FC Portugal 2009 - 2D Simulation Team Description Paper

Luís Paulo Reis[1,3], Nuno Lau[2,4], and Luís Mota[1,3]

[1] FEUP - Faculty of Engineering of the University do Porto, Portugal
lpreis@fe.up.pt
[2] DET - Electronics and Telecommunications Department, University de Aveiro, Portugal
[3] LIACC - Artificial Intelligence and Computer Science Lab., University do Porto, Portugal
[4] IEETA - Institute of Electronics and Telematics Engineering of Aveiro, Portugal

Abstract. FC Portugal 2007 project intends to continue the research performed during the development of previous FC Portugal RoboCup simulation league teams. Since our team has continuously been being developed for several years, its' skills and consistency have become out of date. We therefore decided to redesign the team from scratch, using a novel, open architecture that relies on publicly available low level skills.

Our teams have one of the best sets of results in previous competitions. FC Portugal won the simulation league in RoboCup2000 - Melbourne, got third in RoboCup2001 Seattle, won two European championships (Amsterdam2000 and Paderborn2001). FC Portugal team also won the 2002 Coach Simulation in Fukuoka and achieved two second place awards in this competition (Padova, 2003 and Lisbon, 2004).

1 Introduction

The main research goal of FC Portugal team is the development of a formal model for the concept of team strategy, for a competition with an opponent team having opposite goals, general enough to be instantiated to various dynamic competitive domains. The formal model enables the design of an agent architecture suitable for RoboCup simulation league agents. The project research focus is also concerned with developing general decision-making and cooperation models for soccer playing. Cooperation mechanisms include developments of the previously proposed Situation Based Strategic Positioning and Dynamic Positioning and Role Exchange Mechanisms. These mechanisms have proven their validity by being adopted by several teams in different leagues, namely by 2008's Mid-size champions, Cambada[1].

Coaching is also an important research topic in RoboCup. We have proposed Coach Unilang[2] - a general language to coach a (robo)soccer team. Our coach conveys strategic information to players, while keeping their individual decision autonomy.

The development and optimization of low level skills has never been the focus of FCPortugal. Since research has been conducted on the same team for the last years, the originally developed skills have become out of date, and therefore lost some of their competitiveness. In order to cope with that problem, both in the short and long term, we have proposed a novel architecture, which we designate *Common Framework* for Cooperative Robotics[3], that aims at being open, flexible, robust and fault-tolerant .

In order to allow the easy implementation of coordination between robotic team members, we have also introduced the concept of Setplay[4], a simple and flexible small plan definition language.

The rest of the paper is organized as follows. Section 2 describes the new, open FC Portugal agent architecture. Section 3, briefly describes some of the high-level decision and cooperation algorithms developed by the team. Section 4 describes new work on flexible Setplays definition. The last section contains the paper conclusions.

2 Agent Architecture

Development and optimization of low level skills has never been the main focus of the FCPortugal team. Due to this fact, in recent years, and while working on other research issues, the team's original actions and low level skills have gradually gone out of date and lost their competitiveness. Therefore, in FCPortugal's case, it would be advantageous to be able to work on the relevant research issues independently, or abstracting, from the low level skills. Also, the research the teams aims at applying to different leagues and being independent from the specific implementation of skills and perception.

The *Common Framework for Cooperative Robotics*[3], is a new robotic architecture that intends to be applicable to different leagues. This architecture relies on a multi-agent system (MAS) paradigm, and addresses the following set of requirements:

Open Architecture The architecture should be open, allowing the real-time addition and withdrawal of components without compromising it's stability;

General application High-level components should be applicable to different leagues without further customisation;

Redundancy The architecture should allow the coexistence of redundant components, which may be co-ordinated, or selected, by other components.

The *Common Framework* includes a knowledge representation structure capable of representing organised information pertaining to the robotic soccer domain. In order to control different (simulated and real) robots, the *Common Framework* needs specific components that deal with each agent's perception and action capabilities. Low-level skills and perception mechanisms will be designed for each type of robot, while high-level actions can be chosen through the same, league-independent, decision-making component. A general action vocabulary will be developed to enable the low-level action components to understand high-level decision-making, whereas a perception vocabulary will address the representation of state-of-the-world information.

In order for the *Common Framework* to be truly flexible, allowing the integration and replacement of components in real time, it requires a flexible architecture that can be modified both in real and compile time. It is argued that the best way to answer these requirements is through a multi-agent system.

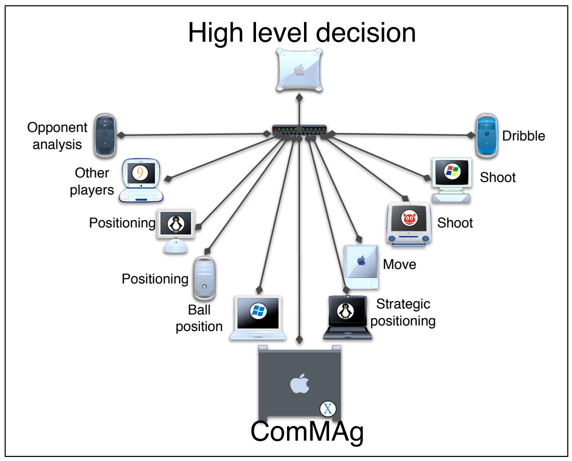

Fig. 1. Proposed architecture: arbitrary services, both in nature and quantity, can connect to the system.

