MICROB: The French Experiment in RoboCup

Sébastien Rocher[1], Vincent Idasiak[1]; Sébastien Doncker[2]; Alexis Drogoul[2] and Dominique Duhaut[1]

[1] Laboratoire de Robotique de Paris, Université PARIS 6, UVSQ, CNRS, France
[2] LAFORIA, Université PARIS 6, CNRS, France

Abstract. In this paper, we present the ongoing research done in the field of robots playing "football". Several development aspects are discussed such as the hardware system, followed by the software implementation. We will highlight three softwares : the central decision system and the embedded software (used for the real robots small RoboCup league), and the simulation software (used for the simulation RoboCup league). So, we will show and discuss, for each kind of competition, the implemented behaviors.

1 Introduction

The idea of robots playing "football" came at the same time within two fields of research : robotics and computer science [4, 5]. We decided to work on this subject at the end of 1995. The idea was to build and exprimental platform : MICROB, to validate on real robots different kind of multi-agents algorithms. Thus, the MICROB platform is well adapted to solve such problems as the football game.

In the first part, we present the team participating for the small league of the RoboCup competition and we focus on the decision system.

In the second part of the paper we present the simulation team. Here, we tried to design a team in which all the players processes the same behaviour. We discuss the limits of the above approaches.

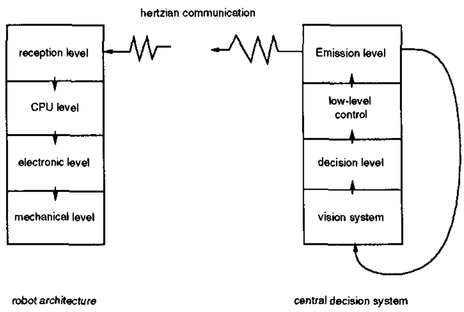
2 The MICROB team for small size competition

Our platform is divided in two parts : the robots and the central decision system (*Fig 1.a*) .

2.1 The robots architecture

The physical structure of the robots (*Fig 1.b*) is divided in four layers *(Fig 1.a)*:

- the **mechanical layer** : the robots designed have two independent wheels on the middle of their structure. They have two degrees of freedom in the plane and they are non holonomic. Each wheel is motorized by an actuator consisting on a small motor. The motor power supply is carried by the robot and consist on a small battery 12V.



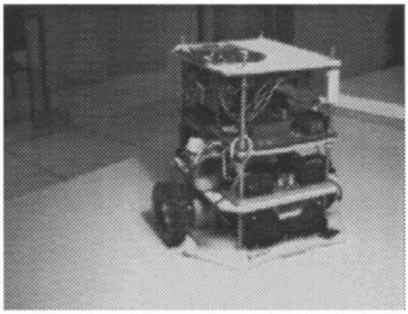


Fig. 1. *a) The general platform architecture b) One of our small size robot*

- the **electronic layer**, which uses a tachometric feed-back.
- the **CPU layer** : the CPU is a 80C552 processor, with a 16 MHz frequency and 64 Kbytes of memory. It has only to decode the data coming from the hertzian receiver of the robot.
- the **communication layer** : it receives orders from the central decision system using an hertzian transmission (9600 bauds).

2.2 The central decision system

For the central decision system, we use an Intel-Pentium 90 CPU with 32 Mbytes of memory and MS-DOS operating system. In this section, we describe the vision system, and the behavior module (the theoretical behavior and the experimental one).

The vision system. The vision system takes a large part in the global system performance [6]. It has to extract pertinent data from video acquisition :

- the **position** and the **orientation** for each robot,
- the **position** and the **velocity vector** of the ball.

Moreover, because of the real-time constraints, the acquisition and the processing of a video frame has to take less than 10 Hertz. The acquisition card used (the IFG Imasys frame grabber card) provides black and white video. The image processing extracts the outline of the objects located on the playing field. To identify the robots, we put various marks on the top of each of them. The directions of the robots are determined using the knowledge of their outline. For the ball, the velocity vector is deduced using the previous detected positions. Due to the time processing, and the resolution of the image (256×256), one pixel maps $1cm^2$, which affects the precision reliability of the system.

