Design of the dynamic behavior of soccer robots based on the Dec-POMDP framework

Thierry Soriano, Valentin Gies, Hoang Anh Pham

Design of the dynamic behavior of soccer robots based on the Dec-POMDP framework

Thierry SORIANO
University of Toulon
Toulon, France
thierry.soriano@univ-tln.fr

Valentin GIES
University of Toulon
Toulon, France
valentin.gies@univ-tln.fr

Hoang Anh PHAM
University of Toulon
Toulon, France
hoang-anh.pham@univ-tln.fr

***Abstract*—A soccer robot is a complex team sport that requires coordinated decision-making and execution from multiple robots. The Dec-POMDP approach provides a natural framework for modeling and optimizing the decision-making processes of a robot soccer team, as it allows for the representation of uncertainty and coordination among multiple robots. In this paper, we describe a model for a robot soccer team and study a specification of team behavior based on Dec-POMDP specialization. Our results show that the approach can effectively model and optimize the decision-making processes of a soccer team, leading to improved team performance.**

***Index Terms*—Dec-POMDP, coordinated decision-making, modeling soccer robots, decision-making.**

I. Introduction

The modeling of robot behaviors in the domain of soccer necessitates a formalism that is capable of accommodating various properties. This formalism must encompass the ability to describe subsystems that pertain to the exchange and coordination of agents, while also considering the impact of events and continuous state space. Furthermore, formalism must allow for the expression of agents' actions, the handling of input observations, and the incorporation of uncertainty and probability variables. Ultimately, formalism must facilitate the development of an optimal action strategy.

Moore's machines [1] lack the semantics necessary to describe collective actions. Mealy's machines [2] provide the possibility of associating actions to transitions that are executed during the shooting of the ball, but they do not allow for a group coordination. Petri nets [3], whatever their variation, offer possibilities of synchronization by semaphore but are not well-developed concerning the actions. The Grafcet or Sequential Function Chart [4] offers possibilities of describing actions, and of synchronizing different models of agents by semaphore, and also by triggered tasks or forcing, but these mechanisms remain rather limited for the coordination of agents. The formalism of the Statecharts [5] is also limited. Hybrid automata [6] offer a good semantics of continuous and discrete states, but the coordination of agents through the composition of automata is insufficiently operational. Markov models [7] introduce noise and uncertainty in the crossing of transitions but almost nothing for collective action.

Decentralized Partially Observable Markov Decision Process (Dec-POMDP) [8] is a mathematical framework used to model and optimize decision-making problems in multi-agent systems. In Dec-POMDP, the optimization of agent behavior involves addressing uncertainty resulting from the presence of other agents and the environment.

In [9], a proposed approach aims to resolve decentralized decision-making through an interaction-oriented approach. The method involves using distributed value functions (DVF) to break down the multi-agent problem into individual agent problems. Additionally, the DVF methodology has been extended to account for full local observability, limited information sharing, and communication disruptions. In [10], a decentralized partially observable Markov Decision Process (Dec-POMDP) is used to model multi-robot soccer and solve it using evolutionary algorithms. This algorithm uses finite state controllers to represent policies and searches the policy space with genetic algorithms. In [11], a decentralized partially observable Markov decision-processes (Dec-POMDPs) are general models for decentralized multi-agent decision-making under uncertainty. They address the case where each agent has macro-actions: temporally extended actions that may require different amounts of time to execute. They model macro-actions as options in a Dec-POMDP, focusing on actions that depend only on information directly available to the agent during execution. They extend three leading Dec-POMDP algorithms for policy generation to the macro-action case and demonstrate their effectiveness in both standard benchmarks and a multi-robot coordination problem. In [12], aims at the application of decision-theoretic (DT) frameworks to real-world scenarios in cooperative robotics. Current work, focusing on efficient communication policies for multiagent POMDPs is discussed. In [13], a decentralized multiagent Partially Observable Markov Decision Process (POMDPs) while maintaining cooperation between robots by using POMDP policy auctions is studied. Furthermore, they address the issue of mismatch with real inter-robot communication by applying a decentralized data fusion method in order to efficiently maintain a joint belief state among the robots.

