Robot Club Toulon Team Description 2023

M. Dhainaut, M. Walter, C. Bohler, Y. Masinski, T. Mottay, T. Carceles, K. Jacq, M. Thomas, A. Mourgue, Y. Carvalho, B. Hanna, Z. El Bannoudi, A. Deverin, H.A. Pham, T. Soriano, V. Barchasz, and V. Gies

Université de Toulon,
Avenue de l'Université, 83130 La Garde, France `rct@univ-tln.fr`
Home page : `http://rct.univ-tln.fr`

Abstract. Robot Club Toulon Middle-size league (MSL) team aims at participating in the RoboCup 2023. This paper explains the architecture of our robots and the main evolutions and developments done this year.

Keywords: RoboCup Soccer, Middle-Size League, Multi-robot, Artificial Intelligence, Image Processing, Lidar

1 Introduction

Robot Club Toulon is representing University of Toulon, France, in the RoboCup Middle Size League (MSL). The team will be participating in the Middle-Size League for the second time in real conditions.

Our team has been participating to the RoboCup for the first time in Sydney 2019. We have also participated in 2021 (online) and in 2022 (without being able to come to Bangkok). Our most important results (link to RCT results) in the competition are :

- first place in the Scientific Challenge 2022
- second place in the Technical Challenge 2022
- second place in the Technical Challenge 2021
- third place in the Scientific Challenge 2021
- 4 national titles in the French Robot Cups.

At the moment of writing this paper, RCT team consists of 1 PhD's, 2 Post-docs, 12 MSc, 7 BSc, 3 staff members including an engineer and 2 researchers in electronics and robotics. For this edition, our robots have been partially redesigned in order to solve mainly mechanical issues

Structure of this paper is the following one :

- Electronics and Software architecture.
- Contributions of RCT in 2023 :
  - Mechanics: introducing a square base using brushless motors.
  - Perception : improved detection of balls, goals and opponents using embedded smart cameras.
  - Strategy : introducing a reinforcement learning model for playing soccer

2 Electronics and Software architecture

To go deeper in the details, all our mechanics and electronics are fully described in the *Mechanical and Electronic Presentations*.
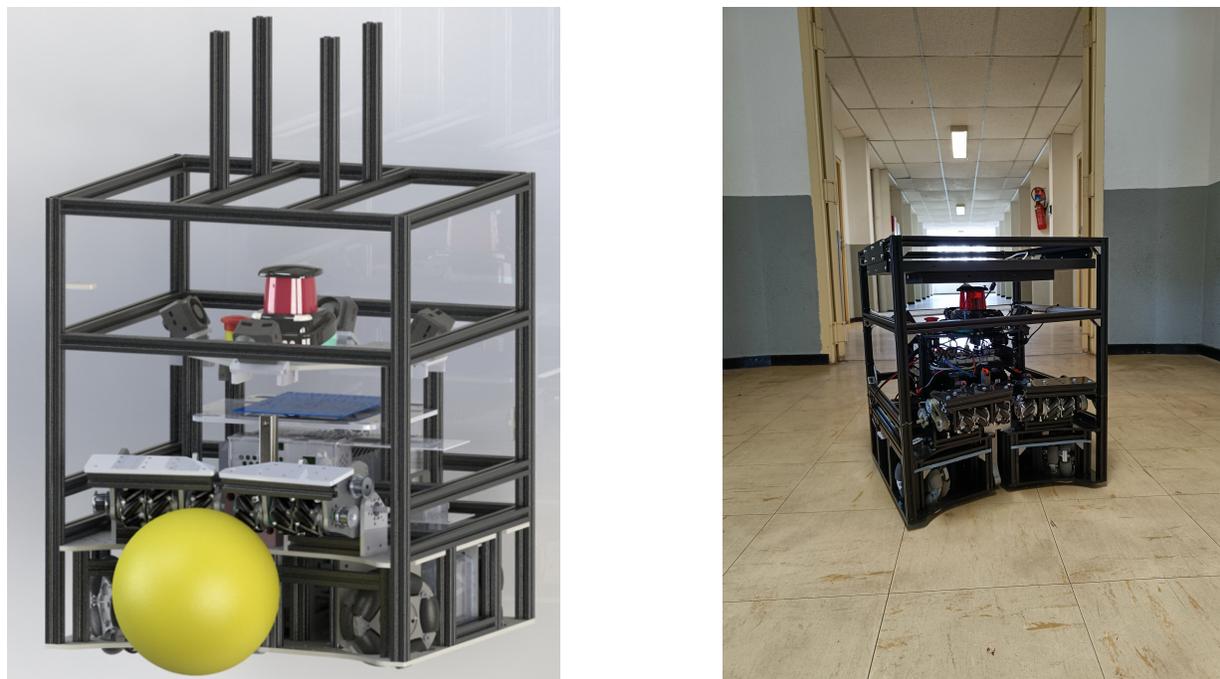
Fig. 1. Computer image of the 2022 robot of Robot Club Toulon Team, and picture of the robot.

