Roles, Positionings and Set Plays to Coordinate a RobotCup MSL Team

Nuno Lau[1], Luís Seabra Lopes[1], Nelson Filipe[1], Gustavo Corrente[1]

[1] Transverse Activity on Intelligent Robotics
IEETA / Universidade de Aveiro, 3810-193 Aveiro, Portugal
{nunolau, lsl, nelson.filipe, gustavo}@ua.pt

Abstract. This paper presents the team coordination methodologies of CAMBADA, a robotic soccer team designed to participate in the RoboCup middle-size league (MSL). The coordination model extends and adapts previous work in the Soccer Simulation League to the MSL environment. The approach is based on flexible positionings and priority-based dynamic role/positioning assignment. In addition, coordinated procedures for passing and setplays have been implemented. With the described design, CAMBADA reached the 1st place in the RoboCup'2008 world championship, becoming the first Portuguese real robot team to win in RoboCup. Competition results and performance measures computed from logs and videos of real competition games are presented and discussed.

Keywords: Multi-robot team coordination, strategic positioning, dynamic role assignment, coordinated procedures

1 Introduction

As robots pervade different areas of human activity, researchers are naturally prompted to investigate how robots can cooperate in order to perform complex tasks. Moreover, progress in wireless communication technologies enables information sharing and explicit coordination between robots. These are basic capabilities needed to support sophisticated cooperation and coordination algorithms. Given this increasing availability of robots and communication technologies, multi-robot systems have, in the last two decades, been receiving increasingly attention from researchers [2][6][25].

Multi-robot systems also present advantages with respect to single robots. First, some tasks are difficult or even impossible to be carried out by a single robot. In other cases, by providing a larger work force, multi-robot systems can carry out tasks faster. Multi-robot systems also facilitate scalability, as larger problems can often be solved by adding more robots to the team. Finally, through their inherent redundancy, multi-robot systems offer robustness, as they may still work when a team member is damaged or malfunctioning.

The development of multi-robot systems raises many new research issues concerned with how robots can coordinate their actions to carry out the assigned tasks as efficiently as possible. Among other issues, the following can be mentioned: How are different sub-tasks assigned to different robots [10]? How can different roles be assigned to different robots [19] [21][25]? If robots need to move in formation, how can it be controlled [5]? How can multi-robot plans be generated and/or executed [1]? Which information should robots exchange to enable coordination [13]? How can multi-robot systems be debugged [9]?

The authors have been addressing several of these issues in the robotic soccer domain. In particular, the authors contributed to the development of CAMBADA, a RoboCup middle-size league (MSL) team (Fig. 1). The MSL is one of the most challenging leagues in RoboCup. Robotic players must be completely autonomous and must play in a field of 12m × 18m [16]. Teams are composed of at most six robots with a maximum height of 80 cm. Human interference is allowed only for removing malfunctioning robots and re-entering robots in the game.

Fig. 1 CAMBADA robotic team

Building a team for the MSL is a very challenging task. Robots must be robust, fast and possess a comprehensive set of sensors. At the software level they must have an efficient set of low-level skills and must coordinate themselves to act as a team. Research conducted within CAMBADA has led to developments concerning hardware [4], computational and communications infrastructure [20], vision system [7], monitoring / debugging [9] and high-level deliberation and coordination [13]. This paper focuses on the last aspect, providing a detailed and up-to-date account of the currently used algorithms and their performance.

The complexity of MSL explains why most teams have implemented relatively simple coordination capabilities. The more advanced teams achieve coordination through the assignment of different roles to the robots [3][19][27]. Typically there is, at least, an attacker, a defender, a supporter and a goalie. As perception and sensorimotor capabilities become more sophisticated it will be possible to develop more sophisticated coordination algorithms. This trend is pushed further by the increase in team size (number of robots) as well as field size. A natural source of inspiration is the RoboCup Soccer Simulation League, where teams have been using coordination layers with strategy, tactics and formations [22][25], coordination graphs [12] and reinforcement learning [23][16].

CAMBADA participated in several national and international competitions, including RoboCup world championships (5th place in 2007, 1st place in 2008, 3rd place in 2009) and the Portuguese Open Robotics Festival (3rd place in 2006, 1st place in 2007, 2008 and 2009). This paper presents the coordination methodologies developed for RoboCup'2008. These methodologies largely explain the great success achieved, as the CAMBADA robots are among the slowest of the league.