Soccer team modeling and competition are intricate tasks involving multiple agents with varying objectives and actions, along with unpredictable and dynamic environments. Conventional methods like Markov Decision Processes (MDPs) presume that the agents possess complete environmental observability, which is not always true in soccer games. In this study, a soccer team-specific model is proposed using the Dec-

POMDP framework, where each player is deemed an agent with their individual observations, actions, and objectives. The aim of the team is to score goals. Additionally, a competition framework is proposed to appraise the performance of soccer teams modeled through the Dec-POMDP approach. The competition comprises simulating games between diverse teams and evaluating their performance based on several metrics, such as the number of goals scored, completed passes, and shots on target.

The paper is organized as follows: Section 2 begins by describing the decentralized partially observable Markov decision Processes. Section 3 discusses the specification of team behavior based on Dec-POMDP specialization. Section 4 presents an analysis of an example of decision-making towards the soccer team. Finally, section 5 provides conclusions and future work.

II. Decentralized Partially-Observable Markov Decision Processes

Decentralized partially observable Markov decision processes (Dec-POMDPs) are a generalization of both POMDPs and MDPs [8], [14], [15], designed to handle multi-agent environments. As illustrated in Figure 1, a Dec-POMDP represents a team of agents that must collaborate to accomplish a task by taking individual actions based on their local observations across a series of time steps. The agents share a common reward function that defines their collective objective, but it is typically unknown during execution. The execution is decentralized because each agent must choose its own action at each time step without being aware of the actions or observations of other agents. Moreover, the problem is partially observable because although the framework assumes the presence of a Markovian state at each time step, the agents do not have access to it.

A Dec-POMDP is defined by a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \Omega, O \rangle$, where $\mathcal{S}$ is a finite set of states, $\mathcal{A}$ is a finite set of actions for each agent, $\mathcal{T}$ is a state transition probability function. $\mathcal{R}$ is a reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, that maps states and joint actions to real numbers and is used to specify the goal of the agents. n-agent Dec-POMDP is said to be reward independent if there is a monotonically non-decreasing function f such that

$$\mathcal{R}(s, a) = f(\mathcal{R}_1(s_1, a_1), ..., \mathcal{R}_n(s_n, a_n)) \tag{1}$$

s_n is the state of agent n, a_n is the action of agent n. Ω is a finite set of observations for each agent. $\mathcal{O}$ is an observation probability function.

The aim of optimizing Dec-POMDP problems is to identify the best action selection policies and determine how state information should be presented and updated. To achieve this, an optimality criterion is used to specify precisely what should be optimized. The ultimate goal is to generate a favorable sequence of joint actions that results in a high long-term reward, known as the return.

III. Specification of team behavior based on Dec-POMDP specialization

The use of a Dec-POMDP approach in modeling a robot football team can provide various advantages, such as:

- Coordination: robots need to work together to achieve a common goal. Dec-POMDP can model the interactions between players, enabling them to coordinate their actions toward a common objective.
- Uncertainty: The outcome of a football robot game is influenced by various factors, such as the opposition's strategies, and state uncertainty. Dec-POMDP can model this uncertainty, enabling the team to make decisions based on the probability of different outcomes.
- Partial observability: robots have limited information about the game's state, including the ball's location, the position of their opponents, and the current score. Dec-POMDP can model this partial observability, enabling players to make decisions based on their and their teammates' observations.
- Flexibility: The football robot game is a dynamic and complex game that requires robots to adapt to changing situations on the field. Dec-POMDP can model this flexibility, enabling the team to change their strategy in response to changes in the game environment.

Assumption 1: **The group of football robots operates in a discrete time and the observed state of the system is discrete.**
The assumption 1 is essential for creating and analyzing the proposed model, and for designing control strategies that work well with the discrete nature of the system. Such strategies may be quite different from those used in continuous-time systems and require specific mathematical techniques.

