

ITAndroids Small Size League Team Description Paper for Robocup 2026

Amadeu Albuquerque Fontenelle Neto, Andrei Freiburger,
Arthur Colvet de Souza, Artur Dantas Ferreira da Silva,
Bernardo Freire Bandeira, Cícero Víctor Souza Crisóstomo,
Filipe Avelino Brasil Nicolau de Oliveira, Lucas Guimarães Neves,
Marcelo Roncato Junior, Marcos R. O. A. Máximo,
Valerio Augusto Neri da Rocha Barros

Autonomous Computational Systems Lab (LAB-SCA)
Aeronautics Institute of Technology (ITA)
São José dos Campos, São Paulo, Brazil
Contact: itandroids-small@googlegroups.com
Website: <http://www.itandroids.com.br>

Abstract. ITAndroids is a robotics competition group associated with the Autonomous Computational Systems Lab (LAB-SCA) at the Aeronautics Institute of Technology (ITA). In the current development cycle, efforts in electronics included enhancements to the kicker control system and the integration – though it is still in the testing period – of an IMU with Kalman filtering for improved motion control. Mechanical developments targeted improved ball control through a new damping system, infrared sensor support and the ongoing development of a chipper mechanism for the goalkeeper. On the software side, the team implemented coordinated passing strategies for free-kick situations and continued refining robot positioning algorithms. This paper presents an overview of these developments and outlines future directions toward RoboCup 2026.

1 Introduction

ITAndroids is a multidisciplinary robotics research group formed by about 70 students from different undergraduate engineering courses at the Aeronautics Institute of Technology (ITA), based on the city of São José dos Campos, São Paulo, Brazil. Currently, about 20 of these are in the SSL team. Over recent years, the team has steadily consolidated its presence in national and international competitions, emphasizing continuous technical development and system integration.

In the current development cycle, the team concentrated on targeted improvements aimed at increasing the reliability, precision and integration of its robotic systems. In electronics, efforts focused on kicker control refinements, inertial sensing integration with Kalman filtering and communication protocol updates to support improved motion estimation. Mechanical developments prioritized ball control and shot consistency through focused redesigns, while software

advances expanded strategic capabilities, particularly in coordinated behaviors during set-play situations.

This paper is organized as follows. Section 2 presents the main developments in electronics, including kicker control, inertial sensing and communication updates. Section 3 discusses recent mechanical improvements and ongoing design efforts. Section 4 describes the software developments related to strategy and coordination. Finally, the paper concludes with a discussion of current challenges and future directions toward RoboCup 2026.

2 Electronics

2.1 Kick force

The microcontroller firmware currently deployed on the team’s Mainboard operates under a FreeRTOS-based task management system [4, 6]. Until approximately the middle of last year, the task responsible for actuating the kicker did not support force modulation, that is, to control the amount of energy discharged from the capacitors onto the solenoid. Although a continuous floating-point value ranging from 0 to 10 was received via radio to represent the desired kick intensity, the firmware always commanded a full discharge of the capacitors, resulting in maximum kick power in all situations.

After internal discussions and exchanges with other teams in the league, such as RoboCin [14, 15], it was decided to implement, as an initial solution, three discrete kicking force levels. Although a continuous force scale remains a long-term objective, small differences in machining and assembly between robots tend to compromise the predictability and repeatability of the mechanical response of the kicking system. As a result, the communication protocol continues to accept values in the 0-10 range, while the firmware classifies this input into three intervals: values between 0 and 3.3 correspond to a weak kick, values between 3.3 and 6.7 to an intermediate kick and values above 6.7 to a strong kick.

