RobôCIn Small Size League Extended Team Description Paper for RoboCup 2026

Alberto Franca, Brenda Alencar, Davi C. Barbosa, Davi Dubeux, Davyd Wendys, Driele Xavier, Felipe N. A. Pereira, Gabriel Alves, Guilherme V. Nova, Heitor Cordeiro, Jessyca Ferreira, João G. Melo, João Pedro Déo, José Otávio Maciel, Lucas H. Cavalcanti, Lucas Leal, Luiz Felipe, Marcela Asfora, Mateus Albuquerque, Mateus Espíndola, Matheus Paixão, Matheus Stepple, Matheus Vasconcelos, Onias C. B. Silveira, Victor Araújo, Vinícius Fialho, and Edna Barros

Centro de Informática, Universidade Federal de Pernambuco.
Av. Prof. Moraes Rego, 1235 - Cidade Universitária, Recife - Pernambuco, Brazil.
robocin@cin.ufpe.br
https://robocin.com.br/

Abstract. RobôCIn has participated in the RoboCup Small Size League since 2019, won its first world title in 2022 and a second in 2023 (Division B), and is currently a six-time Latin-American and Brazilian champion. In 2024, we made our debut in Division A at RoboCup 2024 in Eindhoven, Netherlands, finishing in fourth place and receiving the Most Improved Team Award. Building upon this result, we improved our overall performance and achieved third place at RoboCup 2025 in Salvador, Brazil. This paper presents our improvements to maintain a high level of performance in Division A at RoboCup 2026 in Incheon, South Korea. During 2025, our team has successfully published two papers at the 28th RoboCup International Symposium. Over the last year, we focused on improving robot navigation, initiating the development of our dribbler system, and advancing our electronics platform. On the software side, we introduced improvements to the pass decision-making and robot positioning systems, aiming at more effective ball circulation and offensive play.

Keywords: RobôCIn · RoboCup 2026 · Robotics · Small Size League

1 Hardware

This year's changes focused on achieving long-term stability and resolving reliability issues. To achieve this, we developed a wheel redesign, a new functional prototype of the dribbling device, a coil and plunger redesign for the kicker mechanism. Consequently, minor modifications were made to the robot's base chassis to accommodate the mounting points and geometry of these new components.

Additionally, we have changed the robot's cover design to comply with the new color-changing rule.

Alongside these structural and functional updates, the robot's overall hardware configuration was consolidated and updated. Table 1 summarizes the main mechanical and electronic specifications of the current robot version, providing a reference for the implemented changes.

Table 1. Robot specification comparison between v2024 and v2025

Robot version	**v2024**	**v2025**
Driving Motors	Maxon EC-45 flat - 50W	Maxon EC-45 flat - 50W
Max % ball coverage	17.91%	20.00%
Microcontroller	STM32F767ZIT6	STM32F767ZIT6
Gear Transmission	1:1	1:1
Gear Type	Not used	Not used
Wheel	3D Printed and aluminum	3D printed
Total Weight	2.50kg	2.36kg
Dribbling Motor	Not used	Maxon EC-max 22, 25W
Encoder	MILE 2048 CPT	MILE 2048 CPT
Dribbling Gear	Not used	1:1
Dribbling Bar Diameter	13mm	13mm
Max. Kick Speed	6.5m/s	6.5m/s
Communication Link	nRF24L01+	WiFi 5GHz
Battery	LiPo 2300mAh 4S 35C	LiPo 2300mAh 4S 75C

1.1 Wheel Design

For RoboCup Salvador, we focused on upgrading the wheel rollers. Since the team's first SSL appearance in 2018, the robots have used the same roller design: a prefabricated brass/steel hub with a standard O-ring, rotating around a continuous circular shaft that encircles the entire wheel hub. This design presented several critical drawbacks:

- **High Friction:** The direct contact between the continuous axis and the roller generated excessive friction, reducing motor efficiency.
- **Noise and Vibration:** The friction resulted in significant noise during operation. Furthermore, the small cross-sectional diameter of the O-ring required a high number of rollers per wheel to maintain continuous ground contact. Nevertheless, it still resulted in noticeable vertical vibrations during movement.

The new roller design, shown in Figure 1, was inspired by the mechanics proposed by the TIGERS Mannheim team [10]. We replaced the standard O-ring with an X-ring (quad-ring) to increase the contact surface area and improve traction. Additionally, we integrated a miniature ball bearing inside each roller to eliminate the direct axis-to-hub friction. The results exceeded our expectations, significantly reducing rolling resistance and eliminating the parasitic movements previously observed, particularly along the robot's local Y-axis.