In this study, we have not chosen a policy associated with Bellman's equation [16] because it seemed more complex to implement in real-time, and we have proposed the expected reward $\mathcal{R}_{i,j}$ for each robot j corresponding to each action i.

$$\mathcal{R}_{i,j} = p_{1,i,j} \times p_{2,i,j} \times p_{3,i,j} \times p_{4,i,j} \times p_{5,i,j} \times p_{6,i,j} \tag{2}$$

where $p_{1,i,j}$ is the probability absence interception ball (we, therefore, assume that the robots are oriented correctly so that the pass can be made), $p_{2,i,j}$ is the probability of success of the change of orientation to be made by the robot so that it orients itself in the direction of the pass to be received, $p_{3,i,j}$ is the probability of success of the change of orientation to be made by the receiver (ball carrier) so that he is oriented in the direction of the chosen reception position, $p_{4,i,j}$ is the probability linked to the distance between the position of the robot considered and that of the chosen reception position, $p_{5,i,j}$ is the probability absence interception displacement, $p_{6,i,j}$ is the probability progression toward a goal. It should be noted that the probabilities mentioned here are calculated based on the assumption that if the robot is in the best condition (e.g, the distance from the robot to the goal is the shortest) then there will be a high probability of success.

Assumption 2: **In our present study, we assume that the robots have a common set of observable information.**
Through assumption 2, The robots share their individual observations via communication and therefore can maintain

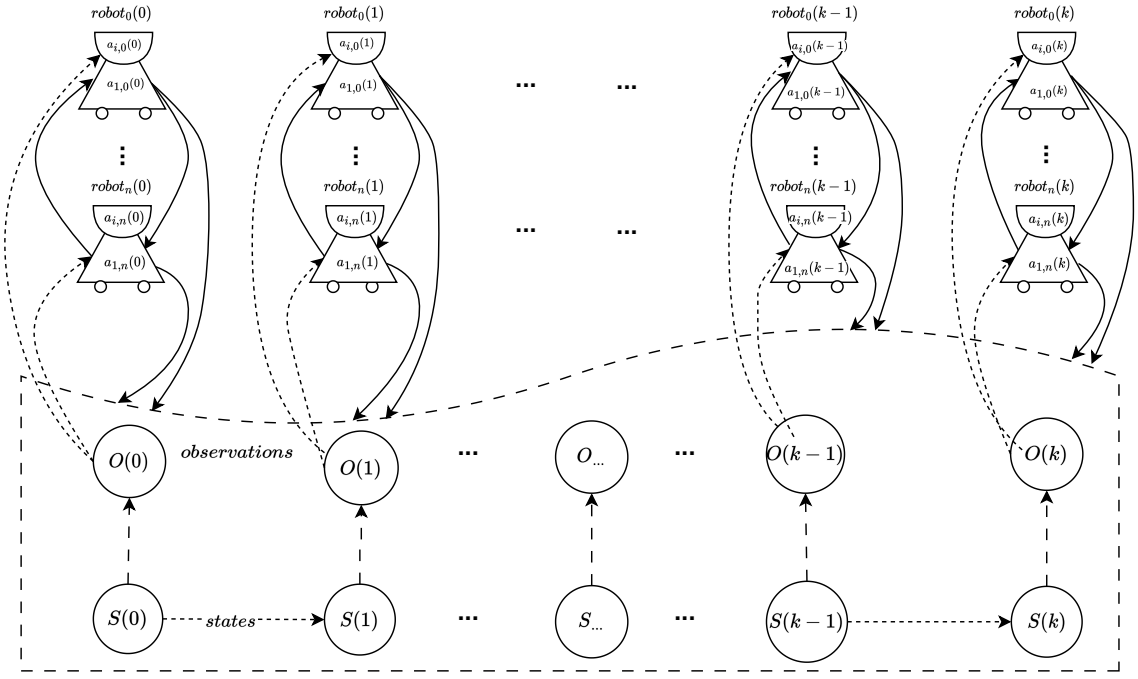

Fig. 1. A more detailed illustration of the dynamics of a Dec-POMDP (*where* $robot_n(k)$ *is the robot* n *at time* k, $a_{i,n}(k)$ *is the action* n *of the robot* i *at time* k), $O(k)$ is the set of observation at time k, $S(k)$ is the set of states at time k