These categories are subsequently mapped, in firmware, to different short-circuit durations between the capacitors and the solenoid. To establish this mapping, an experimental characterization of the capacitor discharge was carried out using an oscilloscope. Figure 1 presents the measured voltage across the two capacitors (2.2 mF each), which were charged to a maximum voltage of approximately 176 V, as reported by the oscilloscope. Assuming the existence of a delay between the firmware command that initiates the discharge and the effective electrical response of the circuit, the discharge behavior was modeled as an exponential decay preceded by a short initial plateau, as described by Equation (1). Based on the experimental data, the model parameters were estimated using the Nelder-Mead optimization [9] algorithm, as implemented in the R statistical computing environment [12]. The resulting fitted curve is shown in Figure 1. The results suggest the presence of an initial plateau lasting approximately 1 ms, followed by an approximately exponential voltage decay.

$$V(t) = \begin{cases} V_{\max}, & t < \Delta t \\ V_{\max} e^{-\alpha(t-\Delta t)}, & t > \Delta t \end{cases} \quad (1)$$

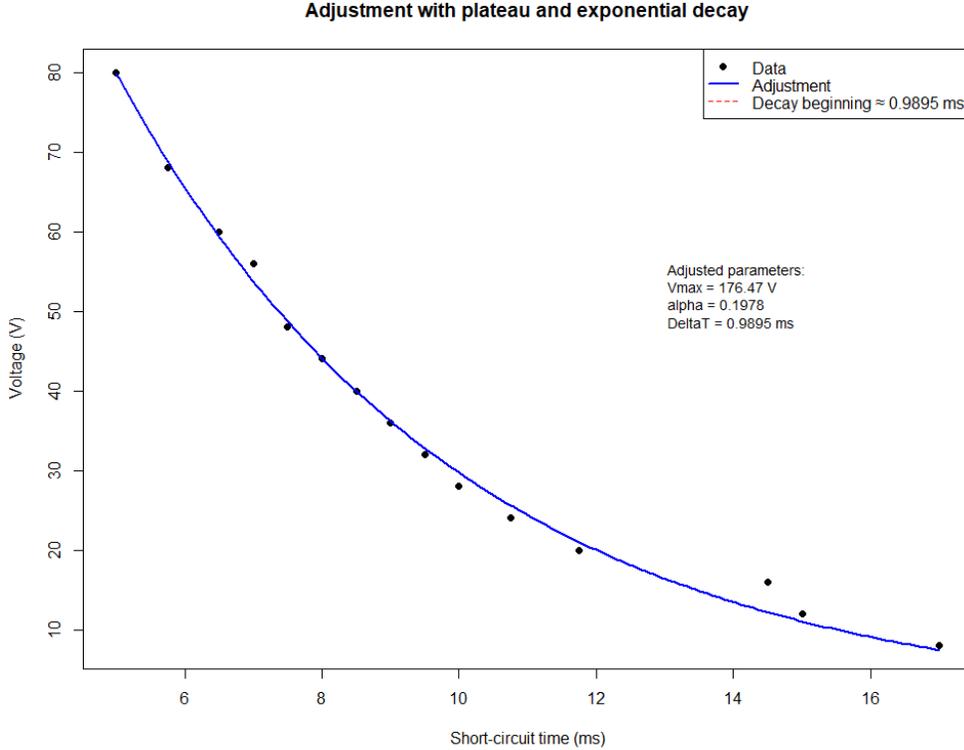


Fig. 1: Mapping the voltage across the capacitors as a function of the short-circuit time between them and the solenoid commanded in firmware. Optimized parameters of Equation (1) with Nelder-Mead algorithm: $V_{\max} = 176.47 \text{ V}$, $\alpha = 0.1978$ and $\Delta t = 0.9895 \text{ ms}$.

The experimental data further indicate that the discharge is nearly complete after approximately 15 ms. Based on this observation, a short-circuit duration of 9 ms was arbitrarily associated with the intermediate kicking force. However, the Kickerboard – which is in its second generation since 2024 [5] – exhibited recurring failures of a critical MOSFET whenever the firmware commanded short-circuit durations shorter than 5 ms. Although the underlying mechanism has not yet been fully characterized, this behavior is suspected to be related to the response times of the microcontroller GPIOs and a possible temporal overlap be-

tween charging and discharging commands. Given the short time scales involved, commanding the discharge, waiting less than 5 ms, and subsequently terminating the short-circuit may not be interpreted by the firmware-circuit system as strictly sequential events, potentially leading to simultaneous conduction paths and component overload.

