Coaching with Expert System Towards RoboCup Soccer Coach Simulation

Ramin Fathzadeh[1], Vahid Mokhtari[1], Morteza Mousakhani[1], and Alireza Mohammad Shahri[2]

[1] Young Researchers Club, Mechatronics Research Laboratory
Department of Computer Engineering, Islamic Azad University
Qazvin Branch, Qazvin, Iran
{fathzadeh, mokhtari, mousakhani}@mrl.ir
http://www.mechatronics.ws,
http://www.yrc-ir.org
[2] Electrical Engineering Department
Iran University of Science and Technology, Tehran, Iran
shahri@iust.ac.ir

Abstract. In this paper we will describe our research in case of using Expert System as a decision-making system. We made our attempt to expose a base strategy from past log files and implement an online learning system which receives information from the environment. In developing the coach, the main research effort comprises two complementary parts: (a) Design a rule-based expert system in which its task is to analyze the game (b) Employing the decision-making trees for generating advice. Considering these two methods, coach learns to predict agent behavior and automatically generates advice to improve team's performance. This structure is tested previously in RoboCup Soccer Coach Simulation League. Using this approach, the MRLCoach2004 took first place in the competition held in 2004.

1 Introduction

In soccer coach simulation league, the coach is one autonomous agent providing advice to another autonomous agent about how to act. This advice is in format of standard coach language called CLang, [1], [2]. The goal of the coach agent is designing intelligent systems to control and observe multiple robots and provide the robots with the suitable methods to enhance their performance, [3]. In accordance with the coach specifications, we have recruited Expert System as a decision management system.

An expert system is a program which attempts to mimic human expertise by applying inference methods to a specific body of knowledge. In this system the information is consistently sensed from the environment, and using a forward-chaining method, system that compares data in the working memory against the conditions of the rules and determines which rules to fire, then the suitable advice is generated and sent to players, [4], [5].

In developing the coach, the main research is concentrated on designing an online learning system. For this purpose, before starting the match log analyzer analyzes the

A. Bredenfeld et al. (Eds.): RoboCup 2005, LNAI 4020, pp. 488 – 495, 2006.

provided fixed opponent log files and gathers data consist ball position, players' positions, game score, play modes and generates an initial strategy based on these data against the opponent team, then puts it in the shared library for use in online coach. Online coach uses this strategy exposed by log analyzer to begin the match against fixed opponent. During the match coach calculates capabilities of the opponent and receives the information such as match time, result of the match, etc. from the field. At first, using expert system with predefined rules, coach models an appropriate strategy against the opponent team. Then by receiving statistical data (such as player position, the players' activity area, etc.) from the environment and using decision-making trees in message generation phase, coach generates suitable advice conforming the CLang structure, afterward sends the messages to the players.

The remainder of this paper is organized as follows: Section 2 comprises log analyzer and online coach. Log analyzer is to provide strategy for online coach. Online coach includes sense phase, expert system, message generation and act phase which its aim is to detect behavior of opponent team and providing a suitable strategy toward opponent team. Section 3 presents the results of our detailed experiments and finally discusses future work and conclusion.

2 Architecture of Coach Agent

This part presents the behavior of coach agent that involves log analyzer and online coach. In Figure 1 this structure is shown:

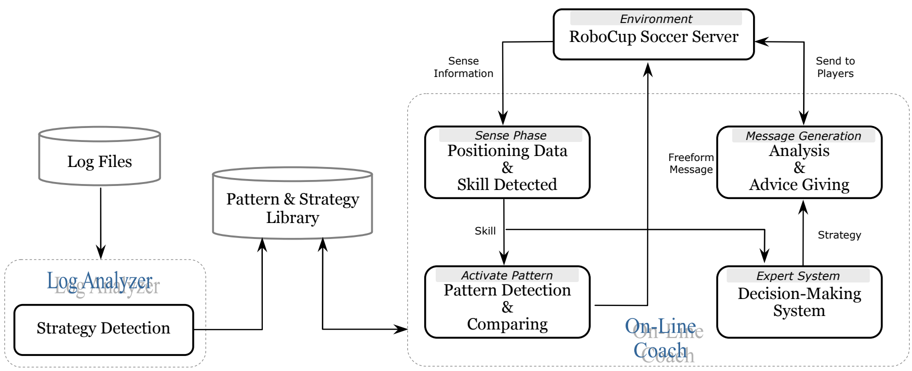


Fig. 1. Agent architecture and its components relation

2.1 Log Analyzer

Before starting the match, log analyzer verifies the provided opponent's log files and collects information such as ball position, players' positions, game score, play modes, sequence possession and number of shoots to target then with similar algorithms used in message generation phase in online coach, creates an initial strategy based on these data against the opponent and puts it in the shared library for use in online coach.

