Evolving Connectionist System Based Role Allocation for Robotic Soccer

L. Huang; Q. Song and N. Kasabov
Institute of Technology and Engineering, Massey University, P. O. Box 756, Wellington, New Zealand
Knowledge Engineering and Discovery Research Institute, Auckland University of Technology,
Private Bag 92006, Auckland 1020, New Zealand
l.huang@massey.ac.nz

Abstract: *Robotic soccer is an intelligent system where a group of mobile robots are controlled to perform soccer play (http://www.fira.net). The allocation of a suitable role for each robot in a team is a key for the success of the play. The paper treats this issue as one of pattern classification, and solves it with an Evolving classification function (ECF), a special evolving connectionist system (ECOS). A robot's role is determined by and evolves with the states of system (robots and target) in real time. The software and hardware platforms are set up for data collection and learning. The effectiveness of the proposed approach is verified by the experimental studies.*
Keywords: *Robotic Soccer, Mobile Robots, Evolving Connectionist Systems*

1. Introduction

In a robotic soccer system, a robot can be assigned one of the basic roles: *attacker*, *defender* and *goal keeper*, or the additional roles such as (active or strategic) *support* (Weigel, T. et al, 2002). The *attacker* tracks the target (ball) and tries to put it into the opponent goal area. The *defender* blocks the opponents and supports the goal keeping actions. The *goal keeper* clears the ball from its home goal area. The *support* assists attacking or defending actions of the team.

In many role allocation approaches, the preferred poses of each role are set by the play strategy. A numerical indication (*utility*) of a robot for each role is defined as a function of the relative postures among the robots, the ball and the preferred poses of the role. The robot with the highest utility will be allocated the corresponding role (Stone, P. et al, 1999; Stone, P. & Veloso, M., 1999; Weigel, T. et al, 2002). Roles are also allocated through an auction mechanism where the robots are treated as ``traders". The offer of each robot for a role is measured through a matching function based on the attributes of the robot (Frias-Martinezl, V., et al, 2004). These approaches tend to quantify, in a closed-form function, the relationship between the system states (coordinates, distance and angle of the robots and the ball) and roles. In practice, it is hard to find or justify such functions.

By viewing the robot roles as patterns, the robot role allocation problem can be reformulated as the selection of a pattern (robot's role) from the system states. This typical pattern classification problem can be handled with many powerful tools such as principal components analysis (Amari, S. et al, 2000), neural network (Haykin, S., 1994), support vector machines (Keceman, V., 2001) and evolving connectionist systems (ECOS) (Kasabov, N. & Song, Q., 2002; Kasabov, N., 2002). In this paper, the ECOS method is adopted for its unique evolving feature and its successful applications.

The paper is organized as follows. In Section 2, the problem formulation is described. In Section 3, the procedure of ECOS-based robot role allocation and some practical issues are discussed. In Section 4, the experimental set-up and the results are presented to verify the proposed approach. The conclusion of the work is given in Section 5.

2. Problem Formulation

The layout of a robotic soccer game (http://www.fira.net) is schematically shown in Fig. 1.

With three wheeled robots (dimension: 75mm X 75 mm 75mm) moving in a field (dimension: 150mm X 130 mm), each robotic soccer team tries to push the ball into the opponent's goal net. The states of the robots and the ball (target) are captured by a camera and processed by a computer. The robots receive the motion commands from the computer through wireless communications.

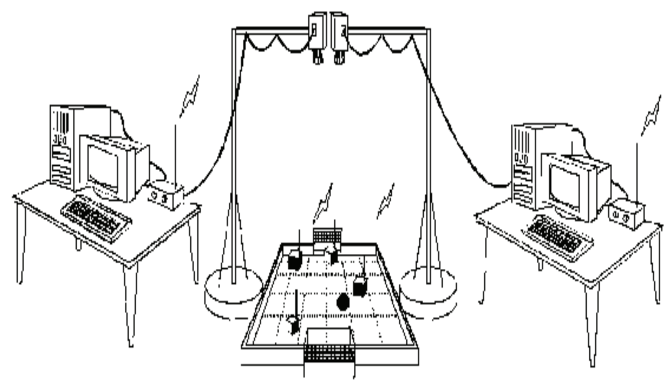

Fig. 1. Robotic Soccer System (http://www.fira.net)