Benchmark measurements of capacitor voltage were performed as a function of the received radio command, confirming the expected operation of the discrete force mapping. Nevertheless, tests conducted with the ball positioned in front of the robot revealed that even the supposedly weak kick still delivers high power, with little perceptible difference between force levels. Consequently, alternative control strategies that do not rely exclusively on short-circuit duration have been investigated, such as direct control of the capacitor charging or voltage, aiming to safely access the region corresponding to short-circuit times of less than 5 ms.

Fortunately, the Kickerboard incorporates an analog-to-digital (ADC) capable of measuring the capacitor voltage [1]. Firmware-level tests are currently being conducted to reliably acquire these measurements. The long-term objective is to replace time-based discharge control with voltage-based control [13, 15], enabling more effective and safer modulation of the kicking force by RoboCup 2026.

2.2 Kalman filters and IMU

The most significant recent development in the team’s electronics, although still under experimental validation, was the integration of an LSM6DS33 inertial measurement unit (IMU) in the Mainboard [2], together with the implementation of Kalman filters in the firmware – integrated into the FreeRTOS task responsible for motor control. The primary objective of this modification is to improve the accuracy of robot motion, particularly orientation control during kicking maneuvers.

At present, the radio communication protocol (multirobot command) consists of 32-byte messages, corresponding to the maximum payload capacity of the nRF24L01+ module [11]. As described in Table 1 and in the 2022 TDP [8], the first byte of the message, called Instruction Identifier (II), specifies the type of command transmitted. Since robot-to-station feedback has not yet been implemented, this field currently assumes a single value (0x01), indicating commands sent exclusively from the station to the robots. The II byte is followed by three Robot Command (RC) packets, each including one Command Flags byte (containing robot ID and activation flags for devices such as kicker, chipper and dribbler), six bytes encoding the tangential, normal and angular velocities in the robot’s local coordinate system, two bytes associated with the kicker and chipper forces and one byte specifying the dribbler motor speed – the dribbler and chipper subsystems are unfortunately not yet implemented due to space constraints in the mechanical design of the current generation of robots. The final byte of the message is the Checksum, computed and verified in firmware to ensure that no corrupted or incomplete messages are read by each robot.

Table 1: Small Size communication protocol – ITAndroids

Multirobot command	Instruction Identifier	Robot X Command	Robot Y Command	Robot Z Command	Checksum
Bytes	1	10	10	10	1

Table 2: Robot Command (RC) packet diagram

RC packet	Command Flags	Tangent Speed	Normal Speed	Angular Speed	Kicker Power	Chipper Power	Dribbler Speed
Bytes	1	2	2	2	1	1	1
Type	–	signed	signed	signed	unsigned	unsigned	unsigned

In the current protocol, the team’s software computes the tangential, normal and angular speed commands required to execute game strategies based on the ball and robot positions provided by the vision system. In the Mainboard firmware, these commands are converted into individual speeds for each of the robot’s four wheels. However, noise and parallax effects affect the angle estimation provided by the camera and render this information unreliable in certain situations, particularly during precise aiming tasks such as goal-oriented shots. The introduction of the IMU enables a more accurate estimation of the robot’s angular velocity, thereby reducing this limitation.

Under the new proposal, the logic for calculating velocities in the robot’s local coordinate system is shifted from software layer to firmware. The software is now responsible only for estimating global velocities in the field reference frame (V_x and V_y), acquiring the robot’s orientation from the vision system and defining a desired orientation according to the game strategy. This architectural change requested a reformulation of the communication protocol. Table 3 presents the proposed new RC packet format, which now includes global velocity components as well as the observed and desired angle. Due to the 32-byte maximum payload limitation imposed by the nRF24L01+ radio module, the kicker and chipper force commands were combined into a single byte and subsequently separated in firmware. Although this modification reduces the resolution of these variables, the impact is considered negligible, since, as discussed in the previous section, the team is not currently pursuing a continuous kicking force scale.