Action		
$a_{1,i}$	`Stopped`	Stop all robot activities
$a_{2,i}$	`GoalKeeping`	Special action for robot as goalkeeper
$a_{3,i}$	`RushToGoal`	Lead the ball to the goal
$a_{4,i}$	`TryToCatchBall`	Try to get the ball from the opponent robot
$a_{5,i}$	`TryToPassBall`	Try to pass on to teammates
$a_{6,i}$	`TryToShoot`	Try to kick the ball into one of the four-goal positions
$a_{7,i}$	`CloseAssist`	Move closer to support teammates who have the ball
$a_{8,i}$	`UnMarking`	Unmark
$a_{9,i}$	`BlockShooting`	Move to be able to cut the ball of the opponent
$a_{10,i}$	`BlockPass`	Move to block the direction of the opponent's movement

TABLE I
LIST OF POSSIBLE ACTIONS (TO BE EXPANDED OR REDUCED)

the same internal state. We then can calculate the probability of successful scoring of each robot at each time k.

Assumption 3: **All $R_{i,j}$ computations for each robot will be synchronous at each iteration k.**
The assumption 3 to ensure synchronization in the calculation of reward points for each action of each robot is identical.

In our research, there are two policies: a global policy and a local policy are proposed. A global policy aims at providing strategies to the whole team (defensive or offensive) or orders to stop the game due to the referee, corner, or ball being out of the field. A local policy is a strategy to choose a specific action for each robot. This also led to the division of the internal state of each robot (which means that the position of each robot is shared on the communication network) and the external state of the robot (e.g. the robot determines its position with respect to the environment) (see figure 2). This study defines several possible actions for each robot (see Table I).

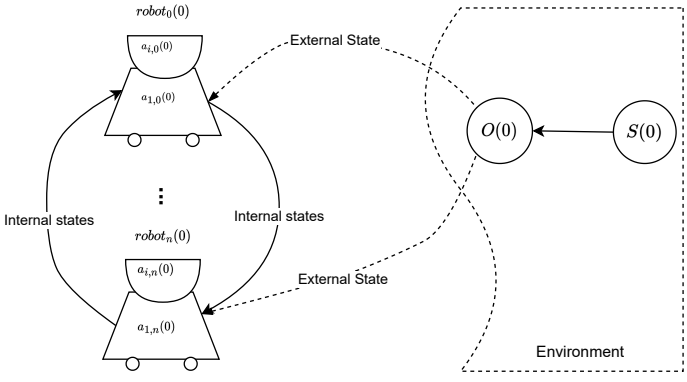

Fig. 2. A novel approach to Dec-POMDP for a scenario with two agents. The agents communicate and exchange their own observations, which allows them to keep their internal state consistent.