The theoretical behavior. A multi-agent architecture (*fig 2.a*) is proposed to define the behavior of the robots (in our study the agents are the robots). This architecture is composed of four main modules : a module representing the

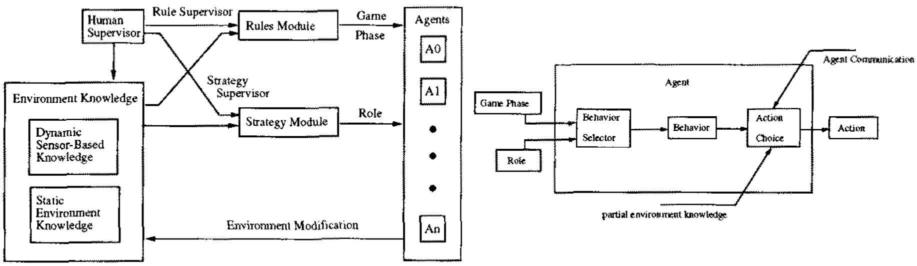


Fig. 2. *a) The general Multi-agent ARCHitecture b) The agent model*

knowledges about environment, a **rules module**, a **strategy module** and a **set of agents**.

• ***The environment knowledge.*** In our system, there are two kinds of environment knowledge:

- the **static knowledges**: the limits of the playing field, the kickoff position and the penalty position.
- the **dynamic knowledges**, which are computed on-line : the position and the velocity vector of the ball, the position and the orientation of the robots.

• ***The rules module.*** The football game is divided in game phases. The different phases we implemented are : the **kick-off** phase, the **free kick** phase, the **penalty kick** phase, the **stop** phase,and the **main game** phase. The rules module allows to define the current phase of the game. It acts mostly under the influence of the human supervisor. Indeed, a human supervisor can decide to stop, to start a game, etc... In certain cases, the module of rules allows to change game phases automatically (for instance, from the kick-off phase, the penalty kick phase or the free kick phase to the main game phase).

• ***The strategy module.*** The goal of the strategy module is to define the role of each agent within the team. This role can be defined in a static or a dynamic way, according to the environment knowledge. Two main roles are defined : the **goalkeeper** role, the **field player** role. Moreover, specific roles can be added to the main ones. These are : the **penalty kicker** role, the **Kick-Off kicker** role.
At most, one robot can have the goalkeeper role during the game. To be efficient, the field player role must be different for each robot of the application. The idea is to distinguish the field player roles by giving to each robot a specific zone on the playing field. So, we will have sub-roles of the **main field player** role : we will be able to have a defender player, a middle field player, a attaquant player, etc. This will be developed in the next section.

• ***The agent behaviors.*** In this approach, each robot is an agent. Two functions are used to select the behavior (function s) (1) and to select the action of a robot (function f) (2) :

$$Behavior = s(Role, GamePhase). \tag{1}$$

$$Action = f(Behavior, EnvironmentKnowledge, Communication). \tag{2}$$

The behaviors are selected according to the role of the agent and the phase of the game (*fig 2.b*). Thus, the selected behavior will define the action to be executed by the robot, according to the environment knowledge and the inter-agent communication. The behaviors will have to obey to the game rules. The different behaviors implemented are : the **specific goalkeeper** behavior, the **specific field player** behavior, the **keep stopped** behavior, the **kickoff player** behavior, the **penalty kick player** behavior.

The experimental behavior. In this section, we will develop the implementation of the main behaviors (the goalkeeper behavior and the specific field player one) for the RoboCup experiment. Two configurations were realized to adjust to the robot number of the opponent team : one game with a five players team and one game with a two players team. Indeed, due to the competition requirements, we played with 2 robots against the japanese team. Thus, we had to change our strategy.