Table 3: Robot Command (RC) packet diagram – New protocol

RC packet	Command Flags	V_x Speed	V_y Speed	Observed Angle	Desired Angle	Kick-Chip Power	Dribbler Speed
Bytes	1	2	2	2	1	1	1
Type	–	signed	signed	signed	signed	unsigned	unsigned

Following the installation of the IMU and the modification of the communication protocol, it was first necessary to develop a simplified STM32 library for sensor configuration and data acquisition. While the long-term goal is to expand this library and release it as open source, its current implementation is limited to device initialization and angular velocity measurements. Based on this infrastructure, a state estimation architecture composed of Kalman filters was introduced to improve both the robot orientation estimate and the wheel velocity measurements used by the motor controllers. The proposed approach comprises one Angle Filter (AF) and four independent Motor Filters (MFs), one for each wheel.

Angle Filter (AF) The Angle Filter aims to improve the estimation of the robot’s orientation by fusing angular velocity measurements from the IMU with orientation data obtained from the vision system. The filter state corresponds to the robot’s orientation angle, denoted by ϕ . The prediction model of the AF is derived directly from the kinematic relationship between angular velocity and orientation. In discrete time, the orientation at iteration $k+1$ is approximated as the orientation at iteration k plus the instantaneous angular velocity measured by the IMU ($\omega[k]$) multiplied by the control sampling period T_C :

$$\phi[k+1] = \phi[k] + \omega[k] \cdot T_C. \quad (2)$$

Since the motor control loop operates at 200 Hz, the sampling period is $T_C = 0.005$ s. The observation model of the AF uses the orientation measured by the vision system, which is transmitted to the firmware via the radio message as the Observed Angle field described in Table 3:

$$z_{AF}[k] = \phi_o[k]. \quad (3)$$

From these models, the standard one-dimensional Kalman prediction and update steps for the AF are obtained. The prediction step computes an *a priori* estimate of the orientation:

$$\phi^- [k] = \phi^+ [k-1] + \omega[k] \cdot T_C, \quad (4)$$

while the update step incorporates the vision-based observation using the Kalman gain K_{AF} :

$$\phi^+ [k] = \phi^- [k] + K_{AF} \cdot (\phi_o[k] - \phi^- [k]). \quad (5)$$

The resulting filtered orientation estimate is subsequently used by the motion control layer.

Motor Filters (MF) In addition to orientation estimation, wheel velocity measurements are also filtered to improve motor control performance. For this purpose, four identical one-dimensional Motor Filters are implemented, each corresponding to one wheel of the robot. The state of each MF corresponds

to the angular velocity ω_i of wheel i . The prediction model is based on a linear discrete-time approximation of the wheel dynamics, where the angular velocity at the next iteration depends on the previous velocity and on the voltage command u_i generated by the PI controller:

$$\omega_i[k+1] = a_d \cdot \omega_i[k] + b_d \cdot u_i[k]. \quad (6)$$

The constants $a_d = 0.9036$ and $b_d = 1.2791$ are derived from the mechanical and control model adopted by ITAndroids. The observation model for each Motor Filter uses the raw angular velocity ω_{H_i} measured by the Hall sensors attached to each motor:

$$z_{MF_i}[k] = \omega_{H_i}[k]. \quad (7)$$

Following the Kalman filter formulation, the prediction step for each Motor Filter computes an *a priori* estimate of the wheel angular velocity:

$$\omega_i^-[k] = a_d \cdot \omega_i^+[k-1] + b_d \cdot u_i[k-1], \quad (8)$$

and the update step refines this estimate using the Hall sensor measurement and the Kalman gain K_{MF} :

$$\omega_i^+[k] = \omega_i^-[k] + K_{MF} \cdot (\omega_{H_i}[k] - \omega_i^-[k]). \quad (9)$$