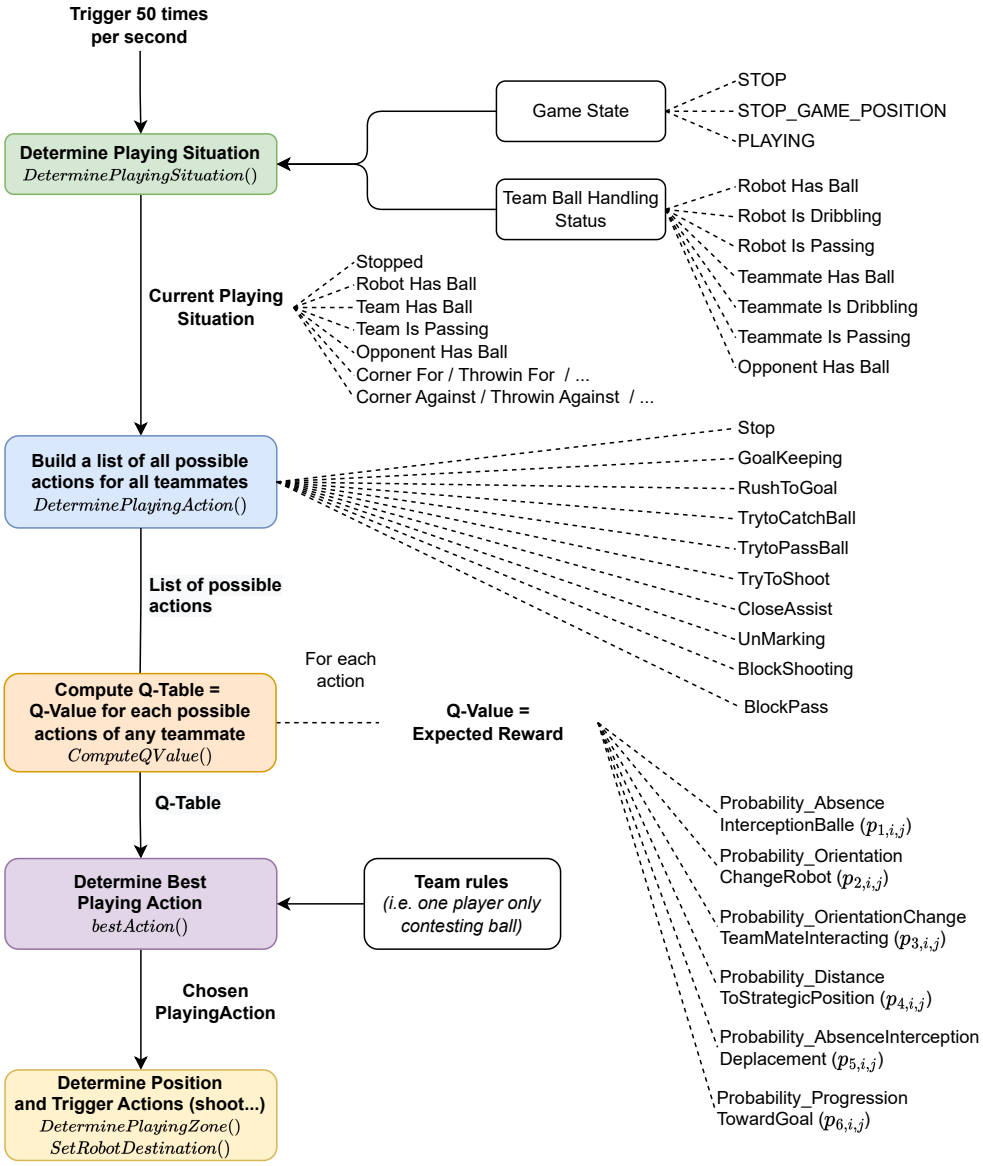

Fig. 3. Strategy algorithm of a soccer team

Our objective is to let the robots decide their best action according to team rules (such as one player maximum contesting a ball) and according 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 choose actions to their robot teammates. This means each robot has to imagine its best action and its teammate's best actions to make a choice.

Assumption 4: **if $c_{1,i}$ is the position of robot i of team 1, $c_{2,j}$ is the position of robot j of team 2, c_b is the position of the ball, We assume that if $c_{i,j} = c_b$, which proves that robot j of the team i has the ball.**

Figure 3 describes the action selection algorithm. Every 20ms, the following steps are done :

- Step 1: the 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) In this study, we used some of the actions listed in table 1.
- Step 3: for each of these actions, the expected reward (Q-Value) is computed based on the equation 2, 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, which 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 $a^*_{i,j}$. This is the end of the decision algorithm.

$$a^*_{i,j} \leftarrow \max_{a_{i,j}} \{\mathcal{R}_{i,j} : i = 1\ldots, 10; j = 1\ldots, 5\} \quad (3)$$

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

By using the approach that gives the highest score based on the probability of success for each specific action, each robot can quickly make decisions based on real time . Moreover, this approach does not require high computing power for each robot.

