# Lazarus Team Description

Anthony Yuen

Dalhousie University, Department of Computer Science, Halifax, Canada
yuen@cs.dal.ca

**Abstract.** This paper describes the Lazarus team that participated in the RoboCup Simulation League in 2001. It is characterized by tight formation of players during play. The purpose is to provide data for training the coach in recognizing team formations using neural networks.

## 1 Introduction

Being able to recognize patterns and structures in highly dynamic environments is an essential requirement in such areas as speech recognition and computer vision. In general, a software system is repeatedly *trained* against a series of labelled data samples, a process called *supervised learning*. The software system typically uses an *artificial neural network* [3] as the backend to do the learning. Once the system is trained, it can then be used on unlabelled test cases.

In the domain of face recognition, a typical task involves taking a series of pictures of a group of people with different facial expressions and orientations, such as front, back, left, and right. The system is trained against some of these pictures. The goal is then to classify the remaining pictures, that is to say, which picture belongs to which person.

The RoboCup Soccer Simulation [6] presents a challenging and interesting problem for the *online-coach*: to recognize the opponent team's **formation**, if any. The term formation used here is loosely based on real soccer formations. For example, **4-3-3** denotes the formation of 4 *defenders*, 3 *mid-fielders*, and 3 *forwards*.

The problem is challenging due to the fact that the players are constantly moving and changing positions on the soccer field. In fact, during the game, the players closest to the ball tend to deviate from their positions. Moreover, some teams, such as FCPortugal [5], uses dynamic positioning during play. On the other extreme, some teams do not pay attention to team formation at all. To keep the problem more managable, we will restrict ourselves to teams that do keep formations and change formations only occasionally (e.g. half-time).

The Virtual Werder team [2] participated in the RoboCup 2000 competition and made use of an online coach agent for the purpose of changing team formations dynamically. Their coach used neural network to recognize the opposing team formation and then chose an appropriate counter-formation from a fixed pool of known formations. Our approach follows a similar path.

A. Birk, S. Coradeschi, and S. Tadokoro (Eds.): RoboCup 2001, LNAI 2377, pp. 522–525, 2002.

## 2 Team Development

Team Leader/Member: **Anthony Yuen**

- Masters Thesis Graduate Student
- Institution: Dalhousie University
- Department: Computer Science
- Country: Canada

## 3 Player Description

Lazarus is a soccer simulation team based on the publicly available FCPAgent [4], which in turn is based on the CMUnited99 [7] code. As a result, a lot of development time is saved for the basic skills such as ball dribbling and shooting. This team has been deliberatly written in such a way that the players keep a strict team formation throughout the game. Only the currently **active** players deviate from their predefined formation position. An active player is a player closest to the ball. If it currently has the ball, then it decides to either *dribble* or *pass* to a teammate. Otherwise, the player engages in *interception* to retrieve the ball.

In addition to a formation position, each player (excluding the goalie) is assigned a **role**. A role is one of: *Sweeper*, *Defender*, *Mid-Fielder*, *Captain*, *Forward*, and *Striker*. The meaning of each role is loosely based on real soccer roles. The role of a player determines several player attributes such as aggressiveness and is dependent on the current team formation and player position.

## 4 Coach Description

A **coach** agent is a special client program that can connect to the soccer server and oversee and ongoing soccer match. The coach receives noiseless information (e.g. coordinates, velocities, staminas... etc) about every object in the soccer simulation. Therefore, at any given moment, the coach has complete knowledge of the ball, its team players, as well as the opponent team players. However, the coach-to-player communication is restricted to when the ball is out of play using free-style message formats. If the coach uses the new standard coach language [1], communication is restricted to one *info*, *advice*, and *define* message every 30 seconds (300 simulation cycles). The Lazarus coach uses the free-style message format.

