The Cambridge University Robot Football Team Description

A. Rowstron[2*], B. Bradshaw[2], D. Crosby[2], T. Edmonds[2], S. Hodges[3], A. Hopper[2], S. Lloyd[2], J. Wang[2] and S. Wray[1]

[1] Cambridge University Computer Laboratory,
Pembroke Street, Cambridge, CB2 3QG, UK.

[2] Engineering Department, Cambridge University ,
Trumpington Street, Cambridge, CB2 1PZ, UK.

[3] Olivetti and Oracle Research Laboratory,
24a Trumpington Street, Cambridge, CB2 1AQ, UK.

Abstract. This paper describes the Cambridge University Robot Football entry and our experiences in the RoboCup'98 Small Robot League of the held in Paris competition.

In the competition we came top of our group, and fourth overall. We had the strongest group, with the team coming second in the group coming second overall in the end. This meant that we were able to play a number of good games, and we were able to evaluate our approach compared to others.

This paper presents an overview of our team, compares it to some of the other approaches used, and highlights the research issues of interest to us.

1 Introduction

The Cambridge University Robot Football Team is a system autonomous five player robot football team. The system uses global vision approach with multiple pitch side servers with multiple platforms linked to the servers by a robust radio communications system[1]. The camera is mounted approximately 4 meters above the centre of the table tennis table.

The pitch side servers perform image processing of the global vision feed and decide where the robots should move, using a novel extension of Potential Fields, called *Time Encoded Terrain Maps*. The image processing software is capable of processing the video stream at approximately 50 fields per second. The software which performs the cooperation planning between the physical agents is written in Java.

The team has a number of unique features; the ability to plan and perform passes between *any* players, and a goal keeper which is capable of capturing and kicking the ball. The cooperation planning software is able to , move another robot into position to receive a pass in advance of the first robot hitting the ball.

* Contact: aitr2@cl.cam.ac.uk

M. Asada and H. Kitano (Eds.): RoboCup-98, LNAI 1604, pp. 422-427, 1999.

In the RoboCup'98 the team came top of our group and 4th overall. Our overall system was one of the most robust present. None of our robots failed either before or during matches, the vision system was able to be reconfigured to work successfully from the opening match with all the pitches provided, and we were able to change communication frequencies as required, and our communications system was robust enough to be unaffected by local interference.

The next sections provide an overview of our system and contrast it to other approaches.

2 Vision System

The vision system analyses each frame to determine the location and speed of the ball and other players, and the location, orientation and speed of our players. The frame grabber produces 768x576 24-bit RGB frames at a rate of 50 fields per second. Once the initial locations of the objects being tracked have been found the vision system software is capable of processing at full speed (in other words 20 milliseconds per field). Only one other team appeared to be able to process as many fields.

Each frame is translated from RGB space to Hue-Saturation-Intensity (HSV) space, where classifying the marker colours on the mobile platforms and the ball is simpler. The ball and other marker colours are thresholded based on their location in the HSV space, colour being described by hue angle.

Due to the lighting conditions of the area where our test table was placed and the height of the camera above the table we designed our vision system to be able to compensate for any curvature of the table caused by the camera lens. During the calibration stage we allow for the light levels across the table to be calibrated for. Often at some points there were 'hot spots' where light is reflected. Our vision system was robust enough to cope with these whilst many teams were not able to. Also, if necessary due to lighting problems or colour confusion with the opposition robots we could quickly choose other marker colours and re-calibrate the vision system to use them.

Our team uses a team colour plus another colour for each player. This has the advantage that the vision system can uniquely identify each player. However, it requires the vision system to be calibrated to detect eight colours, which sometimes proved to be difficult in the lighting conditions provided in Paris. In the next generation system will use an alternative way to identify the robots and then use the vision system to just work out the exact position and orientation.

More details of the vision system can be found in the team description[2].

3 Cooperation Planning

The Cooperation Planner is responsible for the control and coordination of a team of the physical agents on the pitch. Its input is a stream of location data produced by the vision system and generates command sets for each physical agent. This involves analysis of the situation and prediction of the future game state, from which a team strategy is derived. The team strategy is then broken into individual roles which are assigned to the physical agents and the appropriate commands issued.