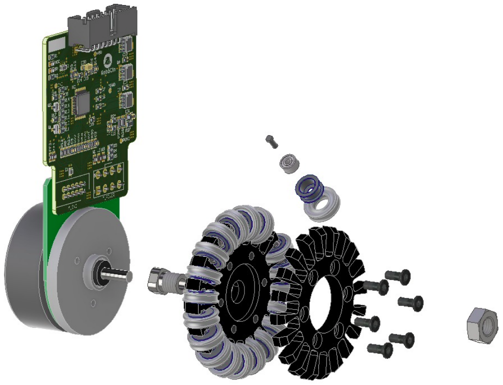


Fig. 1. Exploded view of the new roller design highlighting the internal bearing and X-ring profile.

Despite the functional success, we faced manufacturing challenges. The cost to machine the high volume of rollers required for the entire fleet was prohibitive. As a compromise, we optimized the roller geometry for additive manufacturing and 3D printed the wheel hubs. This introduced assembly tolerances that were difficult to control; minor assembly imperfections in wheels caused kinematic errors. To mitigate this, we plan to manufacture high-precision machined aluminum wheels for the upcoming competition in South Korea.

1.2 Dribbling Device

This year, we successfully developed a functional prototype for the dribbling system. However, due to a shortage of motors (Maxon EC-max 22 25W), we could not equip the entire fleet with active dribblers for offensive plays. Instead, the device was deployed on select robots primarily for pass-damping and ball reception.

The new dribbler features a lateral damping suspension system and a roller bar coated in Polyurethane (PU) with a Shore 65A hardness, as depicted in Figure 2.

This configuration allowed the robot to effectively absorb and dominate passes with velocities up to 5 m/s. Similar to the wheels, the dribbler's structural components were 3D printed to speed up prototyping. This reliance on printed parts required frequent manual adjustments and the reprinting of components

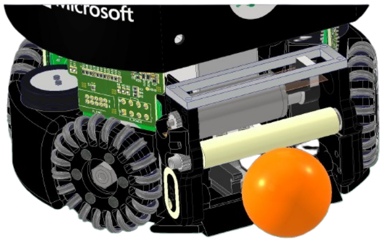
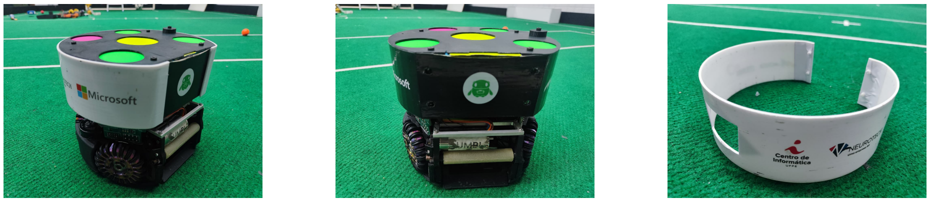


Fig. 2. The new dribbler mechanism prototype featuring the lateral damping system and PU roller.

that fatigued or cracked under stress. During the recent RoboCup competition in Salvador, we identified high wear on the printed mechanism. For RoboCup in South Korea, we are transitioning to a CNC-machined structure for all robots and are currently testing alternative damping materials to increase durability.

1.3 Cover

To comply with the updated RoboCup regulations regarding color-changing requirements, we developed a new modular cover design. The assembly consists of two distinct parts: a fixed base (black) and a removable section (white). The removable component features a circular geometry with a strategic opening and reduced thickness; these characteristics provide controlled elastic deformation that facilitates attachment to the fixed base. Furthermore, the removable part includes structural pins that engage with corresponding holes in the fixed base, ensuring that the components remain securely locked during operation. Improvements were also made to the internal structure of the mounting holes that secure the cover to the robot's chassis. This new design ensures compliance with the color-changing rules while reducing assembly time and simplifying maintenance procedures.



(a) Assembled cover (b) Cover black (c) Cover white

Fig. 3. Views of the cover assembly stages.

1.4 Kicker

In previous competition years, we faced issues related to consistency and standardization in the robots' kicker mechanisms. We observed differences in the dimensions of the plungers for the front and chip kickers across the robots. In v2025, our team maintained the overall kicker structure but, using electromagnetic simulations in femm [9] 4, redesigned the coils and plungers, introducing standardized and better optimized versions across all robots. As a result, both front and chip kicks reliably achieved higher output speeds, reaching a uniform speed of 5.3 m/s; however, due to time and manufacturing limitations, we kept all changes within the structural design of the previous kicker, therefore, limiting the possible efficiency gains shown by the simulations. The next steps on this front is to improve the efficiency by incorporating electromagnetic shielding and improving the collision mechanics of the mechanism.