The coach is assigned the task of recognizing team formations. To tackle this problem, the coach uses neural networks and the backpropagation learning algorithm [3]. For the initial training, the prototypical **4-3-3** formation is used to train the network. Due to the computational intensive nature of the backpropagation algorithm, the training is done offline. During each simulation cycle, the coordinates of each player in the soccer field are recorded in a format suitable for the neural network. The logged data is then used to train the network to classify the formation as 4-3-3 consistently. This process is repeated for several other common formations.

## 5 Research

By observing the top-level teams, it is clear that they employ some kind of team formation system, usually conventional and sometimes proprietary. Moreover, formations do not change from cycle to cycle – it would be rather chaotic otherwise. Instead, it is common for teams to switch from an offensive formation to a more defensive one at half time when it is leading the match.

The Virtual Werder team [2] demonstrated that by using an online coach to switch team formation, their team performed better than without a coach. Their coach uses a *single* neural network to classify approximately fifteen different formation systems, including the Catenaccio. In the same vein, the Lazarus coach also advises players to switch to more appropriate *counter-formations* during play. However, a **layered** approach is used in the learning and classification systems.

In the layered approach, one neural network is used to recognize one particular formation system. For example, the 4-3-3 neural network outputs a *normalized* confidence value to indicate how closely the inputs correspond to the 4-3-3 formation. Similarly, the 4-4-2 network recognizes the 4-4-2 formation. Separate networks are trained in similar ways. The end result is a collection networks for different formations.

To utilize these networks, a larger, layered neural network is formed. The inputs are as before. The second layer consists of the trained networks, now used as *nodes*. A third layer of *hidden* nodes is added. The final layer is the output layer that indicates which formation system is recognized based in the inputs.

After this phase is accomplished, the online coach can then perform team formation classification in *real-time*, during a soccer match. This is possible because the execution of a learned neural network is very fast. How this information is used is implementation-dependent.

## 6 Conclusion

Traditional pattern recognition domains deal with static data samples such as facial images. To make the task more challenging, we can consider the problem of recognizing faces *in motion*. Similarly, we can compare the task of analyzing team formation from a particular **snapshot** of a RoboCup simulation versus that of recognizing team formation with the players *in motion*.

A human can normally recognize a team formation *visually*. A computer program, on the other hand, cannot **see** what is happening in the soccer field. It needs to rely on some advanced machine learning techniques in order to perform this task. It is hoped that the layered neural network approach is a step in the right direction.

## 7 Future Work

It is unclear whether the layered network approach to recognizing team formations is better compared to the single network approach. More experiments are needed to provide a conclusive result. In addition, the usefulness of an online coach in RoboCup Soccer Simulation is yet to be determined. The new coach competition in RoboCup 2001 should provide some insights to this question. This research will become part of a Masters Thesis by the author of this paper.

## References

1. M. Chen, E. Foroughi et al, *RoboCup Soccer Server Users Manual*, http://www.cs.cmu.edu/~galk/rc01/manual.ps.gz, 2001
2. C. Drucker et al, *"Recognizing Formations in Opponent Teams"*, http://www.virtualwerder.de, 2000
3. J. Hertz, A. Krogh, and R.G. Palmer, *"Introduction to the Theory of Neural Computation, Volume 1"*, Addison-Wesley Publishing Company, Redwood City, California, USA, 1991
4. R. Luis, N. Lau, and L. Lopes, *FCPAgent Source Code*, http://www.ieeta.pt/robocup/documents/FCPAgent.tgz, 2000
5. R. Luis, N. Lau, *"FCPortugal Team Description: RoboCup2000 Simulation League Champion"*, in *RoboCup-2000: Robot Soccer World Cup IV*, Springer Verlag, Berlin, 2001
6. I. Noda et al, *"Soccer Server: A Tool for Research on Multiagent Systems"*, in *Applied Artificial Intelligence*, vol 12, pp. 233-250, 1998
7. P. Stone, M. Veloso, and P. Riley. *"The CMUnited-99 Champion Simulator Team"*, in *RoboCup-99: Robot Soccer World Cup III*, Springer Verlag, Berlin, 2000