IV. Analysis of an example of decision making

To illustrate our algorithm, we will execute a scenario like Figure 4. The present scenario entails two football teams, distinguishable by the colors green and pink, respectively. The ball, as of the present moment, is under the possession of robot 1 of the green team. The positions of the robots at time k are respectively $c_{1,1} = (3, 8)$, $c_{1,2} = (5, 6)$, $c_{1,3} = (4, 4)$. Based on the positions of the current robots, they must make action decisions. For simplicity in this case we will only calculate reward points for two actions a_5 `TryToPassBall` and a_6 `TryToShoot` for each robot. Based on the table II, III, IV, we can see that robot 2 will have the highest probability of scoring (corresponding to $p_{6,6,2}$). This also leads to the highest reward ($R_{6,2}$) for robot 2's scoring action. Thus, it can be concluded that the optimal strategy for robot 1 would be to execute a ball transfer to robot 2. The following table shows an example of the Reward spreadsheet for robot 2 based on probabilistic parameters.

-	$p_{1,i,1}$	$p_{2,i,1}$	$p_{3,i,1}$	$p_{4,i,1}$	$p_{5,i,1}$	$p_{6,i,1}$	$R_{i,1}$
a_1	-	-	-	-	-	-	-
a_2	-	-	-	-	-	-	-
a_3	-	-	-	-	-	-	-
a_4	-	-	-	-	-	-	-
a_5	0.1	1	0.1	0.1	0.1	0.1	10^{-6}
a_6	0.1	0.1	0.1	0.1	0.1	0	0
a_7	-	-	-	-	-	-	-
a_8	-	-	-	-	-	-	-
a_9	-	-	-	-	-	-	-
a_{10}	-	-	-	-	-	-	-

TABLE II
An example of the Q table for each action of the robot 1

-	$p_{1,i,2}$	$p_{2,i,2}$	$p_{3,i,2}$	$p_{4,i,2}$	$p_{5,i,2}$	$p_{6,i,2}$	$R_{i,2}$
a_1	-	-	-	-	-	-	-
a_2	-	-	-	-	-	-	-
a_3	-	-	-	-	-	-	-
a_4	-	-	-	-	-	-	-
a_5	0.1	0	0.1	0.1	0.1	0.1	0
a_6	0.1	0.1	0.1	0.1	0.1	0.75	7.5×10^{-7}
a_7	-	-	-	-	-	-	-
a_8	-	-	-	-	-	-	-
a_9	-	-	-	-	-	-	-
a_{10}	-	-	-	-	-	-	-

TABLE III
An example of the Q table for each action of the robot 2

-	$p_{1,i,3}$	$p_{2,i,3}$	$p_{3,i,3}$	$p_{4,i,3}$	$p_{5,i,3}$	$p_{6,i,3}$	$R_{i,3}$
a_1	-	-	-	-	-	-	-
a_2	-	-	-	-	-	-	-
a_3	-	-	-	-	-	-	-
a_4	-	-	-	-	-	-	-
a_5	0.1	0	0.1	0.1	0.1	0.1	0
a_6	0.1	0.1	0.1	0.1	0.1	0.5	5×10^{-7}
a_7	-	-	-	-	-	-	-
a_8	-	-	-	-	-	-	-
a_9	-	-	-	-	-	-	-
a_{10}	-	-	-	-	-	-	-