2.1   Electronics

Electronics architecture of our robots is bio-inspired. The whole system is piloted by an *Embedded Computer* (*P12B-i7-10810U*) acting like a cortex and doing *intelligent* tasks using advanced sensors having a high bandwidth of data : LIDAR scene analysis and artificial intelligence for strategy are embedded in the computer. Computer vision is not embedded in this computer, as fours intelligent cameras (JeVois Pro) are used. Each one embeds artificial intelligence tasks such as image segmentation using deep learning in real time.

This cortex is connected to a *Sensor and Actuator Controller* acting as an autonomous nervous system, and doing repetitive and high frequency tasks such as sensor and motor management as shown in Fig. 2. This second board embeds a *Microchip DSP* having hardware peripherals for multi-threading tasks at a low level. High frequency motor control and sensors management is performed with a DSP using high speed interfaces such as USB, SPI or UARTs for synchronizing up to 20 different peripherals.

The *kicking system controller* is a third board, independent for development and safety reasons due to high voltage.

2.2   Software architecture

RCT robot software architecture is described at Fig. 3. Our robots no longer use a base station, each one acting as an autonomous player, even for decision tasks.

Code is divided in 2 parts corresponding to the 2 main electronics parts :

– Code is written in *C* for the motor and sensor control board based on a *Microchip DSP* 16-bits controller.
– For the cortex part embedded on the computer, code is fully written in *C#*. It is a fully event driven code with more than 60 independent modules linked together like a Matlab Simulink model. This way of coding allows students to work on small parts of the robot without having to know most of the code. It also
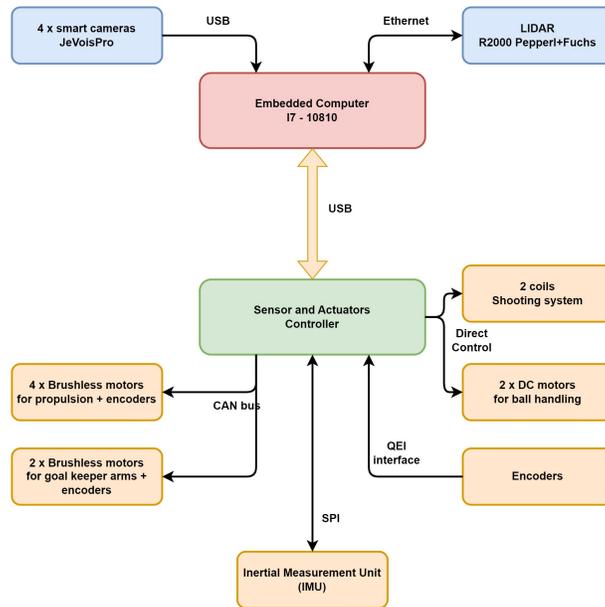
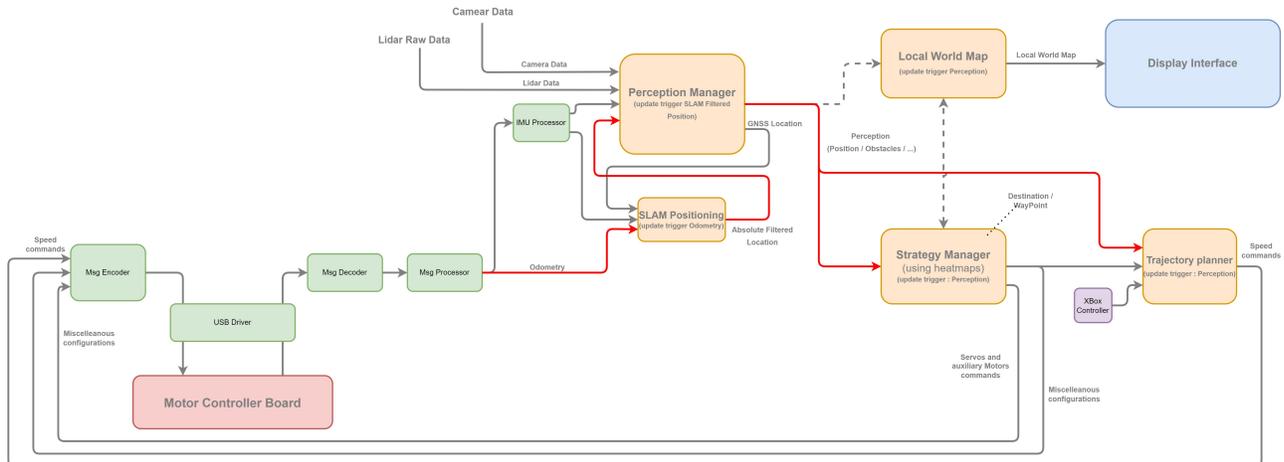**Fig. 2.** RCT robots electronic bio-inspired architecture