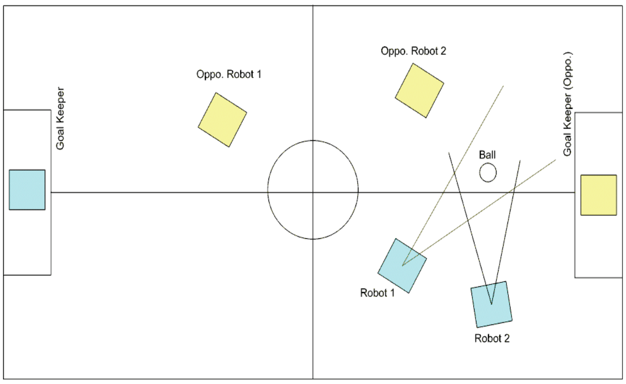

Fig. 2. One Scenario of Robotic Soccer

The role allocated to each robot varies with the progress of the game. Fig.2 shows a scenario when two home robots are near the opponent goal area. The robot in the best attacking posture (the position and the angle of the robot) should be assigned as an attacker, and the others can be defender or goal keeper. For Robot 1 and Robot 2, their positions and angles are denoted as $p_i = [x_i \ y_i]^T$ and $\theta_i \ (i = 1, 2)$ respectively. The ball's position is represented by $p_b = [x_b \ y_b]^T$. Combining p_i, θ_i and p_b, we have a new vector $p = [p_b^T \ p_1^T \ \theta_1 \ p_2^T \ \theta_2]^T$.

The role allocation problem now becomes: given p, what roles, *attacker* or *defender*, Robot 1 and Robot 2 should take ?

3. Role Selection

Evolving connectionist system (ECOS) is a connectionist architecture for modelling of an evolving process and knowledge discovery (Kasabov, N., & Song, Q., 2002). It consists of networks operating continuously and adapting their structures through interactions with the environment. The adaptation of their structures is achieved through a learning mechanism (supervised or unsupervised) in the system. Fig. 3 is the block diagram of a much simplified ECOS with supervised learning. The data Input 1 and Output 1 are for the learning, and the data Input 2 and Output 2 are for the verification. The learning and the verification processes are shown in solid and dashed lines respectively. The structure of this simplified ECOS is similar to those of common supervised learning systems, but it is unique in its learning mechanism able to cater for an evolving process. The structure as well as the parameters of the connectionist elements (neural network, rules etc) are subject to change.

Evolving classification function (ECF), a special ECOS used for pattern classification, generates rule nodes in an N dimensional input space and associate them with classes (Kasabov, N. & Song, Q., 2002; Kasabov, N., 2002). Each rule node is defined with its centre, radius (influence field) and the class it belongs to. A learning mechanism is designed in such a way that the nodes can, be generated

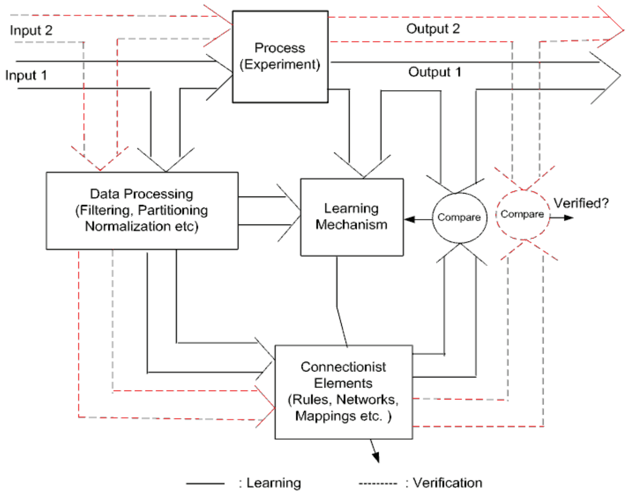

Fig. 3. An ECOS with Supervised Learning

The following notations are used to describe an ECF: C (class set), v_i (i th data vector), $c_i \in C$ (the class associated with v_i), o_j (the centre of j th node), r_j (the radius of j th node), $N_j = (o_j, r_j, c_j)$ (the j th node), $d_{\min}$ (the minimum radius of a node) and $d_{\max}$ (the maximum radius of a node). The range of the indices i and j are determined by the number of data and nodes. Fig.4 schematically shows the classification of a set of data. In the learning phase, a list of (rule) nodes is generated from the following learning algorithm iteratively:

Step 1: Initialize $i = j = 0$ and $N_0 = (v_0, d_{\min}, c_0)$.