TABLE IV
An example of the Q table for each action of the robot 3

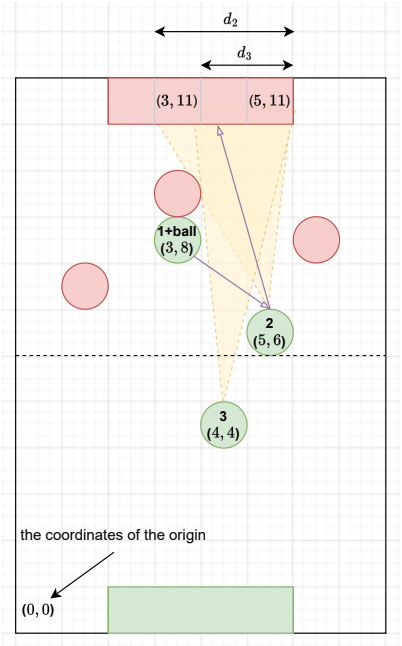

Fig. 4. A scenario that requires a robot to make a decision

Currently, we are engaged in the development of a software application that can effectively assess these algorithms in a convenient manner. This undertaking is in progress and aims to provide a streamlined approach to algorithm testing. Figure 5 shows 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,

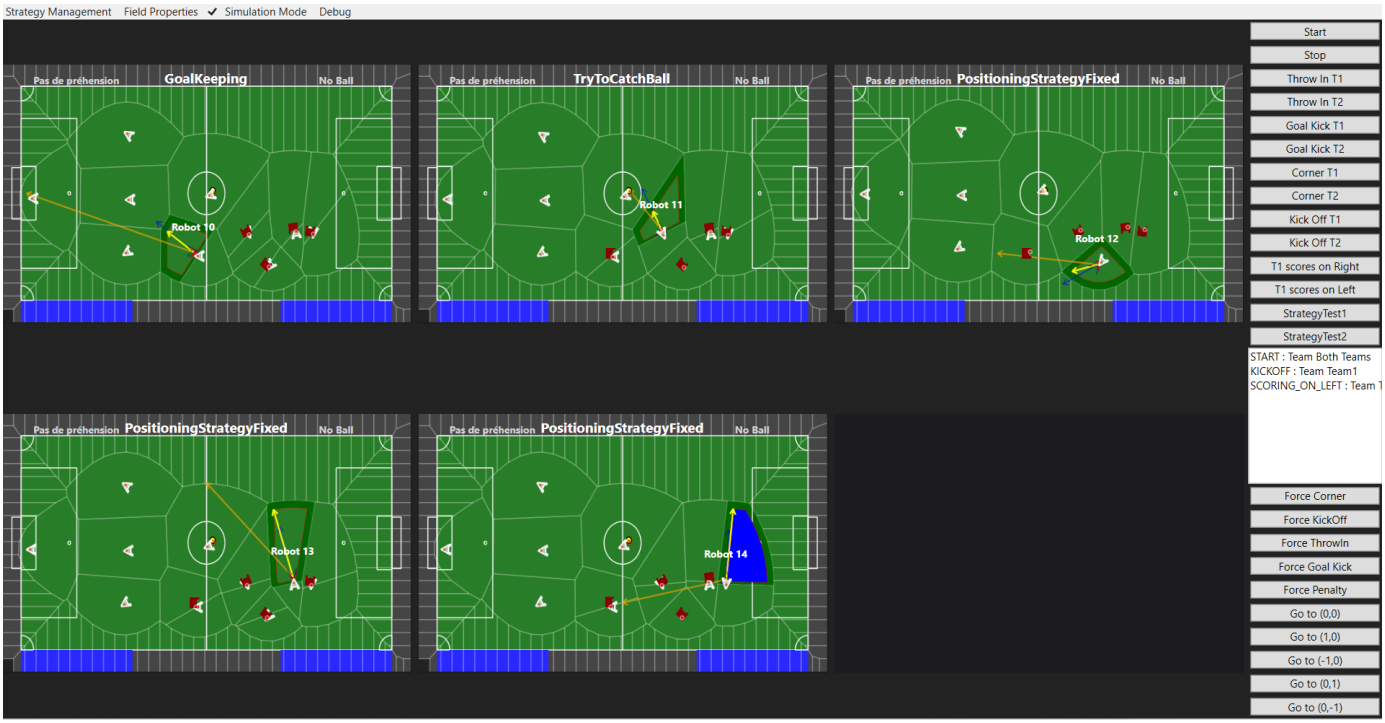

Fig. 5. Example of coordinated strategies for a group of robots

robot number 11 is performing **TrytoCatchBall** action, and robots $12, 13, 14$ are in **PositioningStrategyFixed** state.