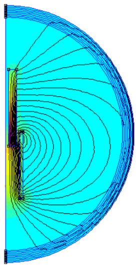
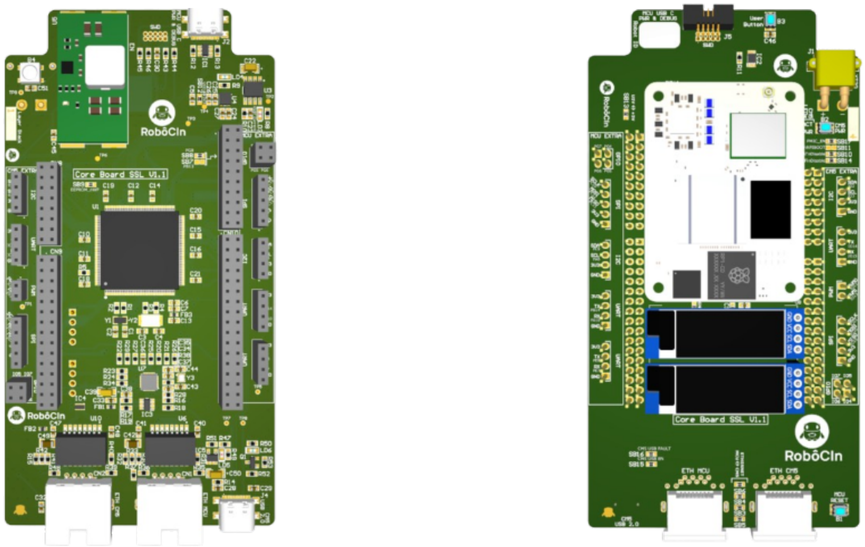


Fig. 4. simulation of the kicker mechanism in femm.

1.5 Core Board

Following the architectural transition initiated in the previous season, the 2026 development cycle focused on the electrical validation and robustness of the Core Board, a custom carrier board for System-on-Module (SoM) integration. While the v2025 iteration defined the high-level topology (coupling the MCU with the SoM), the v2026 revision addresses the engineering challenges of integrating these distinct processing units onto a single, compact Printed Circuit Board (PCB). Historically, the robotic platform's communication system relied on radio frequency (RF) transmission; however, due to signal instability and significant latency observed during competitive events, a migration to a 5 GHz Wi-Fi protocol became necessary. This requirement led to the development of the Core Board to supersede the previous architecture based on the NUCLEO-F767ZI development board. The proposed system integrates a high-performance Microcontroller Unit (MCU), the STM32F767ZIT6, with a Raspberry Pi Compute Module 5 (CM5) on a single 4-layer Printed Circuit Board (PCB).



(a) Front view of the core board

(b) Back view of the core board

Power Architecture and Thermal Management Integrating a Linux-based module (CM5) alongside a real-time MCU introduced significant power constraints, as the CM5's peak power consumption (approximately 5W) exceeded the capabilities of the legacy voltage regulators found on the Main Board, which among other functionalities has the responsibility to provide power to the whole robot. To address this, the Core Board implements a dedicated high-current power stage featuring a local buck converter (TPS51397A) to provide a stable 5.1V rail capable of sustaining the CM5's current spikes. Crucially, the design incorporates a Supervisory Power Gating mechanism managed by the STM32F7. Using a high-side load switch topology, the MCU can perform a hard power cycle of the Linux system in case of an Operating System (OS) hang without manual intervention.

Optimized Internal Communication The data link between the low-level MCU and the high-level CM5 relies on a 10Base-T physical layer transciever (PHY) architecture centered around the LAN8742A-CZ-TR transceiver. The v2026 design solves this through a Discrete Magnetics Implementation on the PCB to manage isolation. Although both processors support higher throughput, the interface is configured strictly for 10Base-T (10 Mbps) using a minimal Media Dependent Interface (MDI) with two differential pairs to maximize signal robustness against electromagnetic interference (EMI) from the BLDC motors and to simplify impedance matching on the 4-layer PCB. As a fail-safe measure, a redundant UART serial link was routed between the processors to ensure basic telemetry remains available during Ethernet initialization errors.

Hardware Configurability and Diagnostics The PCB adheres to a "Sandbox" design philosophy to mitigate risks in this first custom revision. Solder Bridges were placed on critical signal lines to allow for post-fabrication hardware reconfiguration. For instance, a dedicated solder bridge manages the CM5's bluetooth enable signal. When soldered, this bridge ties the enable pin to GND, effectively disabling the onboard bluetooth at the hardware level. This ensures the SoM does not generate unnecessary EMI or consume power for unused wireless peripherals during a match. Furthermore, the board exposes comprehensive diagnostic interfaces:

- **Dual OLED Feedback:**Independent displays connected to the MCU and the SoM provide immediate visual status, such as IP address and battery voltage.
- **Expansion Headers:**, Unpopulated headers break out SPI, I2C, and GPIO buses from both processors, ensuring the hardware remains adaptable for future sensor integration.

