THE DESIGN OF A SMALL-SIZE (F-180) ROBOCUP TEAM

P. Costa[1], A. Moreira[2], A. Sousa[3]

FEUP-DEEC/ISR,
Rua dos Bragas 4099 Porto Codex, Portugal, fax: +351 22 200080,
[1]paco@fe.up.pt, [2]amoreira@fe.up.pt, [3]asousa@fe.up.pt

Abstract: Autonomous Mobile Robotics is a very active research field. The RoboCup challenge provides a standard benchmark environment and the design of a small size RoboCup team is an excellent way to innovate in this field. The evolving design of the team is presented as well as some of the experience accumulated from two participations in this competition. Some key issues are discussed and the adopted solutions presented.

Keywords: Autonomous Mobile Robots, Robotics, Robot Soccer.

1. INTRODUCTION

To design a team of mobile robots that can play soccer, it is necessary a multidisciplinary approach To design a team of mobile robots that can play soccer, it is necessary a multidisciplinary approach combining knowledge from fields as diverse as mechanics, electronics, control, computer science, communications and soccer. This is an on-going project pursued by many other teams like (Rowstron98), (D'Andrea99), (Veloso99) and (Wyeth99).

The kind of soccer expected to be played by the robots is not yet comparable with the human soccer. The teams presented here played in the Robot Soccer World Cup in 98 and 99. The RoboCup Federation organised these events. There are several RoboCup categories, but the classical ones are **small-size** (F180) league, **middle-size** (F2000) league and the **simulation** league. The simulation league runs entirely on a computer with 11 virtual players for each team running in different computers (usually a different computer for each team). The small-size league uses teams of up to 5 robots usually with a global vision based system. The robots play on a field with the dimensions of a Ping-Pong table. The medium-size uses up to 4 robots per team without a global vision system. Other leagues are being proposed and they can be observed in live matches at the 2000 edition, to be held at Melbourne, Australia. The teams presented are from the F-180 League, so all the remaining paper will focus only on this League.

To those not familiar with the RoboCup games it is important to stress the fact that all the gameplay is expected to evolve without human intervention. The team behaviour should be determined only by the decision algorithms implemented in the control system. While the game is being played it is strictly forbidden any human intervention, that can only happen while the game is stopped. A match consists in two parts with ten minutes each. The complete rules can be viewed in the RoboCup official site.

The game is played with an orange golf ball. The goal is 50 cm wide and to ensure that it is not possible to create an impossible to defeat defensive situation (like placing all the robots over the goal line to block it completely) there is a goal keeper area with 22.5 cm, in front of the goal. Inside this area, there can bee only one robot from each team.

The preferred way to acquire the ball and the robots' position is by using a vision system. So, each robot must carry over its top a Table Tennis ball, painted with one of the official colours. One team uses blue balls and the other uses yellow balls. Additional markers are often used to help extract extra information about the robots, like their heading and to identify each of them.

While some teams have attempted to use some kind of local vision, that approach is not very easy due to the small dimensions of each robot. It is very difficult to place inside the robots the memory and the processing power needed to deal with the acquired image.

The main advantage of this kind of competition is that it provides a set-up in which the overall design can be tested and easily benchmarked against other teams. Yearly, we get the possibility of another iteration to measure and latter improve our system. More, the inevitable exchange of ideas plays a very important in fostering the evolution in this field.

This paper starts with a brief description of the official rules that shape the design and their consequences. Section 3 includes a more detailed description of the designs for the 98 and 99 competitions. Section 4 outlines some strategy issues, and finally, in Section 5, some conclusions and future trends are presented.

2. Design contraints

2.1. The Rules

The major constraint design is posed by the rules that define the proper set-up. Another issue is that, from year to year, these rules are updated and some changes can play an important role on future designs, or even break old ones. The inevitably of these changes is even stressed out in the official rules. From 98 to 99, the maximum dimensions for each robot changed. For that year, it was allowed that teams with oversized robots could still compete if they had entered the competition in the previous year. Naturally, this exception only lasted a year. So, it is advisable to choose a design that is not too much dependent on a single issue or too much interconnected, because it can be rendered obsolete if some rule changes. Clearly, a modular design allows an easier evolution. Another advantage of a modular design is that some modules can be easily used in other systems. For example, our radio system was latter used in an industrial project.