• ***The actions.*** We distinguish two kinds of actions : the **high-level** actions and the **low-level** ones. The high-level actions define goals to be realized, such as : replacement, ball kicking, avoidance, move for a position with an orientation. These high-level actions will be divided into low-level actions. These low-level actions are : the **forward gear**, the **reverse gear**, and a **position and orientation low-level servoing**. The third action is the most interesting. To implement it, we used a control system well adapted to our robots configuration, and to our problematic. It allows a position and orientation servoing, using a constant linear velocity of the robot [2].

• ***The goalkeeper behavior.*** The goalkeeper behavior is very simple : during the game, we place the goalkeeper on a straight segment, between the ball and the goal.

• ***The specific field player behavior.*** It uses different zones of influence for each different field player behavior. The algorithm of the behavior can be resumed in a pseudo-code like this :

IF (the ball is not in the zone of influence of the agent)
   THEN ***Replacement.***
ELSE
   IF (the agent is not well placed to kick the ball)

THEN ***Replacement.***
*ELSE /** ***Beginning of contract net*** **/*
the agent ***communicate to other agents the evaluation of his position to kick the ball.***
IF (the agent is the best placed)
THEN the agent ***kick the ball.***
ELSE
Replacement.
ENDIF
ENDIF
ENDIF

The idea is to have stackables zones of influence in the robot team. So, each time, several robots can kick the ball to attack or to defend. Finaly, only one robot will kick the ball (the one which is the best placed), and the other ones execute a replacement.

• ***The five players game.*** For the game with five players, we defined the following roles : one goalkeeper, one defender player, one left middle field player, one right middle field player and one attaquant. To define the zone of influence assigned to each robot, we decided to divide the game field into 9 parts (*fig 3*). Then, each field player has a specific zone of influence. We also have to define

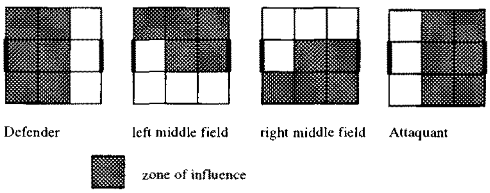


Fig. 3. *Zones of influence for each field player robot*

the replacement position. In our experiment, the replacement position depends onle on the ball position (*fig 4*). During all the five players game, the agent role

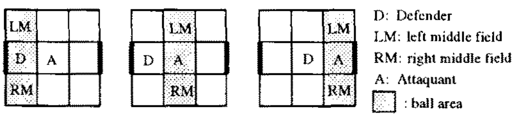


Fig. 4. *Replacement positions of robots, according to the position of the ball*

was always the same.

• ***The two players game.*** The game with two players involving the japanese team is different from the one with five players. Indeed it is much more difficult to kick the ball with two players than with five. So, we used this configuration : one defender/goalkeeper player and one attaquant player. For the game with only two agents, we decided to give the maximum of freedom to our robots. So the zones of influence are the same for the two robots and they represent all the game field. But the behaviors are not exactly the same for the two robots. Indeed, the replacement position is different (*fig 5*). This configuration is

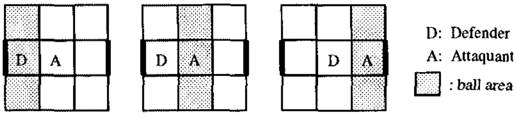


Fig. 5. *Replacement positions of robots, according to the position of the ball*

particularly interesting, because it allows us to test the role selection function. Indeed, the role of one of the robots is dynamic. It plays as a defender or as a goalkeeper according to the environment knowledge. If the ball is in our camp and if the direction of the ball was toward our goal, then the strategy module imposes goalkeeper role to one agent.