Fig. 3. RCT robots software architecture

allows to use the modules in multiple configurations such as the robot itself, or a simulator for the whole team and another opponent team (Fig. **??**). This method allows to increase the reliability of each module.

3   Contributions of RCT in 2023

3.1   Mechanics: introducing a square base using brushless motors

This year our robot has been completely redesigned to become a square robot. Very different from other teams robot shapes, this square one is not motivated by aesthetic aspects but adds new game playing features to our robot : the ability to socre doing volley or heads. Using square robots allows to bounce the ball on our robots with a relatively good control due to large flat areas, in order to deviate the ball and score like a

soccer player could have done with his head.

This innovation is based on the reality of football which isn't only played by foot but also with other parts of the body to control the ball for example. Using this ability requires to shoot on our robots, and implies to have a reinforced mechanical structure. For this we decided to use aluminium profiles which have many advantages : they are light, robust and the most important is that they are completely modular. This permits us to easily implement any features we'd like to add to our robot and also eased the building of the team by reducing the development time for a full team to less than 3 months.

Fig. 4. 4 wheels RCT platform with new motors

Another mechanical evolution of our robots in 2023 is to reduce their weight, mainly for reducing the cost of transportation. Motorisation using DC Maxon motors has been transformed to a brushless one using Robotmaster M3508 brushless motors. Those motors made us gain 3kg on the robot, reducing also the height of the propulsion base by 3cm. Removing chains ans sprockets, this direct drive brushless propulsion also drastically improves precision : wheels are directly attached to the engine axle, removing mechanical slacks and improving reactivity.

3.2   Improved embedded perception algorithms based on smart cameras and lidars

Perception is one the most important keys for being able to play autonomously by taking appropriate decisions in a given context. The most important the detection distance, the best scene comprehension. Being able to see far away from the robot requires appropriate sensors able to detect small objects (balls or robots for example) as far as possible. Omnidirectional vision used by most teams is very interesting as one sensor allows to see everywhere around the robot, but reduces drastically the size of each object on the image, limiting the detection abilities of the system.

Considering that, we have chosen another solution having a better resolution : using four cameras positioned on each face of our robots, each having a field of view of 120°. A neural processor from Hailo is embedded in each camera, providing us with high calculation power (26 TOPS/camera), mandatory to perform accurate real time object detection. It replaces older Google Coral Tensor Processing Units (TPU) having a computing power of 4 TOPS/camera.

Several algorithms have been tested throughout the year using mainly YOLOV3 tony and YOLOV5M. Results are presented in Table 1, using our own measurement and data for research papers [1].
Considering these results, *Yolov5M* has been chosen as good trade-off between speed and accuracy :
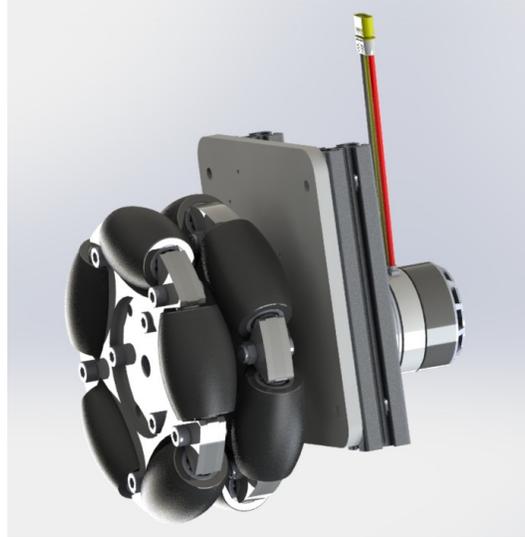
– 40 FPS is enough for playing soccer.