First, the rules state that a team is composed by a maximum of five robots. There is no minimum number of robots but it is clearly optimal to have all the five robots playing.

The convex hull of the projected area of any robot must be under 180 cm^2. The maximum dimension, measured in any direction must be under 18 *cm*. Finally, the height must not exceed 15 *cm* for standard robots and 22.5 *cm* for robots carrying a local vision system. These are the dimension constraints, while it is arguable what are the optimal dimensions of the attacker robots, for the robots in the defence, especially the goalkeeper it is advantageous to be near the maximum allowed. For the goalkeeper we found that a shape like a trapezoid, with the base near the goal line, was a very favourable configuration. When the ball comes from the sides this shape deflects the ball away from the goal. Some teams used this shape with the larger side facing the attack and their goalkeeper scored some goals in its own goal when faced balls coming from one of the sides.

2.2. Some Extra Constraints

To achieve a system that could easily be replicated, it should use mostly standard components, preferably inexpensive and widely available such as general consumer products. In this area that, sometimes, is a difficult option, specially if we are trying to reach a certain performance level. Inevitably, certain tradeoffs are required.

The design of the vision, decision and control systems involved the implementation of several algorithms in a PC. This software must work under hard real time constraints. Most common operating systems do not guaranty real time performance. To ensure that our hard deadlines are kept we must either use a real time operating system or code that behaviour in our software. In that case we must take direct control over the hardware. That was the chosen path. We used plain DOS and wrote the necessary drivers to communicate with the video acquisition board. We avoided any call to the operating system keeping always the full control of the processing time.

3. The Design

3.1. Robocup98

The design for that year was described in (Costa98). Some last minute changes resulted in some differences between the described set-up and the one that played in Paris. Even during the games we used the gathered experience to improve our team.

The ’98 team, shown in Figure 3, had a very simple mechanical project with only a floor and a ceiling to hold together the pieces. The embedded processor was a PIC16F84 that was used to listen to the radio packets, sent by the coach, and to send pulses into the two servos. Those modified servos were used as a cheap source of a motor plus a gearbox. The power supply consisted in five *Ni-Cd* batteries with a nominal capacity of 900 mAh each.

Radio communications were made possible using FM modules. These modules can act as emitters or receivers but we used them only as receivers for the ones fitted in the robots and as emitter for the one assigned to the coach. In order to allow a game with uncoordinated radio protocols, it is mandatory to have two frequencies available. Modules with carrier frequencies of 433 *MHz* and 418 *MHz* could be swapped. In each radio packet we had the requested speed for each wheel.

The limited maximum speed of the servos was somewhat improved by using large diameter wheels but still, our robots were among the slowest in the tournament. The robots' maximum speed was around 25 *cm/s*. Wheels should be kept thin in order to increase the dead reckoning accuracy leading to extra accuracy on the model and its parameters' estimates. The global strategy layer was programmed to gather the teams and ball state using a Kalman Filter. By using this global state, that we expect to represent with some accuracy the game state, it should be possible to deploy a responsive co-operative behaviour.

The information about each robot's position and the team to which it belongs was retrieved by the position of the official blue or yellow ball on the top and centre of the robot. An additional marker of another colour was used to retrieve heading. Identifying individual robots was achieved by using a visual bar code composed by white labels over the black top. This visual bar code superseded the previous method of matching the information from previous frames. That algorithm often led to inaccurate matches, usually when a grouping of robots occurred. In this case, a fail to match correctly a pair of robots from our team, led to having that pair getting the wrong orders and thus loosing the control over those two robots, until the next stop in the game play.