V. Conclusion

This study provides a novel approach for modeling and competing soccer teams using a specialization of the Dec-POMDP framework, which has the potential to improve the performance of soccer teams by taking into account the complex interactions and dependencies among the robots.

In future research, we aim to enhance the efficacy of our reward point methodologies for decision-making in scenarios where robots possess limited information regarding environmental parameters. Moreover, it enhances the decision-making prowess of each robot in scenarios characterized by environmental uncertainty. Furthermore, the calculation of the Q value involves a large number of parameters for each possible action. These parameters will be optimized using reinforcement learning techniques shortly. In addition, the realization and development of software allowing to implementation of coordination strategies between robots quickly will be invested.

Acknowledgment

This work was funded by the French ANR/AID agency under the RoboSCo project, and also by the QUARTZ laboratory.

References

[1] G. Giantamidis and S. Tripakis, "Learning moore machines from input-output traces," *CoRR*, vol. abs/1605.07805, 2016.

[2] M. Shahbaz and R. Groz, "Inferring mealy machines," in *FM 2009: Formal Methods* (A. Cavalcanti and D. R. Dams, eds.), (Berlin, Heidelberg), pp. 207–222, Springer Berlin Heidelberg, 2009.

[3] W. M. P. van der Aalst and A. Berti, "Discovering object-centric petri nets," *CoRR*, vol. abs/2010.02047, 2020.

[4] R. David, "Grafcet: a powerful tool for specification of logic controllers," *IEEE Transactions on Control Systems Technology*, vol. 3, no. 3, pp. 253–268, 1995.

[5] D. Harel, "Statecharts: a visual formalism for complex systems," *Science of Computer Programming*, vol. 8, no. 3, pp. 231–274, 1987.

[6] T. Henzinger, "The theory of hybrid automata," in *Proceedings 11th Annual IEEE Symposium on Logic in Computer Science*, pp. 278–292, 1996.

[7] L. Rabiner and B. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[8] F. A. Oliehoek and C. Amato, *A Concise and Introduction to and Decentralized POMDPs*. Springer, 2015.

[9] L. Matignon, L. Jeanpierre, and A.-I. Mouaddib, "Coordinated multi-robot exploration under communication constraints using decentralized and markov decision and processes," in *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[10] O. Asik, H. Levent, and Akin, "Solving multi-agent decision problems modeled as dec-pomdp: A robot soccer case study," in *RoboCup 2012* (X. Chen *et al.*, eds.), vol. 7500 of *LNAI*, p. 130–140, Springer, 2013.

[11] C. Amato, G. Konidaris, L. P. Kaelbling, and J. P. How, "Modeling and planning with macro-actions in decentralized pomdps," *Journal of Artificial Intelligence Research*, no. options, 2019.

[12] J. V. T. de Sousa Messias, *Decentralized Decision-Making for Real Robot Teams Based on POMDPs*. PhD thesis, Instituto Superior Técnico, 2012.

[13] J. Capitan, M. T. Spaan, L. Merino, and A. Ollero, "Decentralized multi-robot cooperation with auctioned pomdps," *IEEE International Conference on Robotics and Automation*, 2012.

[14] Busoniu, Babuska, and D. Schutter, "Multi-agent reinforcement learning: An overview," *Chapter 7 in Innovations in Multi-Agent Systems and Applications – 1, vol. 310 of Studies in Computational Intelligence, Berlin, Germany Springer*, no. 10-003, 2010.

[15] R. Lowe, Y. Wu, A. Tamar, M. University, U. Berkeley, and U. Berkeley, "Multi-agent actor-critic for mixed cooperative-competitive environments," *arXiv*, 2020.

[16] S. Tiomkin and N. Tishby, "A unified bellman equation for causal information and value in markov decision processes," 2018.