It is proposed to use a multi-agent system (MAS) for the control of each player. Thus, one team would be a system of multiple multi-agent systems. In each of the players, the same kind of components will exist (perception, action, decision, etc.), taking part on a MAS while using standardised communication. Each component might be implemented using a different programming language, or even be running in different machines, with distinct operating systems, as seen on Fig. 1. The components can arbitrarily vary in number, and even be redundant.

In order for the different components to interact freely, it is necessary that they have a way of knowing/discovering each other. With this purpose, there will be a communication management agent (ComMAg) that will keep information on the existing components. Furthermore, this architecture requires a standardised communication language, for the expression of perception, action and state-of-the-world information. This language includes basic concepts, such as regions, locations and time, as well as soccer related items.

The *Common Framework* allows the same high-level controller to decide independently from the low-level skills and perception frameworks. This will allow our players to rely on different, and redundant, low-level implementations. As a starting point, the team has been working on the adaptation of the (publicly available) Helios[5] and UvA-Trilearn[6] codes to the *Common Framework*.

To enable a team to perform cooperative multi-agent tasks, like playing simulated soccer, in a partially cooperative, partially adversarial environment a lot of knowledge is needed. Also, agents must have a world state representation as updated and as accurate as possible. For this type of domains we argue that to correctly perform cooperative tasks, agents should include knowledge at three levels: individual action execution, individual decision-making and cooperation. Knowledge for executing actions is concerned with the specific commands needed to perform a given low-level action. Individual decision-making knowledge is concerned with the way agents choose the action to execute (from the available set of actions). Knowledge for cooperation is concerned with tactics, situations, dynamic formations, roles, dynamic plans and communication protocols [7]. Representation of structures for this type of multi-level knowledge is one of our research goals, and has been carefully considered in the design of the *Common Framework*.

3 High-level decision and Cooperation

We are extending our Dynamic Positioning and Role Exchange mechanism (DPRE)[7], that is based on previous work by Stone et al[8]. How to define roles based on standardized agent behavior characteristics for the RoboCup simulated soccer domain is one of the problems to be tackled. To improve the flexibility of our team, agents are able to switch their relative positions (for a given formation) and roles (that define agent behavior at several levels), at run-time, on the field. We have proposed and continually developed Situation Based Strategic Positioning (SBSP) mechanism [7, 9] that may be used to dynamically spatially position a team using different flexible formations for different situations. This mechanism is based on the distinction between active and strategic situations [7]. If an agent is not involved in an active situation then it tries to occupy its strategic positioning that change according to the situation of the game. Situation is a concept on a high-level analysis of the game (attacking or defending for example). SBSP was one of the main innovations of FC Portugal and is now used directly or as the base for the positioning systems of many simulated soccer teams.

4 SetPlays

We have also developed a framework for high-level setplay definition and execution, applicable to any RoboCup cooperative league and similar domains. The framework is based in a standard, league-independent and flexible language that defines setplays, which may be interpreted and executed at run-time [4].

Setplays are commonly used in many team sports such as soccer, rugby, handball, basketball and baseball. There are surely several important differences between robot soccer and human sports, but setplays can nonetheless be a useful tool for high-level co-ordination and cooperation.