This paper is organized as follows: Section 2 presents the hardware and software architectures of CAMBADA players. Sections 3 and 4 describe the adopted coordination methodologies. Section 5 presents and discusses competition results and various performance measures. Section 6 concludes the paper.

2 Hardware and Software Architecture

CAMBADA robots (Fig. 1) were designed and completely built at the University of Aveiro. Each robot fits into a cylindrical envelope with 485 mm in diameter. The mechanical structure of the players is layered and modular. The lower layer contains motors, wheels, batteries and an electromechanical kicker. The middle layer contains the control electronics. The upper layer contains a laptop computer, a catadioptric omnidirectional vision system, a frontal vision system and an electronic compass.

The robots use three Swedish wheels for holonomic motion, and move at a maximum speed of 2.0 m/s. This is less than many other MSL teams, which can currently move at speeds typically between 2.5 and 4.0 m/s (e.g. [18] [11] [24] [8]). Robots also carry encoders, battery status sensors and an IR ball presence sensor.

The computational system in each robot is a set of processing nodes (small micro-controllers plus a laptop) connected through a CAN bus. Communication within the team is based on standard wireless LAN protocol IEEE 802.11. The team receives referee instructions using a wired LAN TCP link. On the laptop, CAMBADA players run several software processes, for image acquisition and analysis, information integration, deliberation and communication with the low-level modules. The order and activation schedule of the processes is performed by a process manager [20].

The top-level processing loop starts by integrating perception information gathered locally from the vision processes, odometry, compass and ball presence sensors. All this information is stored in a shared data structure called Real-Time Data Base (RTDB). The RTDB has a local area, shared only among local processes, and a global area, shared with the other players. The global area is transparently replicated in all players in real-time. Self-localization uses a sensor fusion engine based on the publicly available engine described in [14]. Compass information is used to resolve ambiguities and detect self-localization errors. The final fusion step is to integrate local information with information shared by teammates.

Deliberation in CAMBADA considerably relies on the concepts of *role* and *behavior*. During open play, the CAMBADA agents use only four roles: *RoleGoalie*, *RoleSupporter* and *RoleStriker*. Further details about the developed roles and respective coordination mechanisms will be presented in sections 3 and 4.

Roles select the active behavior at each time step. Behaviors are the basic sensorimotor skills of the robot, like moving to a specific position or kicking the ball. The set of behaviors that are implemented in the CAMBADA agent are adapted to its catadioptric omnidirectional vision and holonomic driving systems. In brief, the current set of behaviors is the following:

- ***bMove*** uses two symbolic parameters: the target position where to move; and the position which the player should be facing in its path to the target. This behavior may avoid obstacles and avoid the ball (used during the game repositions).
- ***bMoveToAbs*** allows the movement of the player to an absolute position in the game field, and also allows the player to face any given position.
- ***bPassiveInter*** moves the player to the closest point in the ball trajectory.
- ***bDribble*** is used to dribble the ball to a given relative player direction.
- ***bCatchBall*** is used to receive a pass. The player aligns itself with the ball path and, when the ball is close, moves backwards to soften the impact.
- ***bKick*** is used to kick the ball accurately to one 3D position, either for shooting to goal or passing to a teammate. Polynomial functions, whose coefficients were determined by experimentation, are used to compute kick power.
- ***bGoalieDefend*** is the main behavior of the goalie.

3 Positionings and Roles

For open play, CAMBADA uses an implicit coordination model based on notions like *strategic positioning*, *role* and *formation.* These notions have been introduced and/or extensively explored in the RoboCup Soccer Simulation League [25][21]. In order to apply such notions in the MSL, some new algorithms had to be designed. The approach is presented in detail in this section.

3.1 Formations and strategic positionings

Formations are sets of positionings, where each *positioning* is a movement model for a player. The assignment of players to specific positionings is dynamic. Each positioning is specified by three elements:

- Home position, i.e. the target position when the ball is at the centre of the field
- Region of the field where the player can move, and
- Ball attraction parameters, used to compute the *target position* of the player in each moment based on the current ball position

Using different home positions and attraction parameters for the positionings allows a simple definition of defensive and attack strategic movement models. Fig. 2 shows the formation of the team used in RoboCup'2008 for several ball positions.

3.2 Roles in open play

CAMBADA robots use three roles in open play: *RoleGoalie*, activated for the goalkeeper, *RoleSupporter* and *RoleStriker*. *RoleStriker* is an “active player” role. It tries to catch the ball and score goals.