Step 2: End the learning phase if no more data coming; otherwise, get v_i and c_i, and increase i by 1.

Step 3: Add 1 to j; calculate $d_{ij} = |v_i - o_j|$ ($j = 1, 2 \cdots m$, where m is the number of the nodes created).

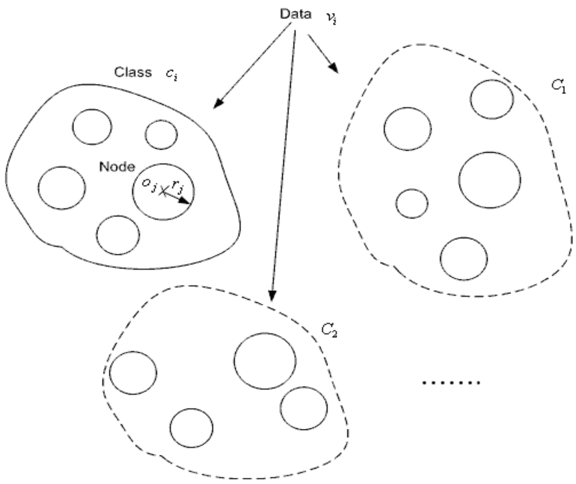

Fig. 4. Data Classification

Step 4: If $d_{ij} > d_{\max}$ for ALL j, increase j by 1, create a new node $N_j = (v_i, d_{\min}, c_i)$ and go to *Step 2*; otherwise:
Step 5: If $d_{ij} \le r_j$ and $c_i = c_j$ for any j, go to *Step 2*; otherwise:
Step 6: If $d_{ij} \le r_j$ and $c_i \ne c_j$ for ALL j, let $r_j = \max(d_{\min}, d_{ij} - d_{\min})$ and then go to *Step 2*; otherwise:
Step 7: If $d_{ij} \le d_{\max}$ and $c_i = c_j$, adjust N_j such that $r_j = d_{ij}$ unless N_j does not cover other nodes; else add j by 1, set $N_j = (v_i, d_{\min}, c_i)$ and go to *Step 2*.
In the recall phase, the class c_i of a new data vector is identified by examining it against the established list of the nodes. It involves the following steps:
Step 1: Input the vector v_i.
Step 2: Calculate $d_{ij} = |v_i - o_j|$ for any node. If $d_{ij} \le r_j$ and c_j is unique, let $c_i = c_j$ and go to *Step 1*; otherwise:
Step 3: If $d_{ij} \le |v_i - o_j|$ and c_j is not unique or $d_{ij} > r_j$ for ALL j, c_i is the same as that of the node with the minimum d_{ij} and then go to *Step 1*.
For the role selection in robotic soccer, the class is defined as $C =$ { *Robot 1 is the attacker, Robot 2 is the attacker*} given one robot is fixed for the goal keeper. The input vector v_i is derived after processing the raw data collected. First, the following variables describing the relative postures among the robots and the ball are defined: $d_{12} = |p_1 - p_2|$, $\theta_{12} = |\theta_1 - \theta_2|$, $d_{ib} = |p_b - p_i|$, $\theta_{ib} = \angle p_b - p_i$, $\gamma_{ib} = \theta_b - \theta_i$ and $d_{ig} = |(L - x_i)\tan\theta_{ib} - W/2|$, where L and W are the length and the width of the field respectively. Next, define a new dimensionless vector $p_t = [w_1 p_{t1} \cdots w_6 p_{t6}]^T$ where $p_{t1} = \gamma_{1b}/(\gamma_{1b} + \gamma_{2b})$, $p_{t2} = d_{1b}/(d_{1b} + d_{2b})$, $p_{t3} = d_{1g}/(d_{1g} + d_{2g})$, $p_{t4} = \gamma_{2b}/(\gamma_{1b} + \gamma_{2b})$, $p_{t5} = d_{2b}/(d_{1b} + d_{2b})$, $p_{t6} = d_{2g}/(d_{1g} + d_{2g})$ and w_i are the weights for adjusting the contribution of each element to the role selection. By default, $w_i = 1$. Note that p_t contains the important information for the role allocation such as the robot's distances (angles) to the ball and the opponent goal respectively,. It is used as a vector v_i in the learning and recall operation in the ECF. For a better classification of data, the raw data p are partitioned according to the position of the ball with respect to the robots :
Case 1: $x_b > x_1$ and $x_b > x_2$ (The ball is in front of all the robots).
Case 2: $x_2 < x_b < x_1$, or $x_1 < x_b < x_2$ (The ball is between the robots).
Case 3: $x_b < x_1$ and $x_b < x_2$ (The ball is behind all the robots).
where the relative position ``in front'', ``between'' and ``behind'' are in reference to the attacking direction. The data can be further partitioned according to the distance between the robots and the ball:
Case 4: $d_{1b} > k_{far} d_{2b}$ or $d_{2b} > k_{far} d_{1b}$ (Big difference between the relative distances to the ball of two robots; $k_{far} > 0$ is a constant).
Case 5: $d_{1b} \le k_{far} d_{2b}$ and $d_{2b} \le k_{far} d_{1b}$ (Normal difference between the relative distances to the ball of two robots).
Case 6: Other cases excluding Case 4 and Case 5.