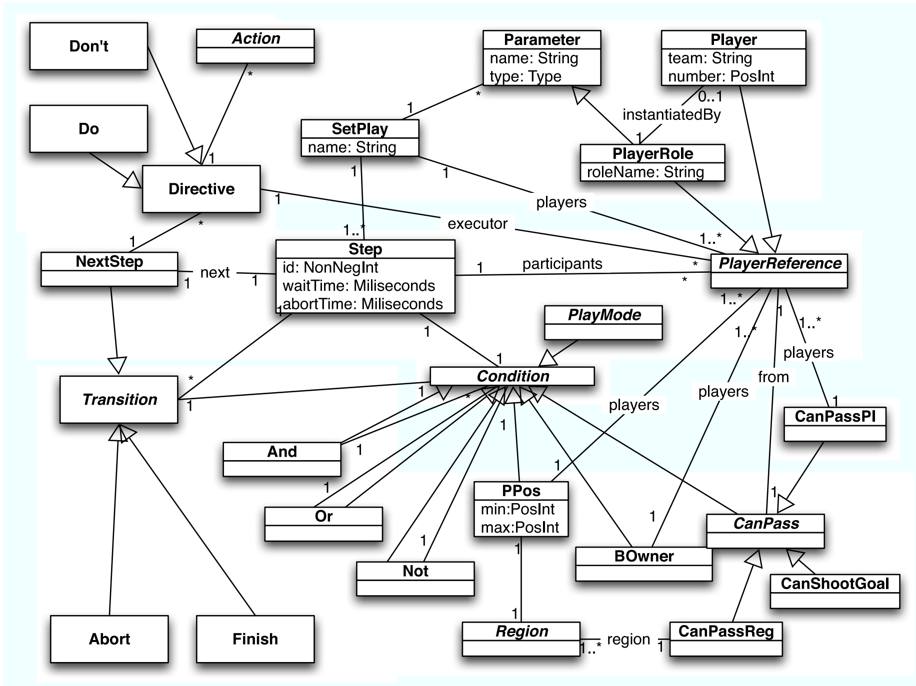

Fig. 2. Setplay definition

The Setplay framework was designed with the goal of being general, flexible, parameterizeable and applicable to any robotic soccer league. Its' general structure is shown schematically in Figure 2.

At the top level, a *Setplay* is identified by a name, and has *parameters*, which can be simple data types like integers and decimals, or more sophisticated concepts as *points* and *regions*. *Setplays* also have *Player References*, which identify players taking part in the Setplay. The *Player References* can point to specific players, or be *Player Roles*, i.e., abstract representations of a particular role in the *Setplay*, identified by a name (e.g., attacker, supporter). *Parameters* and *Player Roles* will be instantiated at run-time.

Steps are the main building blocks of a *Setplay*. A *Step* can be seen as a state in the execution of a *Setplay*. A *Step* has an *id*, which is a non-negative integer. In order to control the *Step*'s execution, the concepts of *wait time* and *abort time* are introduced. A *Step* also has a *Condition*, which must be satisfied before entering the *Step*.

There are several possible ways out of a *Step*, which are defined as *Transitions*. All *Transitions* can have a *Condition*, which must be satisfied for the *Transition* to be followed. Possible transitions are *Abort Transition* , *Finish Transition* and *NextStep*, that is used to link between the different *Steps*. It includes the id of the next *Step* to be reached, and contains a list of *Directives* that will be applied in order to accomplish the *Transition*.

5 Conclusions

FC Portugal 2009, as its predecessors, is a team with a beautiful, fast, 'real soccer like' way of playing simulated soccer. The improvements performed during this year, namely, the usage of setplays, make the team even more flexible and adaptable to different types of opponent strategies. Furthermore, the new agent architecture opens a wide set of possibilities, as it allows the same high-level controller to work with different low-level skills and perception frameworks.

In 2007, the team won the second place in the German Open, losing the final only after penalty kicks. After a year away from the championships, to allow the redesign and redevelopment of the team, FCPortugal aims at restarting the research effort, based on the new architecture.

References

1. Azevedo, J.L., Lau, N., Corrente, G., Neves, A., Cunha, M.B., Santos, F., Pereira, A., Almeida, L., Lopes, L.S., Pedreiras, P., Vieira, J., Martins, D., Figueiredo, N., Silva, J., Filipe, N., Pinheiro, I.: Cambada'2008: Team description paper. Technical report, University of Aveiro (2008)
2. Reis, L.P., Lau, N.: Coach unilang - a standard language for coaching a (robo) soccer team. In: RoboCup-2001: Robot Soccer World Cup V. Volume 2377 of Lecture Notes in Artificial Intelligence. Springer Verlag (2002) 183–192
3. Mota, L., Reis, L.P.: A common framework for co-operative robotics: an open, fault tolerant architecture for multi-league robocup teams. In Carpin, S., Noda, I., Pagello, E., Reggiani, M., von Stryk, O., eds.: International Conference on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2008). Volume 5325 of Lecture Notes in Computer Science., Springer (2008) 171–182
4. Mota, L., Reis, L.P.: Setplays: Achieving coordination by the appropriate use of arbitrary pre-defined flexible plans and inter-robot communication. In Winfield, A.F.T., Redi, J., eds.: First International Conference on Robot Communication and Coordination (ROBOCOMM 2007). Volume 318 of ACM International Conference Proceeding Series., IEEE (2007) 13
5. : rctools Web Page, http://rctools.sourceforge.jp/akiyama/
6. Kok, J.R., Vlassis, N., Groen, F.: Uva trilearn 2004 team description. Technical report, University of Amsterdam (2004)
7. Reis, L.P., Lau, N., Oliveira., E.: Situation based strategic positioning for coordinating a simulated robosoccer team. In Hannebauer, M., Wendler, J., Pagello, E., eds.: Balancing React. and Social Deliberation in MAS. Volume 2103 of LNAI., Springer (2001) 175–197
8. Stone, P.: Layered Learning in Multi-Agent Systems. PhD thesis, CMU (1998)
9. Reis, L.P., Lau, N.: FC Portugal team description: Robocup 2000 simulation league champion. In Stone, P., Balch, T., Kraetzschmar, G., eds.: RoboCup-2000: Robot Soccer World Cup IV. Volume 2019 of Lecture Notes in Artificial Intelligence., Berlin, Springer Verlag (2001) 29–40