Only one player at a time is supposed to run RoleStriker. The striker is helped by teammates that take on *RoleSupporter* [13]. Supporters keep their target positions as determined by their positioning assignments and the current ball position. As a result, supporters accompany the striker as it plays along the field, without interfering. When the striker cannot progress with the ball towards the opponent goal and the ball remains behind the striker for more than some fixed time (e.g. 2 sec), supporters can take a more active behavior. In this case, the closest supporter to the ball also approaches the ball, acting as “backup striker”.

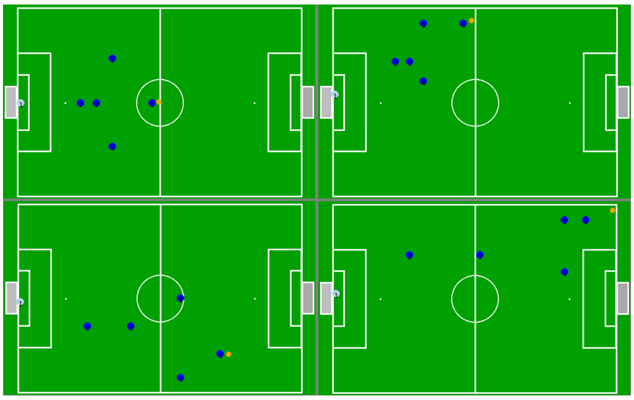

Fig. 2. Target player positions for several different ball positions

```
Algorithm: role and positioning assignment
Input: POS - array of N positioning, BallPos - ball position
Input/output: PL - array of K active players (K =< N)
Local: TP - array of N target positions
{
  clearAssigments(PL);
  TP = calcTargetPositions(POS,BallPos);
  for each POS[i], i ∈ 1..N, in descending order of priority
  {
    if there is no free player then return;
    p = the free player closest to TP[i];
    PL[p].positioning = i;
    PL[p].targetPosition = TP[i];
    if POS[i] has highest priority then PL[p].role= striker;
      else PL[p].role = supporter;
  }
}
```

Fig. 3. CAMBADA Positioning and role assignment algorithm

3.3 Role and positioning assignment

The play-on decision that assigns roles and positionings to the active players is performed using a new algorithm that takes into account different priorities for the different roles and positionings, so that the most important ones are always covered. This is an important feature since the number of available players varies as a result of several common situations in the MSL, namely hardware and software malfunctions and referee orders (see section 5.1 for concrete measures).

The role assignment algorithm is presented in Fig. 3. Consider a formation with N positionings and a team of $K \leq N$ available field players (goal-keeper is not counted).

Firstly, the distances of robots to each of the target positions are calculated. Then the striker role is assigned to the robot that is closest to the highest priority strategic positioning, which is in turn the closest to the ball. From the remaining *K*-1 robots, the algorithm proceeds assigning the positionings, in priority order, to the closest unassigned robot to the associated target position. This algorithm may be performed by the coach agent in the base station, ensuring a coordinated assignment result, or locally by each robot, which may lead to some inconsistencies. In the latest competitions, positioning assignments were carried out by the coach at intervals of 1 second and the role assignments were individually carried out by each player.

4 Coordinated Procedures

Coordinated procedures are short plans executed by at least two robots. In some cases, these plans involve communication resulting in explicit coordination. In the case of CAMBADA, coordinated procedures are used for passes and set plays.

4.1 Passes

Passing is a coordinated behavior involving two players, in which one kicks the ball towards the other. Until now, MSL teams have shown limited success in implementing and demonstrating passes. In RoboCup 2004, some teams had already implemented passes, but the functionality was not robust enough to actually be useful in games [15] [26]. The CoPS and Tribots teams also support pass play [28][16].

Two player roles have recently been developed for coordinated passes in the CAMBADA team. In the general case, the player running *RoleStriker* may decide to take on *RolePasser*, choosing the player to receive the ball. After being notified, the second player takes on *RoleReceiver*. These roles were not used in real competition games in 2008, but they have been demonstrated in RoboCup'2008 MSL *Free Technical Challenge*. A similar mechanism has been used for corner kicks (see below). In the free challenge, several passes occurred until a position to score a goal was reached. The sequence of actions on both players is described in Table 1. The coordination between passer and receiver is based on passing flags, which can take the following values: READY, TRYING_TO_PASS and BALL_PASSED.

