damage and the battery which speeds up maintenance work. These changes respond in large part to the efficient use of internal spaces, one of the main problems that we could identify in the Robocup, Montreal thanks to feedback with other teams[1].

Dribbler

The *Dribbler* is the system that allows the robots take control of the ball, without covering more than 20% of the ball. It consists of a roller located at the front of the robots, which spins in such way that when the ball enters in contact with it, it will begin rotating towards the robot. Thanks to the Dribbler, a robot should be able to receive a high speed (up to 6.5 [m/s], which is the competition limit) ball and control it. This means, to be able to move the robots in the field without losing the ball.

Former dribbler design (*Ricardo Alfaro, 2018*) has the null damping action problem: when the ball collided with the roller at a high speed, it would bounce of the roller. The new design seeks to solve this problem based on the model proposed by Tigers Mannheim team (*Andre Ryll, 2018*), which has two rings made of a compressible material on each side of the Dribbler's structure. This allows the whole system to move (two degrees of freedom: upwards and inwards) when in contact with the ball. Our proposed changes point to the simplification of the design and manufacturing process, ensuring that it fits with the structure of the robot (see Fig. 1). The rings on the sides (black rings in Fig. 1a) would be replaced by a oval-shaped ring of silicone, the structure of the dribbler would be made of plastic additive manufacturing (PLA with a FDM 3D printer), the gears which

---

[1]https://ssl.robocup.org/

(a) Dribbler with two degrees of freedom (*Andre Ryll, 2018*)

(b) Our approach with the same principle

Figure 1. Comparison between Tiger's Dribbler design and our proposal

(a) Drive train 2018 version

(b) Drive train 2019 version

Figure 2. Comparison between the former and current drive transmission

transfer the motion from the motor to the roller was replaced with two brass pulleys and an O'ring. The roller is placed at a height of 38 [mm] over the ground, with a diameter of 11 [mm], this means that the ball coverage is not more than 13% of the area of the ball when viewed from above.

Recent tests show that the dribbler's new design improves ball impact absorption and largely suppresses the null dampening action problem[2]. Currently, the height of the dribbler roller is studied to vary the point of contact with the ball and see its effect on the rotation transfer, looking for an optimum that allows greater maneuverability of the robot when controlling the ball. ZJUNlict team (*Zheyuan Huang, 2019*) proposes that the dribbler motor responds to a suitable speed curve to further improve ball control. In terms of the resistance of the system, we have not tested it in a real match environment, nevertheless, lab tests show that the silicone of the roller has no noticeable signs of damage.

Drivetrain

The previous drivetrain design is replaced by an internal gear system with a reduction of 51:17 and which centers the system on the motor shaft, as shown in Fig. 2b. This configuration allows us to bring the wheels closer to the base, which grants more space to the interior of the robot, distributing its internal components more efficiently, with the reduction of its height as proof of this. The gears are manufactured with hardened SAE45 steel to withstand the speed and torque generated by a 50 [W] power motor. The modules of each wheel are identical to each other, which facilitates repairs, modifications, and maintenance during the competition.

The outer diameter of the wheel is reduced from 55 [mm] to 50 [mm], coinciding with that of the gear. Its design corresponds to a it omniwheel that has small perpendicular wheels and centered
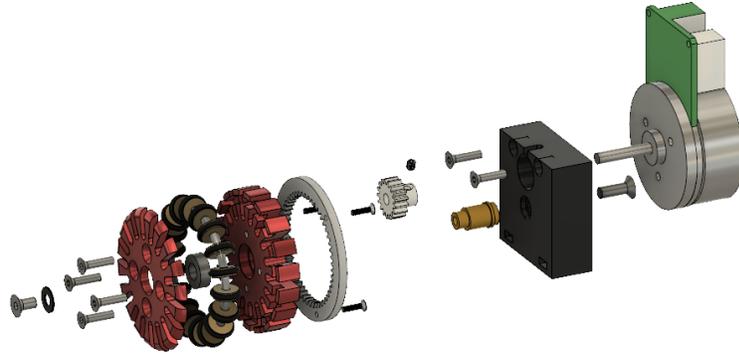
---