The vision system was based on a standard camcorder and the image acquisition was performed by a board intended for web cam communication. This was a PCI board based on the common Bt848 chip. All our software was written for DOS using the DJGPP compiler. This is a port of the free GNU C++ compiler. The frame rate was limited by the PAL standard. We had 25 frames per second (*fps*) because only one the two interlaced frames was used. The resulting resolution of that frame was 384 per 288 pixels, sampled at 16 bits of colour. The hardware platform was a Pentium II at 266MHz.

As can see in Figure 1, we can divide the overall system in two subsystems: the Global and the Local Systems. The Local System is constituted by the robots with the corresponding electronics and firmware. This local system is of limited computing power and we made the choice of concentrating the decision making process on the global layer. This Global system is where the image is processed, the decisions about the team strategy are made and the corresponding actions are forwarded to the robots. Information flows between this two system thanks to a radio link and, obviously, the global vision system that extracts the position of the ball and of each robot.

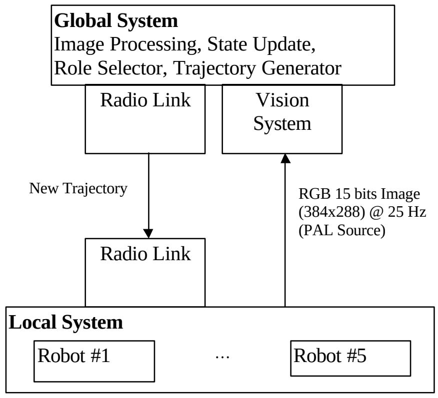

Figure 1 - The Global and Local systems

Our robots were not fitted with any kind of sensor, so there is not any kind of local control loop. The only feedback path exists through the Global System. That setup imposes some delays related to the vision acquisition and analysis and the radio transmission. The speed of our robot was low enough to allow this scheme to work.

3.2. Robocup99

The experience earned with the '98 design lead us to improve some aspects for the '99 version. It was described in (Costa99) but the final version had some differences.

The working concept of the complete system was kept close to the previous year as seen in Figure 2.

The team, for the 99 competition, was fully redesigned on the mechanical aspect (Figure 5). We used a perfectly enclosed aluminium chassis. While it resulted in a very compact and solid robot, it was latter found to be a disadvantage because, aside from unnecessary space restrictions, when an electronic intervention was required, a lot of time was consumed opening the robot and lodging the two electronics boards.

New stepper motors were used in order to get a very high maximum velocity, easily going above 50 *cm/s*.

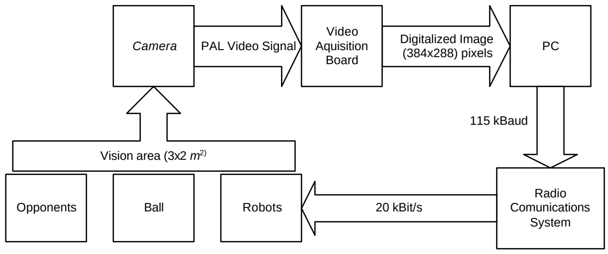

Figure 2 – RoboCup99 block diagram.

These two-phase stepper motors were driven by an integrated dual FET H-bridge. We generated the control signals using an Atmel AVR 90S4414. This microprocessor could generate the four PWM waves with the requested frequency and amplitude. When a new speed was requested it would be reached using a pre-programmed acceleration profile. With that capability we could ensure not to loose steps and avoid the need for a local feedback loop. These motors consumed significantly more than last year's motors. We had to use twelve *Ni-Cd* batteries with a rated capacity of 900 *mAh*. As we used only one source of power, it was necessary some care to avoid the problems associated with the motor generated noise.

The radio communications system was built around another AVR AT90S2313 that is exclusively dedicated to that task. With that extra processing power we resorted to a new improved communications scheme. This new modulation scheme improved the reliability of the radio link. The carrier frequency was chosen using a hardware switch avoiding the task of switching the radio modules. The radio-received commands could, then, be relayed to the stepper controller micro. Again, as we chose not to implement any kind of local sensors, no local decisions could be made. Naturally, we kept the radio link one way only meaning that only the coach could communicate with the robots.