4.2 Set plays

CAMBADA also uses coordinated procedures for set plays, in situations such as kick-off, throw-in, corner kick, free kick and goal kick. Set play procedures define a sequence of behaviors for several robots in a coordinated way. For that purpose, the involved players take on specific roles. This role-based implementation of set plays not only was easy to integrate with the previous agent architecture, but also facilitated the test and tune of different plans allowing for very efficient final implementations.

Table 1. Coordinated action in a pass.

RolePasser	RoleReceiver
PassFlag ← TRYING_TO_PASS	
Align to receiver	Align to Passer PassFlag ← READY
Kick the ball	
PassFlag ← BALL_PASSED	
Move to next position	Catch ball

RoleToucher and *RoleReplacer* are used to overcome the MSL indirect rule in the case of indirect set pieces against the opponent. The purpose of *RoleToucher* is to touch the ball and leave it to the *RoleReplacer* player. This scheme allows the replacer to score a direct goal if the opportunity arises. Two toucher-replacer procedures were implemented. In the case of *corner kicks*, the toucher passes the ball to the replacer, which catches it and shoots to the goal (pseudo-code in Fig. 4). The passing algorithm is as explained above. Another toucher-replacer procedure is used for *throw-in*, *goal kick* and *free kick* situations. Here, the toucher touches the ball pushing it towards the replacer until the ball is engaged by the replacer, then withdraws leaving the ball to the replacer. The replacer also moves towards the ball, waits that the toucher moves away and then shoots to the opponent goal. Both toucher and replacer position themselves on the shoot line, so that, as soon as the toucher moves away, the replacer is ready to shoot. For the kick-off, the procedure is the same, except that robots must be in their own side of the field.

```
Algorithm: RoleReplacer // for corner kicks
{
   if I have Ball then shoot to opponent goal
   else if Ball close to me then move to Ball
     else if Toucher already passed ball then catch Ball
       else wait that Ball is passed
}
```

Fig. 4. Replacer role algorithm for corner kicks

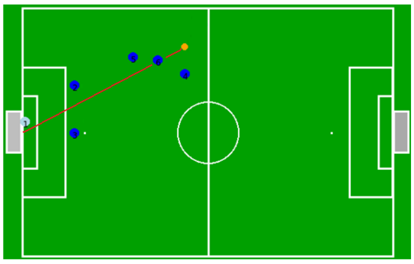

Fig. 5. Placement of RoleBarrier players

Finally, in the case of set pieces against CAMBADA, *RoleBarrier* is used to protect the goal from a direct shoot. The line connecting the ball to the own goal defines the barrier positions. One player places itself on this line, as close to the ball as it is allowed. Two players place themselves near the penalty area. One player is placed near the ball, 45º degrees from the mentioned line, so that it can observe the ball coming into play. The last player position prevents progression of the ball through the closest side of the field. The placement of players is illustrated in Fig. 5.

The assignment of the *RoleBarrier*, *RoleReplacer* and *RoleToucher* roles is executed by sorting the agents according to their perceived distances to the ball and selecting the closest ones, up to the maximum number of agents in each role. When selecting an exclusive role (ex: *RoleReplacer*) the agent looks at the other teammates role decisions and if it finds the same role in a player with a lower *id* it will not select that role. This assignment is always performed locally by each robot.

5 Performance Evaluation

The CAMBADA team participated and won the MSL world championship in RoboCup'2008 (Suzhou, China, July 2008). Most performance evaluation results presented in this section were obtained by analyzing log files and videos of games in this championship. RoboCup'2008 competition results will also be presented. The logs are created by the coach agent. At 1 second intervals, the coach takes a snapshot of relevant information retrieved from each robot, including current role, strategic positioning, behaviour, self position and ball position. A software tool was developed to analyse game logs and extract relevant evaluation measures. As the CAMBADA team made it to the final, it was scheduled to play 13 games. One of them was not played due to absence of the opponent. For two other games, the log files were lost. Thus, the results presented below are extracted from log files of the remaining 10 games.

5.1 General game features

Three main classes of game states are *open play*, *set piece against* CAMBADA and *set piece for* CAMBADA. Table 2 shows the respective time distribution in percentage of full game duration, computed over the 10 game logs mentioned above. The time spent in set pieces, considerably higher than what might be expected, results from the dynamics in MSL games. In fact, robots fast moving capabilities (up to 4m/s) and powerful ball kicking capabilities are not accompanied by sufficiently effective ball control capabilities, thus causing various types of set pieces. The time spent in set pieces justifies the investment in the development of the Replacer & Toucher combination in CAMBADA. A high efficiency rate in set pieces makes a real difference in the final team performance.