[2]https://youtu.be/tgDHXXVo-w4

Figure 3. Drive train's exploded view

on the axis of the perimeter of the main wheel. To obtain smoother rotations[3] the number of small wheels is increased from 15 to 18 concerning the former design, which means a smaller separation between them when the robot is touching the ground. Due these changes the axis of the main wheels is at a distance of 54.5 [mm] to the floor when one of its small wheels is making contact with it, and 53.66 [mm] when two wheels are in contact, versus the 59.5 [mm] and 58.18 [mm] of the former version. In this way, it is estimated that the vertical oscillation of the robot goes from 1.32 [mm] to 0.84 [mm], which is directly reflected in the displacement made by the robot[4]. In order to quantify the changes, we are currently designing a way to measure the impact of vertical vibration caused by this separation in the rest of the systems and movement.

Hardware improvements

For this version of the robot, the architecture of the microcontroller (MCU) is changed to centralize the computing capacity in only one unit, which incorporates the transmission of high-speed data packets, speed control over the motors, measurement from sensors and execution of game actions. As a consequence, the communication system is renewed and the number of sensors arranged throughout the system is increased. The table 1 shows a general comparison of the components of the 2018 generation of the robot and the current version 2019. Some of the hardware is omitted in this section because it has not been validated and will be presented as future work in the conclusions.

Previous architecture was composed of five PIC32MX440F256H MCU's, one of them receiving control packages from the centralized game system. Subsequently, the information is decoded to obtain a speed vector and orders regarding the kicking and driving of the dribbler. The remaining MCU's apply PID control over each of the drive motors, based on a reference calculated by the central microcontroller using the kinematic model of the robot. It is delivered in the form of analog voltage using a digital-analog converter (DAC) MAX5814. For this new version, PIC32 MCU's were replaced by a single high-performance STM32F767BI6 operating at 216 [MHz]. In this case, this MCU is capable of performing the same tasks that multiple MCU's PIC32 architecture carried out in a shorter time. Quantitative comparisons are presented in the next section. Fig. 4 shows a render of the new PCB with some of the components on the surface.

The wireless communication system of the previous version of the robot is based on transparent APC220 transceivers that operate in the 433 [MHz] band. It has a UART TTL interface that reaches a baud rate of 19200, which defines the maximum speed of packet transmission from the centralized system to the robots and forces to maintain unidirectional communication, identified as one of the
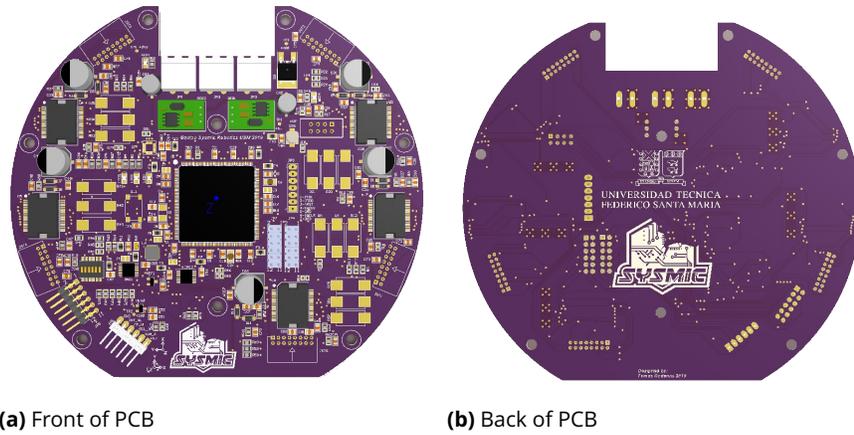
---