Future Work: Hardware Unification and Decentralized Intelligence The successful validation of the v2026 Core Board serves as proof of concept for a more ambitious architectural overhaul. The roadmap for future seasons focuses on two synergistic advancements: the transition to a monolithic electronic design and the migration of high-level strategic processing to the network edge.

Monolithic Mixed-Signal PCB Architecture The current physical separation between the Main Board (power actuation) and the Core Board (logic and processing) imposes mechanical constraints and introduces potential points of failure through interconnects. The ultimate hardware goal is to merge these domains into a single, high-density monolithic PCB. This unification will require a transition to a high-layer count stackup. Increasing the layer count is technically imperative to ensure Signal Integrity (SI) and Power Integrity (PI) in a mixed-signal environment.

- **Noise Isolation:** A multi-layer topology allows for dedicated internal ground and power planes. These planes act as electromagnetic shields, isolating the sensitive high-speed digital signals of the Compute Module 5 (such as PCIe or Ethernet traces) from the significant switching noise generated by the BLDC motor drivers.
- **Thermal Dissipation:** Additional copper layers will function as thermal spreaders, efficiently dissipating heat from the power stages without requiring bulky heatsinks, thereby maintaining the robot's low center of mass.
- **Routing Density:** A unified board permits the use of more compact, fine-pitch components and BGA packages, further reducing the overall electronic footprint and freeing up internal volume for mechanical optimizations.

Edge Computing and Full Autonomy Currently, the Raspberry Pi CM5 is primarily used as a high-bandwidth communication gateway, while its Quad-core Cortex-A76 processing capabilities remain largely underutilized. Our long-term vision is to gradually shift part of the decision-making and processing responsibilities from the external base station to the embedded system, increasing onboard autonomy.

The addition of a CM5 not only improves communication robustness, but also enables onboard execution of more complex tasks, such as path planning, state estimation, and strategy execution. This approach should allow us to shorten the perception-to-actuation loop and better exploit the computational capabilities of the embedded platform.

2 Embedded Software: The MARS Architecture

2.1 Architectural Overview

The migration to Division A and the subsequent adoption of the 5GHz Wi-Fi standard necessitated a paradigm shift in the robot's onboard software. To fully exploit the computational resources of the Raspberry Pi Compute Module 5 (CM5) beyond simple packet forwarding, the team developed a new framework titled MARS (Microservice Architecture for RobôCIn Systems).

Inspired by the legacy SSL-Core repository, MARS adopts a containerized microservices architecture to ensure modularity and scalability, as described in last year's TDP [3]. Instead of a monolithic application, critical robot functions—such as Wi-Fi communication management, Protobuf packet deserialization, and system logging—are encapsulated in independent Docker containers. This isolation prevents a failure in a non-critical module (e.g., logging) from crashing the entire system. Service orchestration is handled via docker-compose, triggered automatically by a systemd unit file upon the operating system's boot, ensuring a streamlined startup sequence.

2.2 Validation and Stress Testing

The initial deployment of MARS occurred during the 2024 RoboCup Brazil, where a secondary squad (Team B) was utilized for in-competition validation. While initial results were inconclusive due to late-stage responsiveness, subsequent laboratory testing revealed significant stability challenges inherent to the new stack.

Extensive longitudinal tests exposed a critical, non-deterministic failure mode: after approximately 20 to 30 minutes of continuous operation, the command transmission loop would abruptly cease. Notably, restarting individual Docker containers proved insufficient to restore Wi-Fi communication, requiring a full restart of the service stack to reestablish connectivity. Reproducing this behavior proved resource-intensive, requiring a full fleet of six robots and complete battery discharge cycles to capture a single failure event. The investigation considered multiple potential points of failure, including:

- Router instability and network congestion.
- Race conditions within the communication container.
- Thermal throttling of the CM5.
- Protobuf serialization errors.
- Power supply inconsistencies to the compute module.

2.3 Fault Tolerance and Recovery Mechanisms

During the RoboCup 2025 debugging process, it was observed that while the hardware remained operational, the software service could enter an unresponsive state. During controlled debugging sessions, a manual restart of the systemd service was sufficient to restore functionality, whereas restarting only the affected Docker containers was not enough to recover the Wi-Fi subsystem, indicating a software-level deadlock or resource exhaustion rather than a hardware latch-up.