Table 2. Time distribution for different classes of game states.

Game state	Time (%)
Open play	53.1
Set piece for	21.5
Set piece against	25.4

Another common feature in MSL teams is that, due to reliability issues, the number of playing field robots is often less than the maximum of five. Table 3 shows the average percentage of game time (in the 10 mentioned game logs) for different numbers of playing field robots in the CAMBADA team.

Table 3. Percentage of game time for different numbers of playing field robots.

	#running robots					
	0	1	2	3	4	5
Time (%)	0.3	4.5	3.5	16.1	39.3	36.3

The average number of running field robots for the CAMBADA team was 3.98. This reveals the reliability problems that were experienced mostly in the beginning of the championship. These were solved to some extent during the championship and reliability improved in later games. In the final game the average number of running field robots was 4.33.

Capabilities for shooting to goal, although not directly based on coordination methodologies, are essential for a team's success. Fig. 6b shows the location from where the ball was shot to goal in the RoboCup'2008 MSL final (CAMBADA-TechUnited). CAMBADA showed good scoring abilities in the competition. Table 4 shows the results of all the shots made in the final game within 9 meters of the opponent goal (for larger distances, a shot doesn't have enough power to pose a real threat to the opponent team). A total of 15 shots were made, of which 1 was missed, 1 hit the post and another hit the bar. The remaining 12 hit the intended target within the goal. This gives an accuracy rating of 80%. From all the 15 shots made, 7 resulted in a goal being scored. This gives a goal scoring success rate (within 9 meters) of 46.7%.

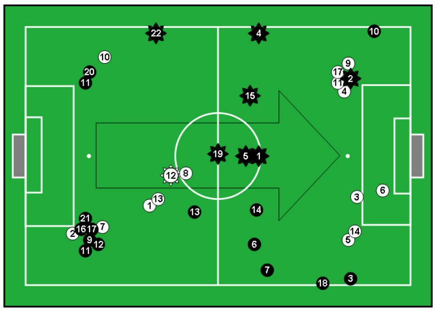

Fig. 6. Shoot locations in the final CAMBADA (black, on the left) - TechUnited (white, on the right) game in RoboCup 2008 (shoots are circles and goals are sun-like forms)

Table 4. Goal scoring performance

Result	Number
Missed	1
Post/bar	2
Defended	5
Goal	7
Total	15

This high success rate is the result of accurate ball placing when kicking. In 5 of the 7 scored goals, the goalkeeper was actually well positioned and in the path of the ball. However, the accurate calibration and power selection for each kick made the ball reach the opponent goal at an height slightly above 80 cm which effectively caused it to go over the goalkeeper, and creating a shot that is very difficult to defend.

5.2 Roles and behaviors

Table 5 shows the average percentage of time any given player spends in each role, with respect to the total time the player is active in each game. It can be observed that players spend a considerable amount of time (45.2 %) as *RoleSupporter*. This is to be expected since there may be up to 4 players with the Supporter role in open play, while there is at most one player acting as *RoleStriker*. This largely explains why the *RoleStriker* time is approximately 1/4 of the *RoleSupporter* time. The small deviation from the exact 1/4 relation is explained by two main factors: first, *RoleSupporter* is also taken by some players during set plays for CAMBADA; and, second, the number of field robots is often less than the maximum of five, as described above.

Table 5. Average time (± standard deviation) spent by players in different roles (in %).

Role	% Time
RoleStriker	10.4 ± 5.2
RoleSupporter	45.2 ± 10.0
RoleToucher	5.9 ± 4.1
RoleReplacer	5.6 ± 4.6
RoleBarrier	28.4 ± 6.5
RoleParking	4.4 ± 6.4

It can also be seen that more time is spent in set plays against CAMBADA (28.4%, since usually four players take the Barrier role in these situations) than in set plays against the opponent (11.5% in Toucher and Replacer roles). *RoleParking* moves robots outside of the field at the end of the first half and at the end of the game

A more in-depth perspective is given by Table 6, which shows the role time distribution across the three classes of game states. It can be seen that in open play basically only *RoleStriker* and *RoleSupporter* are used. In set pieces for CAMBADA, players take the roles of *RoleReplacer*, *RoleToucher* and *RoleSupporter*. In set pieces against CAMBADA, all field robots act is *RoleBarrier*. Underlying the numbers in Table 6 is the fact, already mentioned above, that CAMBADA had an average of nearly 4 field players. That explains why the time spent as supporter in open play is approximately 3 times that of striker, and the time spent as supporter in set pieces for CAMBADA is approximately 2 times that of toucher or replacer.