The information received from the vision system is a timestamped set of position and orientation coordinates for each object on the field for each frame processed. Relevant velocity information derived from subsequent frames is incorporated using a basic physical model to enable predictions of the play state in the short term (up to a second). The prediction accuracy beyond this is limited by the highly dynamic nature of the environment and its inherent unpredictability.

For the same reason, the cooperation planner is implemented using a stateless design. In this way, the state of the system is reflected only in the state of the physical agents and their environment. This improves the reactiveness of the system to any unexpected changes while a sophisticated system of tolerances promotes stability in the system.

3.1 Strategy Planning

The core of the Cooperation Planner is the Strategy Planner which uses the novel approach of encoding time in a terrain map, these *time encoded terrain maps* are used to represent the current state of play and from it derive a plan of action. In this case, the strategy is basically just path planning for the route of the ball.

The time encoded terrain map is formed from the environment state using information from the physical modeller for the time frame under consideration[2]. It provides a method for combining all of the known information about the given situation including agent positions and velocities into a single map. The map is distorted to represent the different velocities and the time distance travelled between various points. In this way, the map can be said to encode time.

The map in this case, is formed by representing each opposing agent as a hill and each of our own agents as a depression. This leads to a terrain map of a landscape where high ground may be considered as bad and low land may be considered as good. Superposition applies, so if two agents are nearby, their effects will combine. One alteration of the method is made at this point – once the map has been formed, points of low altitude (goodness) are brought back up to the sea-level norm (neutral) as when moving the ball, it is of little difference if it passes near to a player of the same team or through a clear area. The important point is to avoid the opposition players while also allowing for the fact that one of our players may negate the effects of an opposition player in the same area.

Tracing a path through this terrain which attempts to remain at the lowest cumulative altitude will produce a path that avoids opposition and favours locations near our own players. Note, that often with terrain map navigation, the

[2] This is usually advanced by a fraction of a second to allow for the latency through the system from data acquisition to command execution.

path is evaluated on the amount of ascent and descent required; however, in this case, the path is evaluated by the absolute altitude of the path. In this manner of operation, navigation through this terrain is very similar to artificial potential field navigation.

When the terrain map is being created the alterations made to it by a robot's presence is termed that robot's *influence*. The influence of a robot over a particular area may be defined as the likelihood that an area may be reached by the robot in a given time. Hence, a robot's influence is a measure of the possibility of a the robot being at some point. As the possibility of a robot being at a certain point depends on the amount of time it is given to get there, the map effectively encodes the foreseen dynamic variations of the field over time. In robot football, the key interest is the ball, and hence an agent's influence over an area can be interpreted as *the physical agent's influence on the ball should it get to that area.*

Once the map has been formed, finding the optimal path through it to the destination is a simple application of Dijkstra's least distance path finding algorithm for a connected network[3] (The A* pathfinding algorithm[4] is actually used for speed).

3.2 Strategy Action

Once the optimal path for the ball has been determined, the physical agents must be set in action to guide the ball along that path. This involves the assigning of different agents to different roles in the plan. The pathfinder is such that it tends to promote passing the ball to clear areas as well as passing directly to the agents. By doing so, it allows more flexible partitioning of roles and a plan more adept at finding gaps in defences.

Hence, the Cooperation Planner can use a general algorithm to both set up passes between team members and also provide effective blocking and interception behaviour. Additional specialised behaviours are invoked for certain boundary conditions such as when the ball is wedged against the side wall. In this case, the physical agent is directed to flick the ball back out into play using its tail. Additional specialised behaviours are incorporated to control the special abilities of the goalkeeper.

The large number of parameters which govern the operation of the pathfinder in terms of aggressiveness and timeliness allow for in depth tuning of the strategy which could in the future be adjusted automatically on the fly based on analysis of the game to date. Careful tuning of these parameters allowed the Cooperation Planner to achieve a high iteration rate while still maintaining effective performance.