To mitigate this issue during matches, the team implemented a Supervisor Watchdog. This mechanism monitors the command throughput from the Raspberry Pi to the microcontroller. Upon detecting a cessation of data transmission (timeout), the watchdog automatically triggers a forced restart of the entire docker-compose service stack. Although this solution acts as a fail-recovery mechanism rather than a root-cause fix — occasionally resulting in brief periods of inactivity during matches — it ensures that the robot can automatically recover from the freeze state without human intervention. This limitation represents a negative operational impact, as the recovery window may cause the robot to miss critical defensive or offensive opportunities, but it still maintains partial availability throughout the game.

3 Positioning and Pass Selection

3.1 Previous Pass Architecture

The current positioning architecture remains largely consistent with the approach detailed in the RobôCIn 2024 TDP [4], with only minor adjustments. It follows a common approach adopted by teams such as Tigers [6] and ZJUNlict [7], illustrated in Figure 5, and composed of four stages: pass generation, candidate filtering, evaluation, and selection.

The final selection stage is a point of divergence among different teams. At RoboCIn, the final score of a candidate position is computed as the product of all metric results [4]. However, this approach presents two limitations. First, a single low metric value disproportionately penalizes the total score, potentially discarding viable positions. Second, the reliance on a timer to trigger probability calculations introduces latency, compromising real-time responsiveness and causing missed passing opportunities.

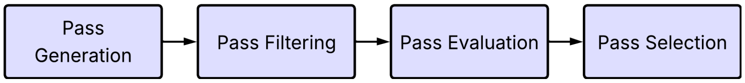


Fig. 5. Standard pipeline for pass selection.

3.2 Refined Pass Architecture

This year's updates refine the metric combination phase to prevent disproportionate penalties caused by individual low scores and replace the fixed calculation timer with a more responsive mechanism.

To improve positioning and pass selection, we model the problem as a multi-criteria decision task. As shown in Figure 6, the proposed pipeline preserves the main phases of the previous approach while introducing a classification stage between evaluation and selection. This stage categorizes candidate passes into qualitative classes (Optimal, Acceptable, and Sub-optimal) using the FTOPSIS-Class technique [2], allowing the system to prioritize promising options before final selection.

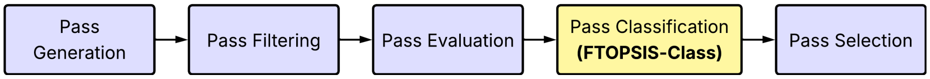


Fig. 6. Modified pipeline for pass selection.

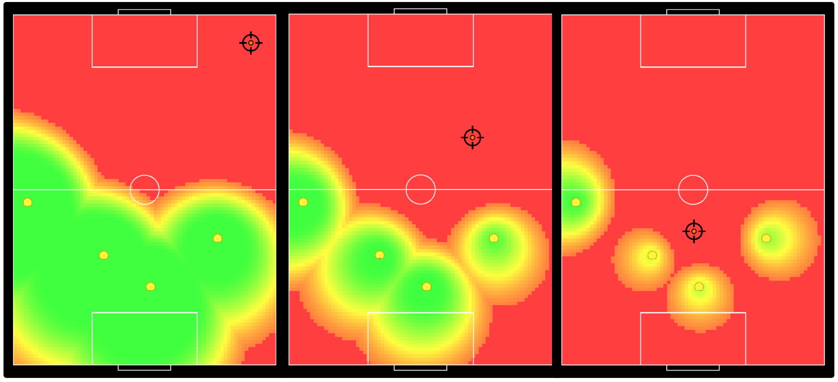
To centralize information, we adapted the pass concept proposed by Tigers [6] to include the attributes: source, target, kick speed, receiving speed, and pass duration.

Pass Generation and Filtering The generation and filtering stages remain consistent with the previous approach. Candidate points are generated on a 24×18 grid to ensure uniform field coverage. All pass attributes are computed, and strategic filters discard invalid candidates, such as points inside the goalkeeper's area or dangerously close to the defensive zone.

Pass Evaluation Each candidate pass is evaluated according to seven independent metrics capturing tactical or spatial aspects of pass quality. Unlike the previous multiplicative strategy, each metric is treated as an individual decision criterion and later processed by the classification module, preventing isolated low values from disproportionately penalizing promising passes.

Ally Time Margin (ATM). This metric evaluates whether an ally robot can reach the pass destination before the ball with sufficient temporal margin to properly receive it. Inspired by reachable point filters and pass reception models [6] [4], this metric favors passes toward locations where an ally arrives earlier than the ball, allowing stabilization and preparation for the next action, as shown in figure 7. Higher scores correspond to destinations reached earlier by allies, while unreachable or late arrivals are penalized.

Fig. 7. Distribution of *Ally Time Margin* metric scores. The scenarios, from left to right, are: high time margin, medium margin, and low margin. The allied goal is at the top and the opponent's goal is at the bottom. Yellow circles represent allied players.