Fig. 5. RCT propulsion element

-	YOLOV3 tiny	YOLOV5M
Accuracy	0.35	0.63
FPS (Google Coral TPU)	30	5
FPS (Hailo 8 SPU)	200	40
Ball detection distance	5m	9m

Table 1. Comparison between YOLOV3 tiny and YOLOV5M

- Accuracy is good and there are very few false positive.
- Detection bounding boxes are well adjusted on objects.

Learning has been done using Nubot MSL RoboCup image repository, adding several additional images taken during RoboCup 2019 and our own field. Different classes have been labelled and learnt : robots, balls, human and goals. Results are shown on Fig. 6.

The detections are outputted through serial, and the four feeds are joined together in a custom chip and is converted to USB. However, due to the high data rate, we are moving from serial to native USB to avoid overflow and data loss. This is especially true if we output more than the minimum information like the accuracy, the class name, the camera name...

Detection is an important task, but localisation is also very important. Positioning precisely objects using a single camera is rather difficult due to the projection of objects in the focal plane of the cameras. Once detection have been done, a lidar is used for finding the accurate position of surrounding identified objects (except for balls).

3.3 Strategy

Improving team strategy is one of the main goals of this year for our team. Our goal is to let the robots decide what could be their best action according to team rules (such as one player maximum contesting a ball) and to the other potential robot actions.

An algorithm has been developed, based on the idea that we will play with humans in a near future, and consequently humans can not send their real position, perception and chosen actions to their robot teammates. This means, each robot has to imagine what could be their best action and their teammates
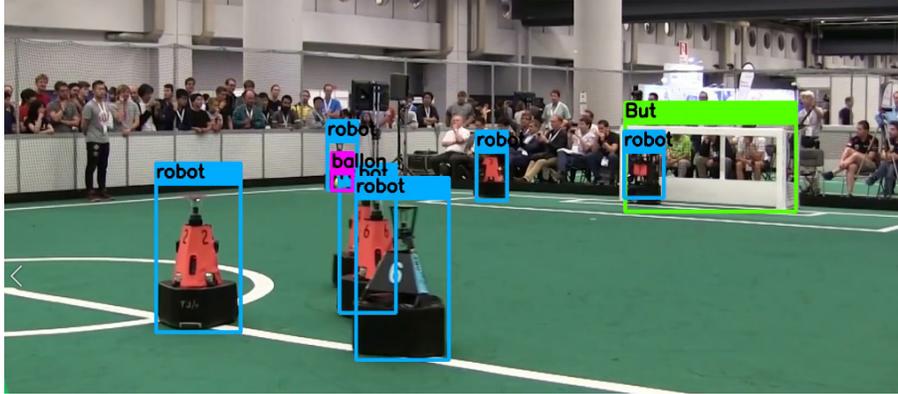
Fig. 6. Object detection result

best actions in order to make a choice.

In order to increase the flexibility of each robot and to improve the interoperability between the robots in different scenarios, several possible actions for each robot has been defined (Table 2).

Action		
A1	`Stopped`	
A2	`GoalKeeping`	
A3	`RushToGoal`	
A4	`TryToCatchBall`	
A5	`TryToPassBall R1-4 (A5-A8)`	Try to pass on to teammates $i, (i = 1...4)$
A6	`TryToShoot P1-4 (A9-A12)`	Try to kick the ball into one of the four-goal positions
A7	`CloseAssist`	Move closer to support teammates who have the ball
A8	`UnMarking`	Unmark
A9	`BlockShooting R1-4 (A15-A18)`	Move to be able to cut the ball of the opponent
A10	`BlockPass R1-4 (A19-A22)`	Move to block the direction of the opponent's movement

Table 2. List of possible actions (to be expanded or reduced)

Action selection algorithm is described at Figure 7. Every 20ms, the following steps are done :