4. Experimental Platform and Results

The Data collection is the first task of applying ECF in the robotic soccer. To make the data collection and learning more efficient and comprehensive, an application program package is developed. It can capture the system state with a camera in real time and to replay it on the computer screen. The user can select the roles of the robots interactively through a user friendly graphic user interface (GUI). The learning and recall algorithms are also programmed in the package. The data sets for ECF learning are automatically generated and saved as a template file. The GUI of the data collection is shown in Fig. 5.
On the screen, the robot is represented by a color square with its identity number (1 or 2). The line going through the rectangle indicates the direction of the robot. The ball is represented by a circle. By clicking the button ``..Last >>'' or ``>> Next..'', the robotic soccer playing process can be played backward or forward. Examining the scenes on the screen, we can select Robot 1 or Robot 2 as the attacker by clicking the button ``ONE'' or ``TWO'' respectively.
A picture taken in a real game is shown in Fig. 6. There are 122 data collected, among which, 82 data are used for learning and 40 data are used for verification. Some data are listed in Table 1.

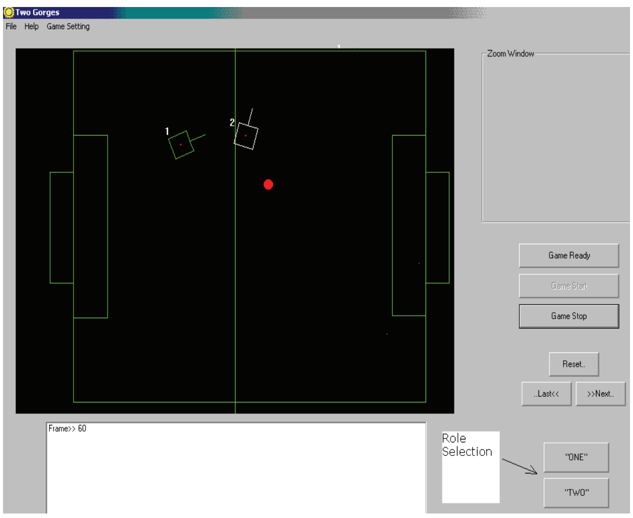

Fig. 5. GUI for Data Collection

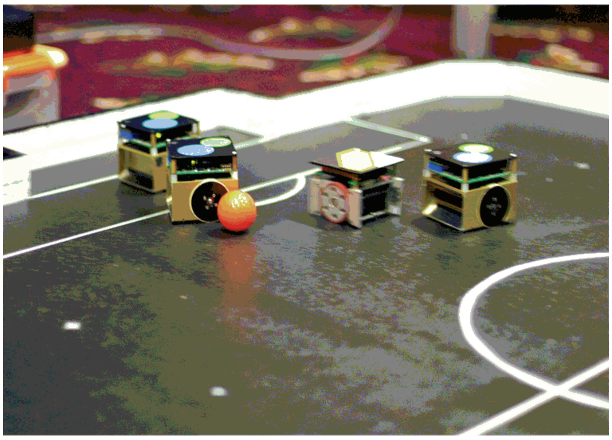

Fig. 6. Robots in Action in a Robotic Soccer Game

xb	yb	x1	y1	θ1	x2	y2	θ2	Attacker's Robot ID
58.61	56.51	48.45	114.88	-20.21	40.2	51.68	8.73	2
66.53	5.33	40.86	50.62	-141.42	49.65	14.01	-98.83	2
66.53	5.39	38.91	27.87	173.95	49.63	7.77	4.52	2
66.48	5.35	53.67	25.05	-94.94	20.18	52.42	38.67	1
113.16	17.76	82.67	10.13	-52.05	93.24	17.56	3.2	2
76.44	124.87	54.1	64.49	-92.48	64	106.67	59.29	2
76.47	124.9	58.2	124.7	3.88	63.83	106.44	55.04	1
21.58	123.75	2.95	105.87	90.75	12.31	104.98	57.14	2
141.11	120.95	115.75	125.02	-69.95	96.75	108.89	25.45	1
...	...	...	...	...	...	...	...	...