Integration into the control architecture Summarizing the proposed architecture, the global velocity components together with the observed and desired angles are transmitted to the firmware via radio messages. After obtaining an improved orientation estimate through the prediction step of the Angle Filter, a proportional controller generates the robot’s desired angular velocity. This replaces the previous approach, in which the angular velocity was directly commanded by the software based solely on camera measurements. Next, a sequence of matrix operations transforms the global velocities into local ones and subsequently into reference angular speeds for each wheel. These reference values are provided to the motor control loop. For each wheel, the corresponding Motor Filter processes the raw Hall sensor measurements to obtain filtered angular velocity estimates, which replace the unfiltered measurements previously used by the PI controllers. The resulting control signals are then converted into voltage commands applied to the motors.

At the time of writing, comprehensive field tests required for full system validation and accurate estimation of filter covariance matrices have not yet been conducted. Moreover, the communication protocol changes imply substantial modifications to the team’s software architecture, which are still under development. These steps are expected to be completed during the preparation phase leading up to RoboCup 2026.

2.3 Updates on the new version of Mainboard

Over the past few months, the team has been developing a new revision of the Mainboard with the objective of addressing known issues in the current design and incorporating additional features. Although progress was limited over the last year due to the prioritization of other demands, this effort has recently regained momentum. In the coming months, updates to the board layout in Altium Design and the selection of suitable components from a firmware perspective are expected to be finalized. Planned improvements include the integration of a second radio module to enable robot-to-station feedback, the addition of temperature sensors for the motors, pinout corrections, the inclusion of a buzzer for debugging purposes and the definition of a more comprehensive testing protocol aimed at early fault detection, among other enhancements.

2.4 Radio station

As reported in the previous TDP [10], the team’s most recent radio station has demonstrated robust operational performance. Building upon the existing game mode selection mechanism, several new features have been proposed. One such project, which depends on the integration of a second radio module in the new Mainboard, involves transmitting feedback data from the robot to the station, including motor states, temperatures and relevant electrical quantities such as currents and voltages. This information could be processed in real time to detect or even anticipate electronic and mechanical failures during matches. Development of this functionality is expected to begin after the release of the new Mainboard revision.

Another proposal under consideration, with potential implementation by RoboCup 2026, is the development of a firmware testing mode that does not rely on commands from the team’s software. While the current Mainboard already includes buttons for basic motor and kicker tests, more complex maneuvers, such as robot translations and rotations, still require a computer containing the software code connected to the station via USB. The proposed testing mode would allow manual selection of predefined commands via the station’s button, saving a computer from having to run the software solely for bench testing.

In spite of that, the definitive adoption of Kalman filters and the revised communication protocol introduces additional challenges for this testing mode. In particular, rotational commands would require the firmware to simulate orientation measurements equivalent to those provided by the vision system in order to correctly feed the filters. One possible solution involves introducing a protocol flag in the radio messages to indicate a testing mode, allowing the firmware to bypass filtering and revert to the unfiltered model. Alternative solutions may be explored and refined throughout the development process leading up to RoboCup 2026.

3 Mechanics

In the current development cycle, the mechanical design of the ITAndroids robot has focused on targeted improvements aimed at increasing ball control reliability and shot precision. Rather than large structural overhauls, the emphasis has been on refining specific subsystems that directly impact in-game performance and robustness. Notable efforts include the final version of a new damping system to improve ball reception, the integration of dedicated mechanical support for infrared sensing to enhance shooting consistency, and the ongoing development of a specialized chipper mechanism tailored for the goalkeeper robot. Together, these updates reflect a design philosophy centered on modularity, functional optimization, and incremental mechanical evolution aligned with the team's strategic objectives.

3.1 Damping System

A new damping solution was developed using EVA foam, leveraging the fact that the roller can now play a more active role in damping, since it does not need to move, as shown in Fig 2. Although not an ideal long-term solution, initial test results have been promising, with the ball coming to a complete stop upon contact with the robot. However, a drawback of this approach is its dependence on ground friction for proper functioning, meaning that small variations in the field carpet can significantly impact the system's efficiency.