[3]https://youtu.be/QuuogsBzJAY
[4]https://youtu.be/Q22TGcXePKI

(a) Front of PCB   **(b)** Back of PCB

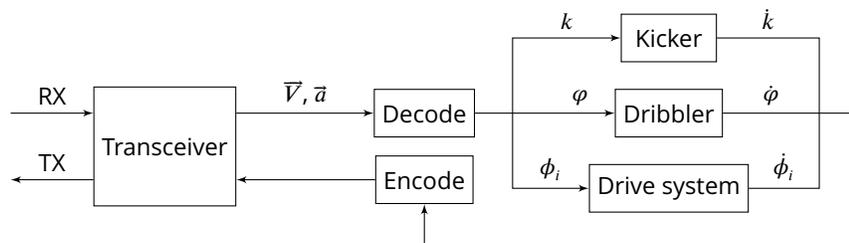**Figure 4.** New PCB render.

Figure 5. Block diagram of robot operation

main problems in the performance of the team on the field. The new communication system is based on the nRF24L01+ transceivers that operate in the 2.4 [GHz] band and reach rates of 2Mbps, increases the rate of data packages sent to the robot during matches. It connects to the MCU through the SPI port and has an embedded protocol that discards packets that do not correspond to the current receiver in addition to the error detection mechanisms. The package with control orders sent from the centralized system encodes a vector of speed references, dribbler speed, and kick drive. The communication protocol and the quantitative improvement of the transmission speed are detailed in the next section.

Maxon MILE 2048 CPT encoders are added for motor speed reading. These are incorporated in the Maxon EC-45 motors, whose power was increased from 30W to 50W. The L6235 drivers are not changed as their characteristics are sufficient to operate both motors. In this way, it is possible to close the speed control loop with a higher sense resolution compared to the 2018 version, improving the frequency of the discrete PID from 16 [Hz] to 1 [KHz].

The new system allows the robot to respond packets to the centralized system, to report the state of the components inside the robot, such as the charge level of the battery, speed readings of driving and dribbler motors, load power of the kick booster, among others. The current implementation has been tested by returning the state of kicking and driving of the dribbler, and the speed readings obtained with the encoders. Fig. 5 shows a block diagram that shows the processes performed internally by the robot from the orders sent by the computer, while Fig. 6 shows the speed control process for the i drive motors of the robot, with $i = 1, 2, 3, 4$.

Firmware and communication

In the firmware programming, new considerations were taken for handling control instructions from the centralized system, due to the high calculation speeds and interaction with MCU modules on each task that in the previous version translated into bottlenecks because of the high dynamism of the matches.
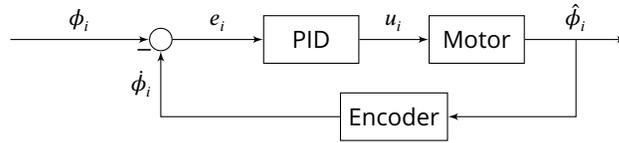
Figure 6. Block diagram of PID loop applied in each motor.

Robot generation	2018	2019
Module	APC220	nRF24L01+
Data rate with device	19200 (UART baud rate)	8Mbps (SPI)
Air data rate	19200 bauds per second	2Mbps
Reception time	38.6 [ms]	68.5 [us]
Complete packet transfer and decoding	38.81 [ms]	474.5 [us]
PID frequency	16 [Hz]	1 [KHz]
Speed measure resolution	18 CPR	2048 CPR
Execution time	590.76 [us]	442[us]

Table 2. Time differences between robot generations

As indicated in the previous section, the APC220 transceiver module is changed to nRF24L01+ of Nordic Semiconductors. The transmission rate of this device is 2Mbps, while the MCU communicates with the device by SPI at a rate of 8Mbps. Time comparisons of the sending and processing control instructions are shown in table 2 and denote a performance increase of around 82 times in the total transaction. Additionaly, a new communication protocol is designed by reducing the size of instruction packets from 60 to 24 bytes in total (4 bytes each robot). Table 3 shows the structure detail of instruction per robot. This protocol is based on a pruned version of the protocol created by ZJUNlict team (*Zheyuan Huang, 2019*), current champion of the SSL league.