- Step 1: game situation is determined, based on the game state from the referee box and the team ball handling status computed by aggregating ball handling information from each player.
- Step 2: according to the game situation, each robot computes a list of possible actions for itself and for other teammates (approximately 300 in real conditions).
- Step 3: for each of these actions, the expected reward (Q-Value) is computed, depending on the probability of success of the action, which is equal to the product of the probabilities of success of the action considering specific criteria (such as the probability of interception or the score of progression toward the opponent goal). It is important to note that this expectation of reward can be zero in some cases (for example, making a pass if one does not have the ball).
- Step 4: when all the Q-Values for all the possible actions of all the teammates have been computed, the decision of the best action can be taken. This one depends first on team rules, that can impose that a specific action has to be done by a given number of robots at a time. for each team rule, one or more teammates are assigned an action. When team constraints have been all used, if the considered robot has not been attributed an action yet, it chooses the action having the best score for him. This is the end of the decision algorithm.

– Step 5: Chosen action is transmitted to the trajectory planner and the game manager.

This algorithm has been implemented successfully on the team simulator as well as on RCT robots. Its drawback is that there are many hyper-parameters is the Q-Value computation for each possible action. These parameters will be optimised using reinforcement learning techniques in near future.
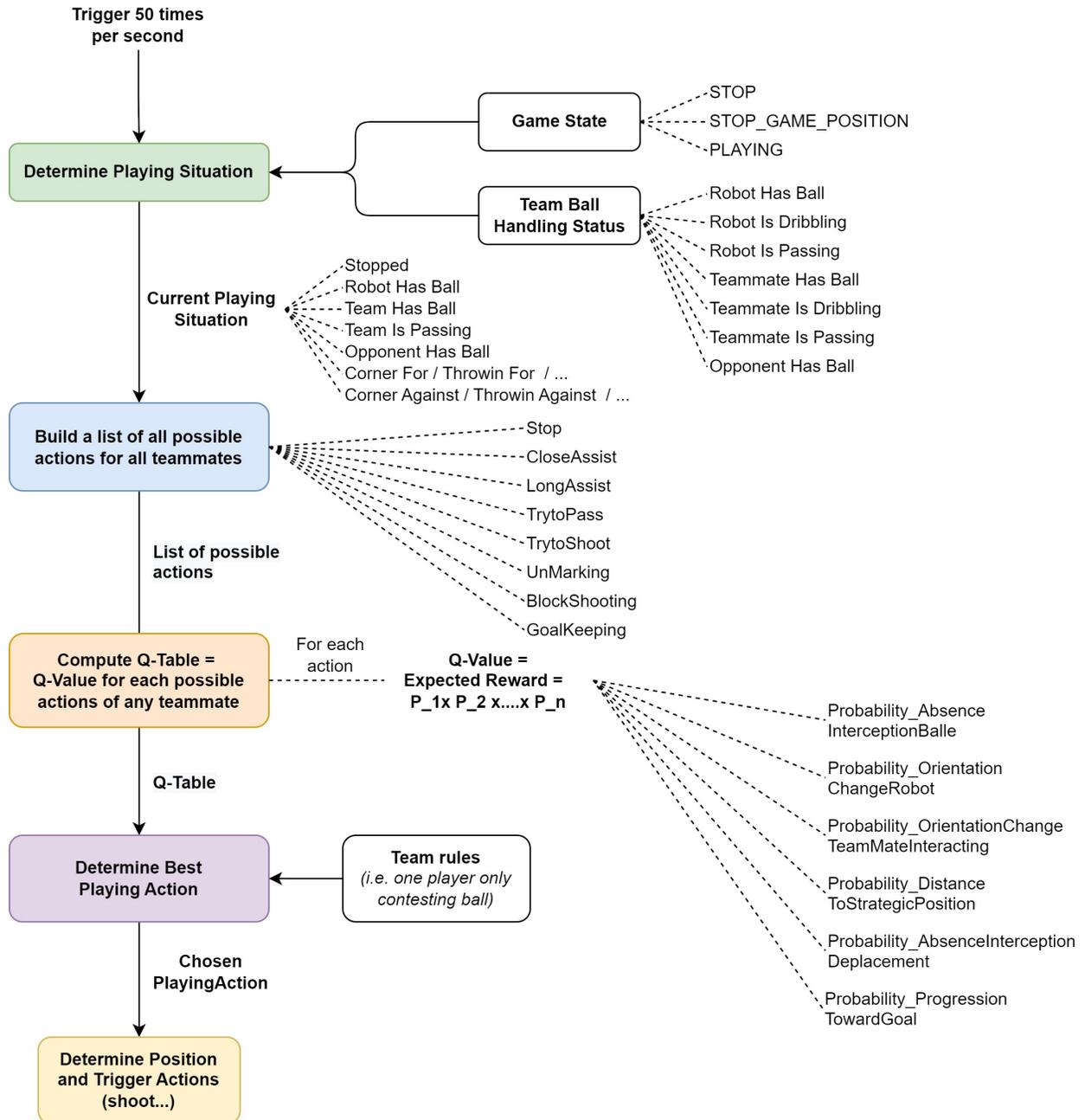