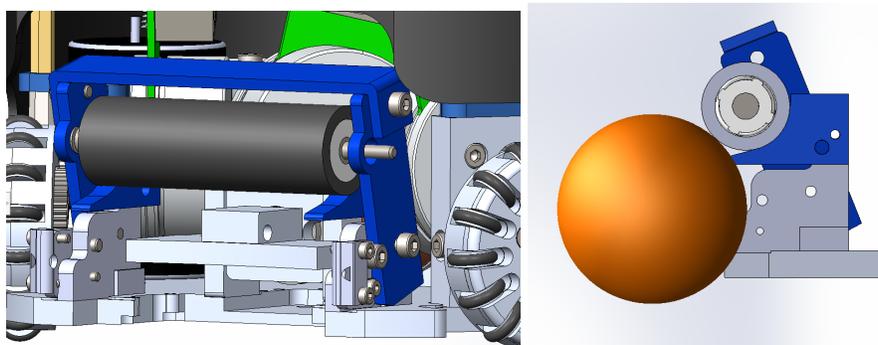


Fig. 2: New damping system.

3.2 Infrared (IR) Support

To enhance shooting precision, an infrared (IR) sensor was implemented, enabling the robot to better detect ball presence and positioning. A custom support structure was designed and 3D-printed in PLA to house the sensor securely,

ensuring correct alignment with the ball path. This support structure, shown in Fig. 3, was designed to be compact and lightweight, minimizing interference with other internal components while still providing the necessary stability. Its position was also chosen to avoid obstruction of the kicker mechanism and maintain visibility of the ball during motion.

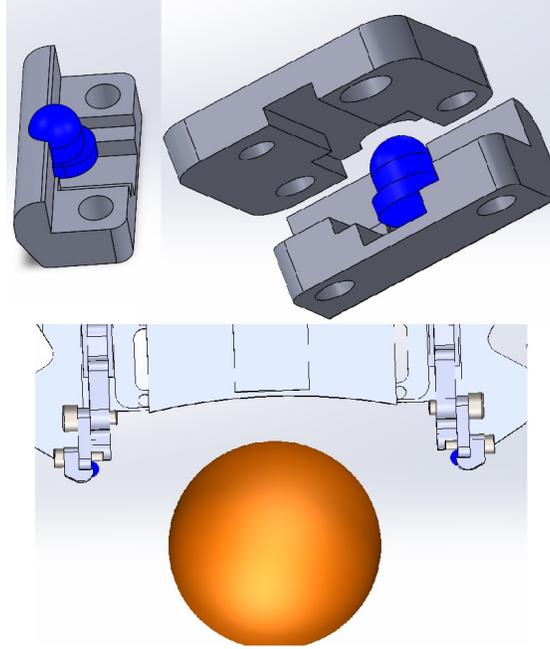


Fig. 3: Infrared sensor support.

3.3 Chipper Mechanism

A new chipper mechanism is being planned specifically for the goalkeeper robot, aiming to enable chip passes as part of the team's strategic development. The design was conceived to take advantage of the space originally allocated for the kicker, as shown in Fig. 4.

Due to spatial and resource limitations of the current robot generation, it was not feasible to integrate both the chipper and the kicker in a single unit. Consequently, this chipper implementation is exclusive to the goalkeeper unit. Despite the added functionality, the current prototype has not yet reached the desired pass strength and reliability. This highlights the need for further development, especially regarding volumetric optimization to reposition the chipper mechanism for better efficiency and force transmission.