Table 7 shows the average percentage of time any given player spends running each implemented behavior. The second column of the table shows such percentages irrespective of the role taken. The third column shows the percentages of time considering only the periods in which players acting as *RoleStriker*.

Table 6. Average time spent by players in different roles (in %) for different game states

Role	Open play	Set piece for	Set piece against
RoleStriker	24.3	0.3	0.4
RoleSupporter	75.3	51.5	0.3
RoleToucher	0.4	23.7	0.0
RoleReplacer	0.0	24.5	0.0
RoleBarrier	0.0	0.0	99.3

Table 7. Average time (± standard deviation) spent by players running different behaviors

Behaviour	%time (any role)	%time (Striker)
bMove	4.9 ± 3.0	43.7 ± 4.4
bMoveToAbs	74.7 ± 12.6	25.3 ± 4.7
bDribble	1.4 ± 1.2	13.4 ± 4.5
bKick	1.8 ± 1.5	14.6 ± 7.7
bCatchBall	0.2 ± 0.3	

These values highlight the specificity of *RoleStriker*: much less time moving to absolute positions, since the striker most of the time ignores its strategic positioning assignment; much more time in moving (to the ball), dribbling and kicking.

5.3 Coordination

In the final game of RoboCup'2008 (CAMBADA-TechUnited), the ball was in the opponent's side 73% of time, mainly in the centre of the field towards the opponent's side. This results from the effectiveness of the CAMBADA's coordination approach.

Some measures of coordination performance have been extracted. According to the logs, players change roles 2.02±1.02 times per minute. As role assignment is distributed (implicit coordination), it occasionally happens that two players take on *RoleStriker* at the same time. On average, all inconsistencies in the assignment of the Striker role have a combined total duration of 20.9±27.4 seconds in a game (~30 minutes). The high standard deviation results mainly from one game in which, due to magnetic interference, localization errors were higher than normal. In that game, role inconsistencies occurred 45 times for a combined total of 101 seconds.

Concerning strategic positionings, relevant mainly to supporters, the average distance of the player to its target position is 1.38±0.48 m. The strategic positioning assignment for each player is changed on average 9.83±2.23 times per minute.

As the CAMBADA players do not track the positions and actions of the opponent players, it is not possible to compute an exact measure of ball possession. However, the game logs enable to compute related measures, as shown in Table 8. The closest player to the ball is at an average distance of 1.2m from the ball (the field is 18m × 12m). The ball is perceived by at least one robot of the CAMBADA team 91.7% of the time. The ball is engaged in a robot's grabber device 9.8% of the time.

Some additional analysis was carried out based on the log of the final game. Table 9 provides information on set pieces, identifying the total number of times each set piece was executed as well as the number of times it was correctly executed.

Table 8. Measures related to ball possession (average ± standard deviation)

Measure	Value
Average minimum distance to the ball (meters)	1.246 ± 0.325
Average time with ball visible (%)	91.7 ± 3.5
Average time with ball engaged (%)	9.8 ± 4.7

Table 9. Set-piece performance in the final game

Set piece	#Occurrences	#Correct
Kick-off	2	2
Free kick	1	1
Throw-in	6	5
Goal kick	10	8
Corner kick	2	2
Total	21	18

In 21 set pieces, 18 were correctly executed (85.7%). The failed throw-in occurred due to magnetic interference in one area of the field, causing the robot to mislocalise itself. The two missed goal kicks occurred because the robot acting as *RoleToucher* movement, while pushing the ball towards the Replacer, wasn't accurately aligned and did not succeed in delivering the ball to the Replacer. This can be due to some small localisation errors experienced near the goal kick marker.

Table 10 provides information on goal scoring success in set piece situations in which the set piece procedure was correctly executed and the distance to the opponent goal was less that 9 meters. In the 6 set pieces for CAMBADA carried out under these conditions, 4 resulted in a goal being scored. This is a very good success rate. It should be noted that from the 7 goals scored in this game, 4 resulted from set pieces. This shows the importance of having accurate, reliable and swift set pieces in MSL games. These high values were observed consistently throughout the whole championship. They were crucial in the team's success, proving to be a powerful asset for achieving victories.