The vision system was improved in order to process the full 50 *Hz* allowed by the PAL standard. Each frame was processed as it was arriving. In the end of a received frame the decision making was performed. That did not take too much time on the PC we were using, a Pentium III 450 *MHz*. The visual bar codes of the robots were maintained in a slightly different combination. We were surprised to see that many other teams had also adopted this solution which was introduced by us on the previous year.

The colour mappings could be optimised from game to game, or even during a pause in the game. Our software allowed us to view, in real time, the effects of any colours calibration. This is important as lighting conditions change over the field and also it allowed for changes on the opposing teams colours, without any trouble. This calibration could be performed using example areas marking what pixels are acceptable and others that are definitely not interesting. These colour issues were addressed using a three dimensional histogram of colours, as presented in Figure 6.

In '99 the rule stating the maximum camera height changed. This was due to the playing conditions, as shown in Figure 4. In order to keep up with that rule change, a more sophisticated way to deal with the camera distortions was needed. Those effects were much more severe and had to be dealt with a non-linear mapping of the field as shown in Figure 7. The mapping could be estimated from a set of points supplied by the user. The result was presented interactively and could be used to improve the mapping.

As the year before, the whole system was programmed in C/C++ under DOS. In that way we could use the full extent of the PC computational power and also gain as much real time control as possible.

Controlling the '99 robots was easier because of the fact that as stepper motors were used, no local control loop was required. Likewise, the model and parameter estimation was much simpler.

The control system implemented was also based on a Kalman Filter that held the robots and the ball state. Team behaviour was enhanced and the robots were given more mobility in order to take advantage of the new increased peak speed.

4. Team Strategy

Team strategy has been kept quite simple in order to provide robustness and understandability. The presented issues are common to the '98 and '99 teams.

Strategy is based on the fact that each player has a simple individual role.

The first role is the goal keeper and the robot assigned this role does not change during the gameplay. This player's position is always along the goal area. If the ball is moving fast towards our goal, the location of the goal keeper is changed in order to avoid the goal. Lateral position is found by calculating the interception of the goal line with the current direction of the ball.

There are two defense players that are positioned in parallel to the goal line. In order to minimise the possibility of the ball reaching beyond their defensive line, calculations are made using ball direction vector and goal keeper position to make a tight defensive block of all the defenders: to two defence players and the goal keeper.

Two attack players are typically placed on opposing sides of the offensive midfield. The one nearer the ball charges it and pushes it to a random zone of the opponent goal line where the opponent keeper is not placed.

In certain well specified situations robots change roles. One typical situation this occurs is when the ball sits in front of one of the defence robots. In this situation, the robot closer to the ball changes to attacker and another attacker changes to defence.

This kind of arrangement where each role is individually programmed gives quite a lot of freedom still contributing to the understandability of the overall system. Of course, role changing is of the utmost importance by providing the flexibility and team co-operative performance.

5. Conclusions and Future Trends

Mobile Robotic competitions are a very motivating way of fostering research, development and education. They also provide a way to interchange knowledge and allow for a standardised benchmark for the overall system performance.

Many different issues must be dealt with and problems ranging from mechanical, electronics, control and computer science must researched. Treatment of these interdependencies demand for a well structured and modular design. The presented design tried to keep up with these issues without forgetting classical concerns like reliability, maintainability and pricing.

The presented paper describes the main design issues and adopted solutions used by our 5dpo team for the small league (F180) robotic soccer over a period of two years. Of course, we are working an even better system for the next year.

One of our perceived major handicaps was the absence of a kicking device. Both teams that reached the final had efficient kicking devices. Also, a new kind of team strategies will be made available by the addition of this capability.

Moving toward the elimination of the wall surrounding the field, it was decided to make them sloped. These 45º walls change the behaviour of an incoming ball and the decision and control systems must be improved to deal with this new situation.

The outlined strategy is very simple but powerful. It is somewhat defensive but provides for many counter-attacks that are both impressive and effective.

6. Related Web Links

http://www.robocup.org
The official Webpage for the robocup Federation.

http://www.delorie.com/djgpp/
Home of the GNU C compiler port for MS-DOS. This compiler is open-source and free.

http://www.talula.demon.co.uk/allegro/
The Graphical Library, also free and open source.

http://www.fe.up.pt/~robosoc
Our home page.