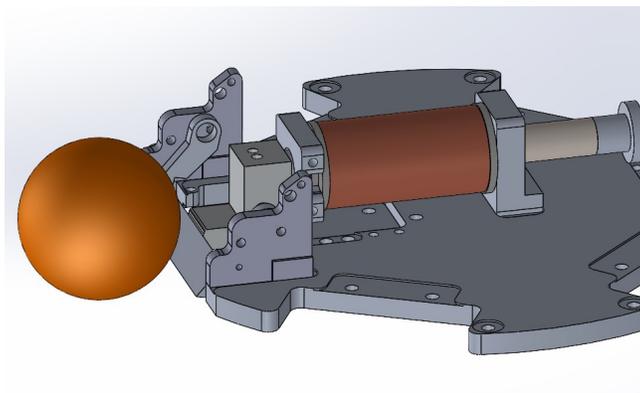


Fig. 4: Tested chipper design for goalkeeper.

3.4 Future planning

We have initiated the volumetric analysis process. This crucial phase consists of a detailed analysis of the design's dimensions, motion envelope, and mass properties, using the 3D model as a basis. The objective is to validate the occupied space, ensure the absence of interference between moving and static parts, and extract precise data that will serve as essential parameters for the development of the robot's control systems, autonomous navigation, and task planning.

4 Software

In this section we dive into our efforts related to our software in a general manner.

4.1 Free kick pass

Previously, our robots were unable to pass the ball between one another under any circumstances. As a result, during free kicks, our only option was to shoot directly at the goal, which often led to stagnant game states. To address this limitation, we implemented a passing mechanism that allows the ball to be passed to another robot specifically during free-kick situations.

Once a free kick is conceded to our team, we check if there are enough players for a pass to occur and if the goal visibility is blocked for the robot taking the free kick. If both of those conditions are met, a pass will be executed.

The passing procedure works as follows: the robot designated to receive the pass enters a receive-pass behavior, which involves switching to a dedicated Delaunay-based positioning configuration [3] and positioning itself in anticipation of the incoming ball. Meanwhile, the robot responsible for taking the free kick remains in a hold state until the receiving robot reaches the appropriate position and signals that it is ready to receive the pass. Once this signal is received, the pass is executed.

This approach has been successfully validated in training sessions and will be tested in official matches for the first time at RoboCup 2026.

4.2 Future planning

We are in the process of refactoring the way robot positions are assigned within our Delaunay algorithm. An implementation involving the Hungarian algorithm [7] is nearly complete and our objective is to have more dynamic and versatile robot positioning during matches.

5 Conclusion and Future Work

This paper described the main developments carried out by the ITAndroids Small Size League team during the most recent development phase toward RoboCup 2026, covering advances in electronics, mechanics and software. The work carried out reflects the team's focus on incremental improvements, system reliability and tighter integration between hardware and software layers, while building upon the experience accumulated in previous competitions.

In electronics, advances in kicker control, inertial sensing, Kalman filtering, and communication protocol design aimed to improve motion accuracy and system robustness. Mechanical efforts concentrated on enhancing ball control and shot consistency through targeted redesigns, while software developments expanded the team's strategic capabilities, particularly in coordinated behaviors during set-play situations. Together, these improvements contribute to a more stable and adaptable robotic platform.

Although several of the proposed solutions are still under validation or ongoing development, they establish a solid foundation for future refinements. Upcoming work will focus on extensive field testing, protocol consolidation, further mechanical optimization and deeper integration of sensing and control strategies. Through these efforts, the team aims to continue strengthening its performance and competitiveness in the Small Size League at RoboCup 2026 and beyond.

Acknowledgment

We would like to acknowledge the RoboCup community for sharing their developments and ideas. We especially acknowledge RoboFEI, Skuba and TIGERs for open sourcing their electronic and mechanic designs. Moreover, we would also like to thank members of CMDragons, RoboCin, Warthog and RoboIME for helping in various contexts. Finally, we thank our sponsors Altium, CENIC, EDGE, ITAEx, MathWorks, Nubank, Rapid, Siatt, Solidworks, ST Microelectronics and Visiona.