2.3 The limits of the team

The limits of our robots team are due partly to the low quality of equipment used. Actually, to carry out our experiment, we used a cheap equipment : a PC (pentium 90), a black and white acquisition card with an acquisition rate of 18 Hz. For the robots, we used secondhand actuators, and mechanical engineering and the electronic card have been realized by our own. The cost of all the equipments is approximatively 5000 dollars.

The main problem arise from the vision system : the vision system is too slow and the precision of the detection is inaccurate. Indeed, the global precision of our system is about 2 or 3 cm (regarding robots speed). This problem made useless the implementation of evolved actions and so, the behaviors were limited.

The second problem is the low quality of the hertzian transmission between the central decision system and the robots. Now the path processing being computed by the central system, the robots are very dependent on the communication chanel quality.

3 The Microb simulation team : exploring a strict bottom-up approach

Unlike the approach used for the real robots, where the goal was to provide existing robots with coherent behaviors regardless of the methodology employed, the development of the simulation team has been initiated to track the methodological issues raised by the soccer problem (see [1] for a different perspective on the same issue), and to illustrate some properties of reactive multi-agent systems [3]. The basic idea was to follow a strict bottom-up approach to the problem of organizing a team, beginning with very simple players and supplying them with the behaviors that could appear necessary with regard to their collective play. The hypotheses that have underlined our work, and to which we have remained committed throughout the different versions are :

• **Hypothesis 1 :** behavioral homogeneity . All the team mates are identical. As a consequence, the behavior of the team does not rely on a coordination between specific roles (e.g., the goalie, defenders, etc.), which are therefore not described nor used in the implementation. However, even if the agents are not provided with a given role, their position on the field will induce different behavioral responses from their partners.

• **Hypothesis 2 :** environmental heterogeneity. The activity of reactive multi-agent systems usually rely on the information the environment can provide them. The computation of a simple topology, which divides the field in eleven zones of influence supplied with different marks, is intended to dispatch dynamically the agents on the playground during the game, regardless of their current behavior).

• **Hypothesis 3 :** no direct communication. This restriction lets the agents use the characteristics of the game as a communication-like protocol : passing the ball or entering/leaving a zone is supposed to be sufficiently explicit for the others.

• **Hypothesis 4 :** no planning. The agents do not look ahead. They do not plan their behaviors after the current cycle they are engaged in, nor do they anticipate the behaviors of the other participants.

• **Hypothesis 5 :** no memory. The agents do not record the previous information they have obtained through their perception. Of course, their velocity, as well as their direction, can be considered as a sort of embedded memory of their former behavior, in the sense that it implicitly influences their current behavior.

• **Hypothesis 6 :** no explicit deliberation. Due to the constraints above, the agents do not use explicit deliberative mechanisms for controlling their activity. Rather, a subsumption-like architecture, with a fixed structure, allows them, at each cycle, to behave accordingly to a fixed set of situations they may encounter (*fig 6*).

• **Hypothesis 7 :** a small set of behaviors. The behaviors organized within this architecture are only five : **move to another zone**, **look at the ball** (i.e. continuously changing direction to follow the position of the ball, but without moving), **chase the ball** (i.e. the same with a movement), **kick the ball** (i.e. kick the ball toward the opposite team), **pass the ball** (i.e., kick the ball toward a team mate).

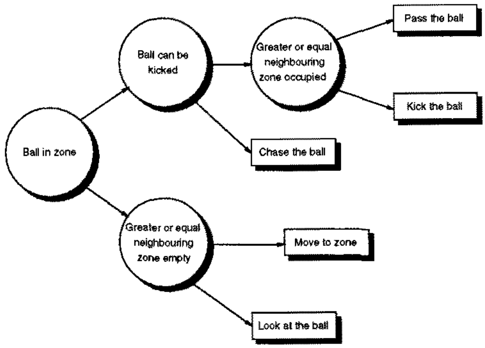


Fig. 6. *The tree-like decision algorithm and the associated behaviors*

Given these hypotheses, one could have expected the collective game to be fairly poor. Indeed, the behavior of a sole player is easily predictable, and it is quickly defeated by any other opponent (like Ogalets's ones, for example). However, collectively, things appear to be a lot more interesting, with truly coordinated movements emerging from the interactions between the simple behaviors of the players. It is especially the case when the team is attacking, elaborating surprisingly good collective actions that can easily beat any defense. The main reason is that the players, because they do not need to communicate nor plan anything, are much faster at moving or passing the ball than most of the teams they have already been facing. But the behavioral homogeneity allows the players to occupy any zone on the field and thus making the team dynamically and smoothly cover the playground.