Table 10. Set-piece performance

Set piece	Occurrences	Success
Kick-off	2	2
Free kick	1	0
Throw-in	3	2
Total	6	4

5.4 Competition results

Table 11 presents the competition results of CAMBADA in RoboCup'2008. The team won 11 out of 13 games, scoring a total of 73 goals and suffering only 11 goals.

Table 11. RoboCup 2008 competition results.

	#games	#goals scored	#goals suffered	#points
Round-robin 1	5	41	2	15
Round-robin 2	4	16	3	9
Round-robin 3	2	5	2	3
Semi-final	1	4	3	3
Final	1	7	1	3
Total	13	73	11	33

6 Conclusion

The paper presented and evaluated the coordination methodologies of the CAMBADA team, the current world champion in RoboCup'2008 MSL.

During open play, an implicit coordination approach, based on role assignment, formations and flexible positionings, is used. The positioning of the team adapts to the external conditions and has maintained a strong defense and a good backup to striker role during the competition. In particular, although robot malfunctions decrease the number of field players, the dynamic positioning/role assignment algorithm maintains a competitive formation. This can be seen, not only from the competition results, but also from the detailed analysis of game logs and videos presented in the paper. More importantly, these results were obtained despite the fact that CAMBADA robots clearly move at low speed (2m/s), when compared to most of the main competitors.

The development of predefined role-based set plays proved to be very efficient both during the development phase, and during their execution in games. More than half of the 73 scored goals are direct result of these set plays.

One of the most significant aspects of this work is the integration of these coordination methodologies in a complex multi-robot system and their validation in the challenging RoboCup MSL competition. This contrasts with other approaches which are validated in controlled robotic environments, if not in simulation.

Acknowledgments. The CAMBADA team was funded by the Portuguese Government – FCT,POSI/ROBO/43908/2002 (CAMBADA) and currently FCT, PTDC/EIA/70695/2006 (ACORD). We would also like to thank the rest of the CAMBADA team for an excellent work environment.

References

[1] Lesser, V., et al., "Evolution of the GPGP/TAEMS Domain-Independent Coordination Framework," *Autonomous Agents and Multi-Agent Systems*, 9(1), pp. 87-143, 2004.

[2] Noreils, F. R., "Toward a robot architecture integrating cooperation between mobile robots: Application to indoor environment," *International Journal of Robotics Research*, 12. pp. 79-98, 1993.

[3] Arbatzat, M., et al., "Creating a Robot Soccer Team from Scratch: the Brainstormers Tribots," *Proc. of Robocup 2003, Padua*, Italy, 2003.

[4] Azevedo, J.L., M.B. Cunha, L. Almeida, "Hierarchical Distributed Architectures for Autonomous Mobile Robots: a Case Study," Proc. *ETFA2007- 12th IEEE Conference on Emerging Technologies and Factory Automation*, Patras, Greece, 2007, pp. 973-980.

[5] Balch, T., and R.C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, 14(6), pp. 926–939, 1998.

[6] Balch, T., and L.E. Parker, *Robot Teams: From Diversity to Polymorphism*, Natick, Massachusetts, A K Peters Ltd, 2002.

[7] Cunha, B., J. Azevedo, N. Lau, L. Almeida, "Obtaining the Inverse Distance Map from a Non-SVP Hyperbolic Catadioptric Robotic Vision System," U. Visser, et al., editors, *RoboCup-2007: Robot Soccer World Cup XI*, LNAI, Springer Verlag, Berlin, 2008.

[8] *EtherCAT Robots win German Open*, Press Release, EtherCAT Technology Group, 8 May 2008, at http://ethercat.org/pdf/english/etg_032008.pdf

[9] Figueiredo, J., N. Lau, A. Pereira, "Multi-Agent Debugging and monitoring framework," *Proc. First IFAC Workshop on Multivehicle Systems (MVS'06)*, Brasil, October, 2006.

[10] Gerkey, B.P., and M.J. Mataric, "A formal analysis and taxonomy of task allocation in multi-robot systems", *International Journal of Robotics Research*, 23 (9), pp. 939-954, 2004.

[11] Hafner, R., S. Lange, M. Lauer, and M. Riedmiller, "Brainstormers Tribots Team Description, *RoboCup International Symposium 2008, CD Proc.*, Suzhou, China.