7. References

(Costa98) Paulo Costa, António Moreira, Armando Sousa, Paulo Marques, Pedro Costa, "5dpo Team Description", *Robocup 1998 - Proceedings of the second RoboCup Workshop*, Paris, France, July 2, 1998, pp 511-517.

(Costa99) Paulo Costa, António Moreira, Armando Sousa, Paulo Marques, Pedro Costa, Aníbal Matos, "5dpo Team Description", *Robocup 1999 - Robot World Cup Soccer Games and Conference*, Stockholm, Sweden, pp 85-89, 1999.

(D'Andrea99) Raffaello D'Andrea, Jin-Woo Lee, "Description of Cornell BigRed Small League Robocup Team" - *Robot World Cup Soccer Games and Conference*, Stockolm, Sweeden, pp 14-23.

(Rowstron98) A Rowstron, B. Bradshaw, D. Crosby, T. Edmonds, S. Hodges, A. Hopper, S. Lloyd, J. Wang, S. Wray, "CURF: Cambridge University Robot Football Team", *Robocup 1998 - Proceedings of the second RoboCup Workshop*, Paris, France, July 2, 1998, pp 503-509.

(Veloso99) Manuela Veloso, Michael Bowling, Sorin Achim, "The CMUnited - 99 Small Robot Team" - *Robot World Cup Soccer Games and Conference*, Stockholm, Sweden, pp 24-32.

(Wyeth99) Gordon Wyeth, Brett Browning, Ashley Tews "UQ Robroos: On-going Design of a Robot Soccer Team" - *Robot World Cup Soccer Games and Conference*, Stockholm, Sweden, pp 77-80.

Figure 3 – RoboCup98 Team

Figure 4 – RoboCup99 league fields.

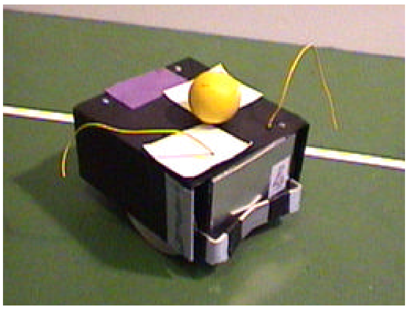

Figure 5 – RoboCup99 Robot.

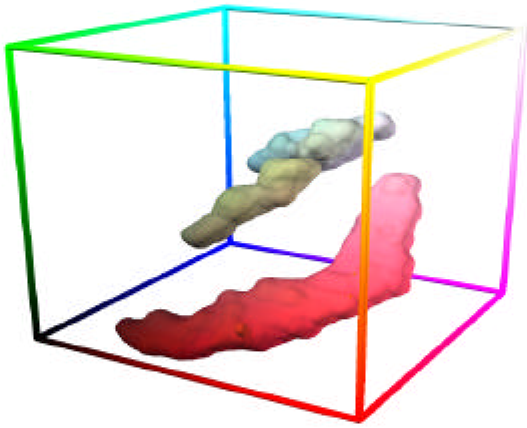

Figure 6 – Three dimensional colour histogram. Top left corner is Green; bottom far corner is Blue; bottom near is Red

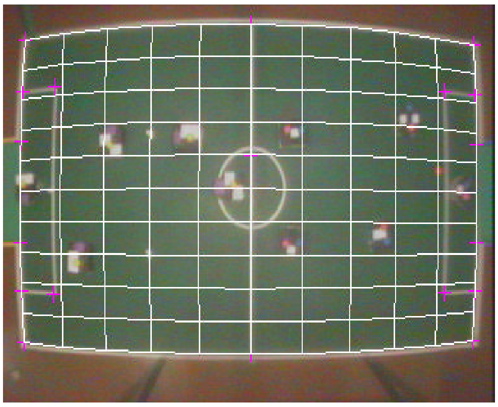

Figure 7 – Non-linear mapping of field co-ordinates