Of course, we have detected, during the experiments, numerous drawbacks for such a simple approach :

- The players often miss the ball because they are always heading toward it regardless of its speed or direction. This lack of anticipation is especially painful for the "goalie" (i.e. the player being placed in the corresponding zone), which does not care about moving until the ball reaches its zone.
- During difficult and long actions, four or five players can happen to chase the ball as if they were alone in their zone. It is usually due to the initial dash which made them leave their zone (without them paying attention to it). Because they can not replace themselves while the ball is too close, the result slightly disordered.

These two defaults combined often ruin the interesting collective behaviors that could otherwise emerge. But the interesting thing is that, given the proposed methodological (and somewhat radical) approach to the problem, we now know exactly what to improve in the individuals' behaviors for dynamically obtaining emergent collective strategies that could remain stable throughout the game. Our approach, of course, which will consist in carefully adding new behaviors, will not prevent us from possible side effects (i.e. unwanted new collective structures). However, it will be certainly the price paid to understand the link between individual and collective behaviors.

4 Conclusion

From experimental observations, we suggest to give futher enhancements to our design. To have good result for the real robots competition, the hardware must be very efficient. Thinking it is possible to control a mechanical structure which is not optimal, is not a good idea. Having good result means having a good software, but also a good hardware. Each layer is important : control, trajectory generation, behaviours, vision, communication. We tried to use some cheap components thinking that a good behaviour could correct the uncertaincy of the system. This is not reasonable. For the next competition we can expect to have the best at each layer.

The future of the simulation team relies in our capacity to explore the link between individual and collective beaviors verifying hypotheses presented above. The important thing is that collective responses to unpredictable situations appear to be often qualitatively surprising, especially when facing good teams. But the issue is that the robots can not take advantage of their experience of these collective achievements, because they have no available mean to adapt their behaviors (both the way they classify the situations and their reaction to them) with respect to the previously encountered situations. As a consequence, learning capabilities, built on top of their current behaviors, are already being provided to our players (this requires us to release some constraints, such as the absence of memory). And we intend to prove in a near future that having adopted a behavioral bottom-up approach greatly eases this process.

References

1. Collinot A., Drogoul A. and Benhamou P., "Agent Oriented design of a Soccer Robot Team", Proceedings of ICMAS 96, AAI Press, Nara, Japan, 1996.
2. S. Delaplace, "Navigation a cout minimal d'un robot mobile dans un environnement partiellement connu" *Ph.D thesis, University of Pierre et Marie Curie (Paris 6)*, France, 1991.
3. Drogoul A., Corbara B., Lalande S. "MANTA: new experimental results on the emergence of (artificial) ant societies", in Gilbert N. and Conte R. (Eds.) Artificial Societies, UCL Press, 1995.
4. H. Kitano, Y. Kuniyoshi, I. Noda, M. Asada, H. Matsubara, E. Osawa, "Robocup: a challenge problem for AI," *AI magazine Vol. 18 N°1*, pp 73-85, 1997.
5. M. K. Sahota and A. Mackworth, "Can situated Robots Play Soccer?", in proc. Artificial Intelligence '94, pp. 249-254.
6. R. Sargent, B. Bailey, C. Witty, A. Wright, "The importance of fast vision in winning the first micro-robot world cup soccer tournament" *Journal of Robotics and Autonomous System Vol. 21, N° 2*, pp 139-147, 1997.