Fig. 7. Strategy algorithm of RCT team

3.4   Simulation example of attack and defense strategy

Figure 8 show an example of coordinated strategies for a group of robots, which is implemented in our simulation software. In which, robot number 10 is performing **GoalKeeping** action, robot number 11 is performing **TrytoCatchBall** action, and robots 12, 13, 14 are in **PositioningStrategyFixed** state.
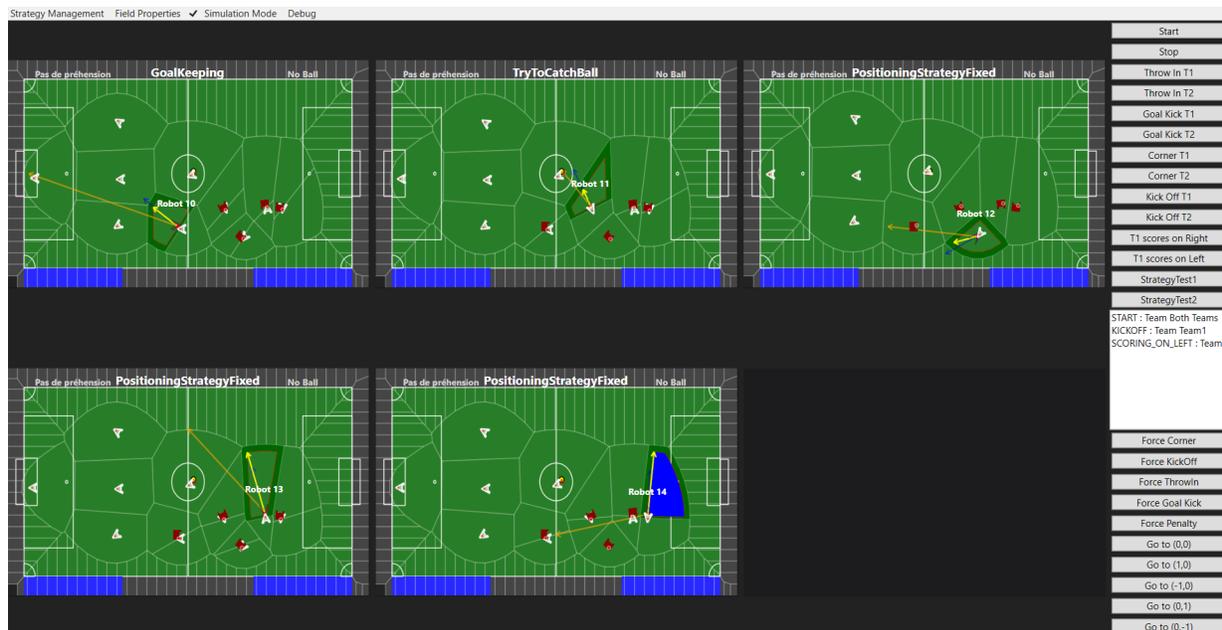

Fig. 8. Example of coordinated strategies for a group of robots

4   Conclusion

Participating in the RoboCup is a challenge for our young team. A strong emphasis has been put on the game this year. Everything is ready now for playing as a team having a basic strategy.

Thanks to the help of other teams, it has been a great adventure since 2019, and we are proud to have now a robot almost functional and to have developed some novel ideas for the MSL community.

Our challenge this year is to be able to play games against opponents, but it is still a real challenge for us !

References

1. Nepal, U., Eslamiat, H.: Comparing yolov3, yolov4 and yolov5 for autonomous landing spot detection in faulty uavs. Sensors **22**(2) (2022). https://doi.org/10.3390/s22020464, `https://www.mdpi.com/1424-8220/22/2/464`