Enemy Proximity (EP). Enemy Proximity assesses defensive pressure at the pass destination by estimating how quickly nearby opponents can reach the target location. Based on temporal interception concepts [8], it penalizes passes that terminate in areas where opponents arrive before or shortly after the ball, favoring safer receiving locations where the receiver can act with greater freedom and safety. Only the closest adversaries are considered to focus on immediate threats.

Goal Distance (GD). This metric rewards passes that position the ball closer to the opponent's goal, reflecting the increased likelihood of successful shots from shorter distances. Following similar principles adopted in prior works [8] [6], this metric assigns maximum scores to destinations within an ideal shooting range and progressively penalizes locations farther from the goal.

Goal Opening (GO). Goal Opening evaluates how much of the opponent's goal is visible from the pass destination, considering occlusions caused by defenders and goalposts. Inspired by goal visibility and shooting quality estimators [5] [6], it favors passes that place the receiver in positions with a wide, unobstructed shooting angle receive higher scores.

Ideal Range (IR). The Ideal Range metric evaluates pass length, reflecting the trade-off between excessively short passes and overly long passes. Following the concept of an optimal passing distance [5], this metric favors distances within a predefined optimal interval while penalizing excessively short or long passes.

Pass Interception (PI). The Pass Interception metric estimates interception risk by evaluating opponent-induced shadows along the pass path, accounting for enemy motion. Similar to approaches used in interception risk assessment [8] [6], it promotes passes with lower chance of interception.

Redirect Goal Kick (RGK). The Redirect Goal Kick metric favors passes that enable immediate one-touch shots toward the opponent's goal. Motivated by prior work on redirected shots and fast finishes [8] [7] [6], it evaluates the angular alignment between the incoming pass direction and the shooting direction. Passes that arrive with small angular deviation allow direct redirection and receive higher scores, while large misalignments are penalized, as they require ball control and reorientation. This metric promotes fast offensive actions that reduce defensive recovery time.

Fuzzy Classification with FTOPSIS-Class The core innovation of our approach is the application of FTOPSIS-Class [2] to categorize passes. This method uses fuzzy logic to assign each alternative to one of **p** predefined qualitative classes, handling the uncertainty inherent in dynamic environments, which is particularly suitable for pass quality assessment.

Our implementation utilizes Triangular Fuzzy Numbers to represent linguistic variables (e.g., "Very Good", "Neutral", "Poor"). The general flow of the classification phase is illustrated in the flowchart in Figure 8.

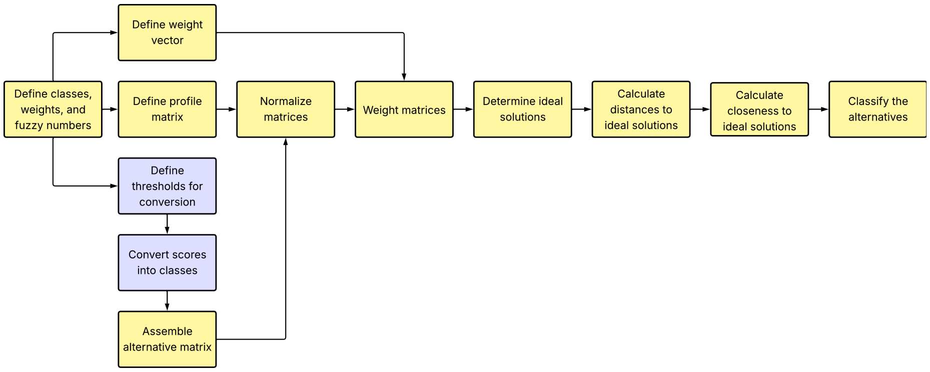

Define weight vector
Define classes, weights, and fuzzy numbers
Define profile matrix
Normalize matrices
Weight matrices
Determine ideal solutions
Calculate distances to ideal solutions
Calculate closeness to ideal solutions
Classify the alternatives
Define thresholds for conversion
Convert scores into classes
Assemble alternative matrix

Fig. 8. Flowchart of the classification phase. In yellow are the FTOPSIS-Class steps. In blue, extra steps necessary for the method to work.

Initially, we defined three output classes: *Optimal*, *Acceptable*, and *Sub-optimal*, each associated with a specific boundary profile (e.g., an *Optimal* pass requires "Very Good" scores in metrics like *Goal Distance*). Raw metric scores are converted into qualitative classes using predefined thresholds, manually defined based on prior domain knowledge and relevant decision intervals. These qualitative labels form the alternative matrix, where each candidate pass is represented by the classes associated with its performance across all criteria.