The Cooperation Planner was able to plan passes between any players, and have a physical agent move to receive the ball *before* the first physical agent had hit the ball. Only one other team could perform passes, and this was limited to their attack robots. Although, it was difficult to evaluate the performance of the passing under playing conditions, on several occasions successful passes were observed.

It should also be noted that the ability to alter the characteristics of the system was important. When we arrived we had a system that avoided the oppositions physical agents. However, less than half the teams had this capability. This meant that those teams which avoided the opposition were at a disadvantage. Tuning the system to be more aggressive towards such teams meant that we were not disadvantaged by such behaviour.

4 Physical Agent

We used two different physical agents, a standard player[2] and a goal keeper. Our goal keeper was unique in that it could capture the ball and then kick it at high speed. This feature of the team led to extended periods of play during matches (over two minutes of action). More details of the goalkeeper and its impact can be found in Hodges[5].

The standard player was designed to be ‘thin’; this meant that the physical agents should have little computing power, and most processing should be done remotely on the servers at the side, in a similar strategy to the network computer. The platform received commands such as; move forwards x cms, or turn left x degrees.

We were surprised therefore to see that some of the teams had ‘thinner’ platforms, and purely sent commands of the form, left/right motor on/off. This is something we intend to investigate this coming year. However, these systems appeared to have to use the full bandwidth available to do this, and had no error detection or robustness in the communications systems. This meant they were often unable to change frequency channels, or cope with small amounts of background noise.

5 Communication System

To provide communications between the central server and the mobile platforms we use a wireless network, called Piconet[1]. Piconet is a low-rate, low-range, low-power system intended for use in the investigation of embedded networking. We created a system that allowed the available bandwidth to be focused on certain physical agents. The Cooperation planner told the communication systems which physical agents were important, and the commands to these robots were given much higher priority. Therefore, players not involved in the current action would sometimes appear to pause whilst the ones involved in the action would not.

Our communications systems was very robust, able to survive noise on the frequency we were using (the noise was eventually tracked down to the security guards mobile radio communication system). We were also able to change frequency with about 5 minutes warning.

6 Conclusions

Our system had several novel features, which allowed us to do well. Because of the nature of RoboCup we often disabled parts of our system to give other teams a fair advantage (we often disabled the goal keeper so other teams could track the ball). Our semi-final game against CMU'98 really showed how robust our system was. Firstly we had to change pitches and then recalibrate the system. We then had to change frequencies because CMU could not get their second frequency working, and then we played with only four physical agents because they could only get four platforms working. We then lost the game, however, we were able to show how a robust system could be made.

We will be entering the 1999 games, with a new system building on our experiences at Paris. We intend to push further the limits of sensors and sensing and to create faster, thinner platforms.

Acknowledgements

We were funded and general supported by Prof. Andy Hopper and the Olivetti and Oracle Research Laboratory. Thanks to Stuart Wray, Frank Stajano and others, who helped and advised us during the development of this Robot Team. We also thank Chris at Engineering for his help with platform manufacture.

References

1. F. Bennett, D. Clarke, A. Hopper, A. Jones, and D. Leask. Piconet - embedded mobile networking. *IEEE Personal Communications Magazine*, 4:8–15, October 1997.
2. A. Rowstron, B. Bradshaw, D. Crosby, T. Edmonds, S.Hodges, A. Hopper, S. Lloyd, J. Wang, and S. Wray. CURF: Cambridge University Robot Football Team. In Minoru Asada, editor, *Proceedings of the second RoboCup Competition*, pages 503–510, July 1998.
3. Robert Sedgewick. *Algorithms*, volume 2nd edition. Addison-Wesley, 1988.
4. Stuart C. Shapiro. *Encyclopedia of Artificial Intelligence*, volume 2nd edition. Wiley, New York, 1992.
5. S. Hodges, D. Crosby, A. Rowstron, B. Bradshaw, T.Edmonds, A. Hopper, S. Lloyd, J. Wang, and S. Wray. The challenge of building and integrating a sophisticated goalkeeper robot for the small-size RoboCup competition. Submitted to ??, 1998.