References

1. Adc122s051 2-channel, 200 ksps to 500 ksps 12-bit a/d converter datasheet. Texas Instruments (2013), <https://www.ti.com/lit/ds/symlink/adc122s051.pdf>, rev. E, Nov. 2004, Revised Mar. 2013
2. Lsm6ds33 inemo inertial module: always-on 3d accelerometer and 3d gyroscope – datasheet. Pololu / STMicroelectronics (2015), <https://www.pololu.com/file/0J1087/LSM6DS33.pdf>, docID027423 Rev. 4, Oct. 2015
3. Akiyama, H., Noda, I.: Multi-agent positioning mechanism in the dynamic environment. In: RoboCup 2007: Robot Soccer World Cup XI, Lecture Notes in Computer Science, vol. 5001, pp. 377–384. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-68847-1_38
4. Freiberger, A., Maximo, M.R.O.A., Loubach, D.: Implementation of firmware based on real-time operating system for small size league robot. Autonomous Computational Systems Lab, Aeronautics Institute of Technology, São José dos Campos, SP, Brazil
5. Gonçalves, A., Freiberger, A., Martins, A., Penna, A., Moreira, B., Yamamoto, B., Simplício, E., Morais, E., Pironi, G., da Hora, G., Mendes, L., Neves, L., Zoppi, L., Maximo, M., Martins, M., Egger, M., Pedroso, M., Farias, N., Barros, V., Wakugawa, V., Queiroz, Y.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2023
6. Gonçalves, A., Freiberger, A., Sousa, A., Souza, A., Silva, A., Yamamoto, B., Morais, E., Filho, F., da Hora, G., Barbosa, J., Santos, J., Neves, L., Barros, L., Fusinato, L., Alves, L., Mendonça, L., Máximo, M., Castro, M., Viana, M., Pedroso, M., Farias, N., Façanha, P., Barros, V., Sousa, Y.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2024
7. MacAlpine, P., Stone, P.: Prioritized role assignment for marking. In: Behnke, S., Lee, D.D., Sariel, S., Sheh, R. (eds.) RoboCup 2016: Robot Soccer World Cup XX, pp. 306–318. Lecture Notes in Artificial Intelligence, Springer Verlag, Berlin, Germany (2017)
8. Maranhão, A., Schuch, A., Freiberger, A., Albani, A., Rodrigues, A., Simplício, E., Benício, G., Haddad, L., Zoppi, L., Rezende, L., Maximo, M., Egger, M., Moreira, M., Silva, M., Celente, R., Lima, R., Augusto, V., Wakugawa, V.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2022
9. Nelder, J.A., Mead, R.: A simplex method for function minimization. *The Computer Journal* **7**(4), 308–313 (1965)
10. Neto, A.A.F., Freiberger, A., de Souza, A.C., da Silva, A.D.F., Yamamoto, B.D., dos Santos, J.P.S., Lima, J.M.A., dos Santos, J.V., Neves, L.G., de Almeida Santos, M.R., Júnior, M.R., Máximo, M.R.O.A., Feitosa, M.L.M., Borges, M.F.R., Farias, N.F., da Rocha Barros, V.A.N.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2025
11. Nordic Semiconductor: nRF24L01 Single Chip 2.4GHz Transceiver (July 2007), https://cdn.sparkfun.com/assets/3/d/8/5/1/nRF24L01P_Product_Specification_1_0.pdf, rev 2
12. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2024), <https://www.R-project.org/>
13. Ryll, A., Ommer, N., Andres, D., Klostermann, D., Nickel, S., Pistorius, F.: TIGERs Mannheim Team Description for RoboCup 2013

14. Silva, C., Martins, F., Machado, J.G., Cavalcanti, L., Sousa, R., Fernandes, R., Araújo, V., Silva, V., Barros, E., Bassani, H.F., de Mattos Neto, P.S.G., Ren, T.I.: Robôcin 2019 team description paper
15. de Souza, C.O., da Silva, C.V.S., Martins, F.B., da Silva, J.G.M., Santos, L.H.C., Bezerra, R.S., Fernandes, R.C., dos Santos Araújo, V.H.S., da Silva, V.B.A., de França Bassani, H., da Silva Barros, E.N.: Robôcin 2018 small size league team description paper