The FTOPSIS-Class procedure then normalizes and weights both the evaluation and class profile matrices according to the criteria weight vector. After that, it computes weighted Euclidean distances to the fuzzy ideal solutions, producing a closeness coefficient for each class. Each pass is assigned to the category (*Optimal*, *Acceptable*, or *Sub-optimal*) for which this coefficient is maximized. Figure 9 illustrates the resulting classification of points on the field for a given configuration of allied robots (yellow), opposing robots (blue), and the ball.

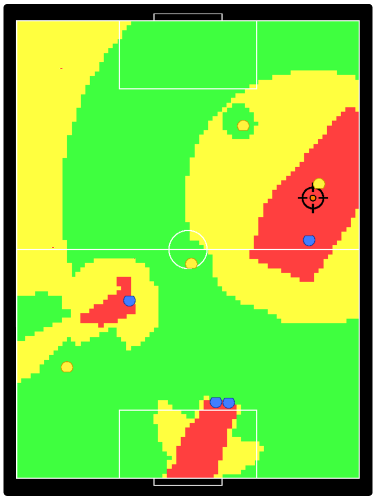


Fig. 9. Distribution of the classification on the field. Allies are in yellow, opponents are in blue, and the ball is in orange with a target around it. Green regions indicate Optimal passes; Yellow ones are Acceptable; and those in red correspond to Sub-Optimal.

Selection Strategy Finally, the **Selection** phase operates hierarchically based on the classification results. It prioritizes *Optimal* passes. If the required number of support points is not met, it cascades to *Acceptable*, and finally *Sub-optimal*. A minimum distance constraint ensures spatial distribution, and a hysteresis mechanism prevents erratic switching between candidates with similar scores.

3.3 Comparison between architectures

To validate the proposed approach, we conducted a comparative experiment between the baseline multiplicative model and the implementation of the FTOPSIS-Class. We performed 10 complete matches using the *Simulator-CLI* and the SSL Game Controller to ensure unbiased and automated gameplay. Both versions operated under identical conditions using the RobôCIn SSL-Unification Software.

We analyzed performance using three categories of metrics:

- **Match Outcomes:** Wins, draws, losses, and goals scored.
- **Offensive Metrics:** Total passes, pass completion rate, progressive passes [13], key passes (leading to shots) [12], and Field Tilt [11] [1].
- **Efficiency:** Goal conversion rate (goals/shots).

The overall results, summarized in Table 2, show a clear improvement in performance. The refined architecture achieved 6 wins, 3 draws, and only 1 defeat, while scoring four times more goals than the baseline. These results indicate that the fuzzy classification strategy more effectively identifies high-value passing opportunities than the previous multiplicative model.

Table 2. Overall Results

Metric	**Refined Architecture**	**Previous Architetcure**
Wins	**6**	1
Draws	3	3
Defeats	**1**	6
Goals Scored	**12**	3

Table 3 provides a more detailed view of offensive behavior. The refined version shows higher passing volume and accuracy, along with a higher Field Tilt, indicating a more sustained possession in the opponent's half. Although the number of shots decreased, goal conversion increased substantially, rising from 2.16% to 10.62%. This behavior aligns with the observed increase in the proportion of key passes, reflecting more selective and effective chance creation, where passes are prioritized for their potential to create high-quality scoring opportunities.

Table 3. Per-match averages and percentages of metrics for the previous and refined versions

Metric	**Previous Architecture**	**Refined Architecture**
Total passes	31,3	**36,9**
Completed passes	15,8 (50,48%)	**20,7 (56,10%)**
Progressive passes	**9,0 (28,75%)**	9,7 (26,29%)
Key passes	4,2 (13,42%)	**5,8 (15,72%)**
Field tilt	46,22%	**53,78%**
Shots	**13,9**	11,3
Goals scored	0,3 (2,16%)	**1,2 (10,62%)**

Values represent **per-match averages**.
Goal percentage is calculated relative to the average number of shots.
Pass-related percentages are calculated relative to the average number of passes.

Discussion The integration of FTOPSIS-Class significantly improves the pass selection pipeline by replacing the previous multiplicative model with a fuzzy classification mechanism that mitigates penalties from isolated low metrics. When combined with a hierarchical selection strategy, this results in more stable decision-making and improved offensive efficiency. The validation confirms superior offensive performance, including increased possession and goal-scoring efficiency, validating the practiacl impact of the method on competitive performance.