byte \ bit	7	6	5	4	3	2	1	0
0		Robot ID			Dribbler strength		Shot	CB
1	±	Speed v_x						
2	±	Speed v_x						
3	±	Speed v_θ						

Table 3. Communication protocol

Once the information has been decoded, to translate it into motion it is necessary to apply some coordination strategy between the components, to ensure integrity in the execution of control loops and communication. Because in practice the centralized system will send data packets with a frequency of 60 [Hz], the PID must be able to operate at a high speed and therefore high sampling rate and high resolution speed sensing is needed. The calculation of speeds by motor and setting analog references to their controllers by DAC reaches an approximate time of 170 [µs] total, which allows to apply the control loops for each of the motors with a frequency of 1 [KHz]. Table 2 shows quantitative comparisons with respect to the previous version of the robot.

Firmware coding is done using FreeRTOS[5], an operating system kernel developed to be used in MCU's and offering a large section of OS features, such as tasks scheduling and assignment of priorities, queues, semaphores, mutexes, among others. In this way it is possible to set the period of operation of the PID exactly and the detection of packets with fixed time. Although, it is possible to use different options for these tasks such as timers, this option allows the scalability of the

---

[5]https://www.freertos.org

system much more simple and modular.

Software

A modified version of the RoboJackets team software[6] has been used so far by our team, replacing its radiofrequency system to make it compatible with the APC220. This software is responsible for performing image processing, obtaining the current state of the game, determining the actions to be performed by the robots and the composition of data packets to be sent once all the calculations have been made. We are currently developing an independent software repository that seeks to solve various compatibility problems generated by RoboJackets software on our complete system. Due to its early stage, it is interpreted as future work to develop.

Recently a base station has been designed, composed of an STM32F767 microcontroller and an nRF24L01+ that communicates with the centralized system using USB Full Speed protocol at a rate of 12Mbps. A test client developed by the software team sends commands from a joystick connected to the computer and allows movement tests on one robot. The base station allows bidirectional communication and therefore it is possible to monitor internal variables of the robots. Later it will be incorporated into the new software of the team.

Conclusions

Laboratory tests empirically validate the changes we have made between versions; the low center of gravity and the reduction in height has shown movements with greater stability largely thanks to the new design of the drivetrain that also allows a more efficient distribution of the internal parts; the reduction of the diameter of the wheels and the increase of small wheels in its perimeter the movement of the robot is much smoother, also attributable to the addition of the high precision encoder and the increase of speed in the speed control loop; the renewal of a high-speed communication system has allowed receiving packages at a higher rate and along with this a firmware has been developed that is capable of carrying out all tasks in a coordinated manner, and; the new hardware design allows the correct execution of all the necessary tasks centralizing the computation in a single high-performance MCU that handles multiple actuators and sensors that were added in this version.

All areas of development in the team have future challenges that are under development and were not incorporated throughout this article:

- The new voltage booster circuit for kicking is being set up and validated from a boost to flyback topology. According to our design the loading speeds should be considerably increased.
- Accelerometers and gyroscopes have been added to the PCB and are currently working on the model that allows us to estimate the position of the robot independently and in cases of intermittent centralized system being able to coordinate work between the robots themselves.
- The independent game client of the Sysmic team is being developed looking for greater compatibility to our platforms, from the user interface to the game action selectors. Along with this, we seek to optimize the selection of actions based on reinforcement learning algorithms and neuroevolutive topologies. The feasibility of generating plays from these algorithms is being investigated as well and compare the performance obtained with respect to the manually programmed moves.

References

Andre Ryll CCNO Mark Geiger. TIGERs Mannheim - Extended Team Description for RoboCup 2018. . 2018; .

Ricardo Alfaro ea. AIS: Artificial Intelligent Soccer. Proceedings of the 4th Congress on Robotics and Neuroscience. 2018; .

Zheyuan Huang ea. ZJUNlict - Extended Team Description for RoboCup 2019. . 2019; .

---

[6]https://github.com/RoboJackets/robocup-software