Table 1. Raw Data Collected (Partial)

Center						Radius	Class
0.000067	0.687735	0.07988	0.999933	0.312265	0.92012	0.01	2
0.011935	0.797166	0.530409	0.988065	0.202834	0.469591	0.100786	2
0.028868	0.548886	0.039924	0.971132	0.451114	0.960076	0.01	1
0.042638	0.373441	0.054242	0.957362	0.626559	0.945758	0.162074	1
0.068601	0.462405	0.597472	0.931399	0.537595	0.402528	0.01	1
0.085874	0.598857	0.39181	0.914126	0.401143	0.60819	0.01	2
0.094402	0.192386	0.07488	0.905598	0.807614	0.92512	0.17178	1
0.135132	0.745151	0.092002	0.864868	0.254849	0.907998	0.01	2
0.140514	0.467248	0.509745	0.859486	0.532752	0.490255	0.01	1
0.152978	0.401701	0.278058	0.847022	0.598299	0.721942	0.01	1
...	...	...	...	...	...	...	...

Table 2. ECF Nodes (Partial)

The learning parameters are set as $d_{\min} = 0.01$, $d_{\max} = 0.15$, $w_i = 1$ ($i = 1,2 \cdots 6$) and $k_{far} = 1.5$. After the learning process, 30 ECF rule nodes are generated, among which 10 nodes are shown in Table 2. In the recall process for verification, the classes of 39 data (from 40 data) are identified correctly. The success rate of 97.5% is achieved.

5. Conclusion

This paper addresses the issue of robot role selection for soccer playing based on the concept of evolving connectionist system (ECOS). The role selection problem is converted into one of pattern classification solved by an evolving classification function, a special ECOS. The development of an integrated application program for data collection and learning is described. The experimental study and results are presented to demonstrate the effectiveness of the approach.

8. References

Amari, S., Cichocki, A. & Yang, H. (2000), Blind Signal Separation and Extraction: neural and Information-theoretic Approach, *Unsupervised Adaptive Filtering, Vol 1 (ed. S. Haykin)*, John Wiley & Sons, New York, pp. 63—138.

FIRA Official Web Site (2004), http://www.fira.net.

Frias-Martinez1, V., Sklar1, E. & Parsons, S. (2004), Exploring Auction Mechanisms for Role Assignment in Teams of Autonomous Robots, *Proceedings (CD-ROM) of ROBOCUP2004 Symposium*, Instituto Superior Técnico, Lisboa, Portugal, July 4-5.

Haykin, S. (1994), *Neural Networks - a Comprehensive Foundation*, Prentice Hall, Englewood Cliffs, NJ.

Keceman, V. (2001), *Learning and Soft Computing (Support Vector Machines, Neural Networks, and Fuzzy Systems)*, MIT Press, Cambridge, MA.

Kasabov, N. & Song Q. (2002), GA-Optimization of Evolving Connectionist Systems for Classification with a Case Study from Bioinformatics, *Proc. of ICONIP'2002*, Singapore.

Kasabov, N. (2002), *Evolving Connectionist Systems*, Springer.

Stone, P., Veloso, M. & Riley, P. (1999), CMUnited-98: Robocup-98 simulator world champion team, *RoboCup- 98:Robot Soccer World Cup II (ed. M. Asada and H. Kitano)* Berlin, Germany: Springer-Verlag, pp. 61–76.

Stone, P. & Veloso, M. (1999), Task Decomposition, Dynamic Role Assignment, and Low-Bandwidth Communication for Real-Time Strategic Teamwork, *Artificial Intelligence*, 110(2): 241-273.

Weigel T., Gutmann, J.-S., Dietl, M., Kleiner, A. & Nebel, B. (2002), CS Freiburg: Coordinating Robots for Successful Soccer Playing, *IEEE Trans. on Robotics and Auto.*, Vol 18, No 5, pp. 685--699.