[12] Kok, J., M. Spaan. and N. Vlassis, "Non-communicative multi-robot coordination in dynamic environments," *Robotics and Autonomous Systems*, 50 (2-3), pp. 99-114, 2005.

[13] Lau, N., L. Seabra Lopes, G. Corrente, "CAMBADA: Information Sharing and Team Coordination," *Autonomous Robot Systems and Competitions: Proc. of the Eigth Conference. 2 April 2008, Aveiro*, Portugal, Universidade de Aveiro, 2008, p. 27-32.

[14] Lauer, M., S. Lange and M. Riedmiller, "Calculating the perfect match: An efficient and accurate approach for robot self-localisation," *RoboCup 2005: Robot Soccer World Cup IX*, A. Bredenfeld, et al., eds., LNCS 4020, Springer, 2006.

[15] Lima, P, et al. (2005) RoboCup 2004 Competitions and Symposium: A Small Kick for Robots, a Giant Score for Science, *AI-Magazine*, 6 (2), 2005, p. 36-61.

[16] Muller, H. et al., "Making a robot learn to play soccer using reward and punishment", *KI 2007: Advances in Artificial Intelligence*, LNCS 4667, 2007, Springer, p. 220-234

[17] MSL Technical Committee 1997-2008, *Middle Size Robot League Rules and Regulations for 2008. Draft Version - 12.2 20071109*, November 9, 2007.

[18] Oubbati, M., M. Schanz, T. Buchheim, P. Levi, "Velocity Control of an Omnidirectional RoboCup Player with Recurrent Neural Networks," A. Bredenfeld, et al., Eds., *RoboCup 2005: Robot Soccer World Cup IX*, LNCS, vol. 4020, 2006, 691-701.

[19] Pagello, E., A. D'Angelo, E. Menegatti, "Cooperation Issues and Distributed Sensing for Multirobot Systems," *Proc. of the IEEE*, 94 (7), pp. 1370-1383, 2006.

[20] Pedreiras, P., L. Almeida, "Task Management for Soft Real-Time Applications Based on General Purpose Operating Systems," *Robotic Soccer*, Pedro Lima (ed.), Itech Education and Publishing, Vienna, Austria, 2007, pp. 598-607.

[21] Reis, L.P., et al., "Situation Based Strategic Positioning for Coordinating a Team of Homogeneous Agents," *Balancing Reactivity and Social Deliberation in Multiagent Sytems: From RoboCup to Real Word Applications*, M. Hannenbauer, et al. eds., LNAI 2103, Springer-Verlag, 2001, pp. 175-197.

[22] Reis, L.P., and N. Lau, "FC Portugal Team Description: RoboCup 2000 Simulation League Champion," *RoboCup-2000: Robot Soccer World Cup IV*, P. Stone, et al., eds., LNCS 2019, Springer, 2001, pp. 29-40.

[23] Riedmiller, M., and T. Gabel, "On Experiences in a Complex and Competitive Gaming Domain: Reinforcement Learning Meets RoboCup," *Proc. of the 3rd IEEE Symposium on Computational Intelligence and Games (CIG 2007)*. IEEE Press, 2007, pp. 17-23.

[24] Sato, Y., S. Yamaguchi, et al., "Hibikino-Musashi Team Description Paper," *RoboCup Int. Symposium 2008*, CD Proc., Suzhou, China.

[25] Stone, P., and M. Veloso, "Task Decomposition, Dynamic Role Assignment and Low Bandwidth Communication for Real Time Strategic Teamwork," *Artif. Intelligence*, 110 (2), 1999, pp. 241-273.

[26] van der Vecht, B., and P. Lima, "Formulation and Implementation of Relational Behaviours for Multi-robot Cooperative Systems," *RoboCup 2004: Robot Soccer World Cup VIII*, Springer LNCS Vol. 3276/2005, 2005, pp 516-523.

[27] Weigel, T. W., M. Auerbach, B. Dietl, et al., "CS Freiburg: Doing the Right Thing in a Group," *RoboCup 2000: Robot Soccer World Cup IV*, P. Stone, G. Kraetzschmar, T. Balch, eds., Springer-Verlag, 2001, pp. 52-63.

[28] Zweigle, O., R. Lafrenz, T. Buchheim, U.-P. Käppeler, H. Rajaie, F. Schreiber and P. Levi, "Cooperative agent behavior based on special interaction nets," *Intelligent Autonomous Systems 9*, T. Arai, et al., Eds. Amsterdam, The Netherlands: IOS Press, 2006.