4 Conclusion

This paper presented the technical advancements of the RobôCIn team for RoboCup 2026, focusing on hardware unification, software resilience, and strategic decision-making. In electronics, we detailed the Core Board engineering, emphasizing the transition to a high-layer stackup for signal integrity and the roadmap toward fully autonomous edge computing using the CM5. In software, the MARS architecture was refined with fault-tolerance mechanisms, specifically a supervisor watchdog, to mitigate non-deterministic failures in the containerized environment. Finally, the strategic layer introduced a fuzzy multi-criteria approach (FTOPSIS-Class) for pass selection, which significantly enhanced offensive efficiency and goal-scoring capabilities. These improvements collectively aim to increase the robot's autonomy, robustness, and overall competitive performance.

5 Acknowledgement

First, we would like to thank our advisors and the Centro de Informática (CIn)-UFPE for all the support and knowledge during these years of project and development. We also would like to thank all our sponsors: Moura Batteries, ITEMM, Microsoft, CESAR, Incognia, HSBS, SENAI, Neurotech and STMicroelectronics.

References

1. Analyst, T.F.: Field tilt – football statistics explained. `https://the-footballanalyst.com/field-tilt-football-statistics-explained` (2025)
2. Ferreira, L., Borenstein, D., Righi, M.B., de Almeida Filho, A.T.: A fuzzy hybrid integrated framework for portfolio optimization in private banking. Expert Systems with Applications **92**, 350–362 (2018). https://doi.org/https://doi.org/10.1016/j.eswa.2017.09.055, `https://www.sciencedirect.com/science/article/pii/S0957417417306619`
3. Franca, A., Barros, B., Barbosa, D.C., Xavier, D., Ara´ujo, E., Pereira, F., Melo, J.G., Silva, J.R., Cavalcanti, L.H., Asfora, M., Albuquerque, M., Alves, M., Paix˜ao, M., Vasconcelos, M., Silveira, Onias C. B. Dutra, V., Araújo, V., Barros, E.: Robocin small size league extended team description paper for

robocup 2025 (2025), https://ssl.robocup.org/wp-content/uploads/2025/04/2025_ETDP_RoboCIn.pdf, roboCup Small Size League, Eindhoven, Nederland, 2024

4. Franca, A., Barros, B., Gomes, C., Silva, C., Alves, C., Barbosa, D.C., Xavier, D., Ara´ujo, E., Pereira, F., Batista, H.A., Cavalcanti, L.H., Asfora, M., Alves, M., Paix˜ao, M., Vasconcelos, M., Vin´icius, M., Melo, J.G., Silva, J.R., Cruz, J.V., Leite, J., Santana, P.H., Oliveira, P.P., Rodrigues, R., Morais, R., Teobaldo, T., Dutra, V., Ara´ujo, V., Barros, E.: Robocin small size league extended team description paper for robocup 2024 (2024), https://ssl.robocup.org/wp-content/uploads/2024/04/2024_ETDP_RoboCIn.pdf, roboCup Small Size League, Eindhoven, Nederland, 2024
5. Geiger, M., Carstensen, C., Ryll, A., Ommer, N., Engelhardt, D., Bayer, F.: Tigers mannheim (team interacting and game evolving robots) extended team description for robocup 2017 (2017), robocup Small Size League, Mannheim, Germany, 2017
6. Geiger, M., Ommer, N., Ryll, A., Ratzel, M.: Tigers mannheim (team interacting and game evolving robots) extended team description for robocup 2025. https://ssl.robocup.org/wp-content/uploads/2025/04/2025_ETDP_TIGERs-Mannheim.pdf (2025)
7. Huang, Z., Han, C., Shen, N., Yang, J., Yu, J., Zhao, A., Chen, Z., Du, H., Wen, L., Wang, Y., Guo, D., Xiong, R.: Zjunlict extended team description paper small size league of robocup 2023. https://ssl.robocup.org/wp-content/uploads/2023/02/2023_ETDP_ZJUNlict.pdf (2023)
8. Huang, Z., Zhang, H., Guo, D., Jia, S., Fang, X., Chen, Z., Wang, Y., Hu, P., Wen, L., Chen, L., Li, Z., Xiong, R.: Zjunlict extended team description paper for robocup 2020. https://ssl.robocup.org/wp-content/uploads/2020/03/2020_ETDP_ZJUNlict.pdf (2020)
9. Meeker, D.C.: Finite element method magnetics (femm). https://www.femm.info (2023)
10. Ryll, A., Jut, S.: Tigers mannheim (team interacting and game evolving robots) extended team description for robocup 2020 (2020), roboCup Small Size League, Mannheim, Germany, 2020
11. Willis, S.: Should we care about field tilt? https://www.cannonstats.com/p/should-we-care-about-field-tilt (2025)
12. Wyscout: Key pass. https://dataglossary.wyscout.com/key_pass/ (2025)
13. Wyscout: Progressive pass. https://dataglossary.wyscout.com/progressive_pass/ (2025)