2.2 On-Line Coach

Online coach is divided to sense phase, expert system phase, message generation phase and act phase.

Sense Phase. By a set of features in which consist: our score, opponent score, goal difference, cycle number, distance from each player to the ball, coordinate of the ball and coordinate of each player calculates some skills such as detecting formation, pass graph, each player's activity area, ball position, sequence possession and number of shoots to target then the results of these skills are used to predict opponent ability and generating appropriate advice.

Expert System. In this phase coach makes its effort to analyze and investigate abilities of the opponent team and finally models the state of match in order to provide a strategy against the opponent team. This structure is shown in Figure 2.

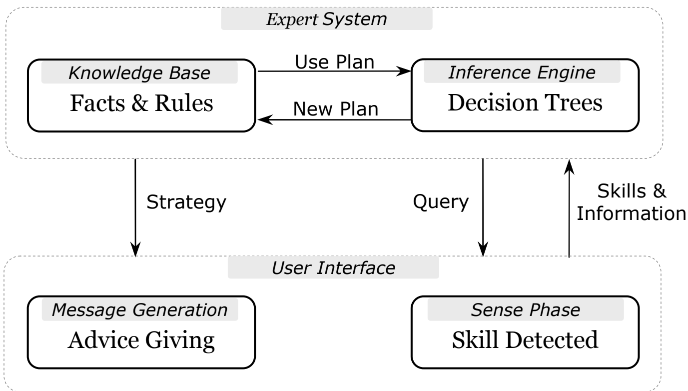


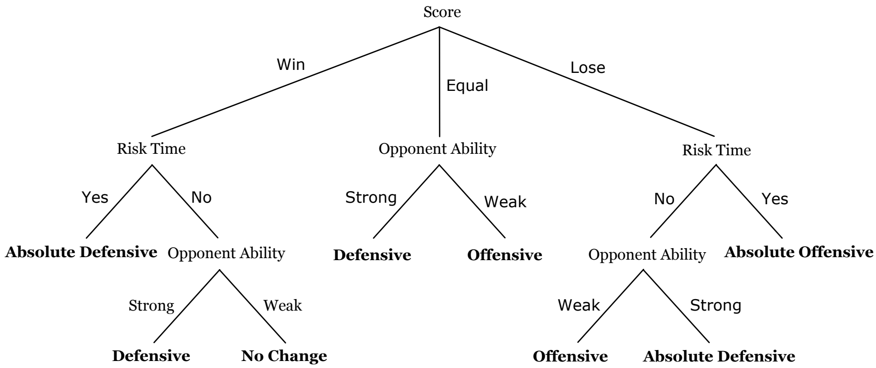
Fig. 2. Process of providing a strategy against the opponent team

To estimate the abilities of the opponent team, information such as ratio of shoots to goal (which amounts to number of attacks of either team), successful passes, ball possession and compression ratio in triple areas of field is investigated. We define some weights for the parameters mentioned above to calculate opponent ability using the following formula:

$$\sum_{i=1}^{max}\left(\left(\kappa_i / |\kappa_i|\right)\left[\beta_i + \left(|\kappa_i|\right)\times\alpha_i\right]\right) \tag{1}$$

max: number of parameters
$\boldsymbol{\beta_i}$: base point for each parameter
$\boldsymbol{\kappa_i}$:difference between the parameters of our team in comparance with the opponent
$\boldsymbol{\alpha_i}$: point scored for each κ_i.

For next step, using a decision-making tree designed on the basis of rule-based expert system architecture, the opponent team is modeled. In Figure 3 the decision-making tree for diagnosing opponent team's performance and giving the strategy against opponent team is presented:




```
Assertions:
IF (GoalDifference > 0) assert (Score Win)
IF (GoalDifference < 0) assert (Score Lose)
IF (GoalDifference = 0) assert (Score Equal)
IF (OpponentAbility > 0) assert (OpponentAbility Strong)
IF (OpponentAbility < 0) assert (OpponentAbility Weak)

Rules:
IF (Score = Equal AND OpponentAbility = Strong)
     OR
     (Score = Win AND RiskTime = No AND OpponentAbility = Strong)
THEN switch to defensive mode
IF (Score = Win AND RiskTime = Yes)
     OR
     (Score = Lose AND RiskTime = No AND OpponentAbility = Strong)
THEN switch to absolute defensive mode
IF (Score = Equal AND OpponentAbility = Weak)
     OR
     (Score = Lose AND RiskTime = No AND OpponentAbility = Weak)
THEN switch to offensive mode
IF (Score = Lose AND RiskTime = Yes)
THEN switch to absolute offensive mode
IF (Score = Win AND RiskTime = No AND OpponentAbility = Weak)
THEN don't change strategy
```

Fig. 3. Decision tree learnt and its IF-THEN rules for giving strategy

Each path from root to leaf analyzes the state of match and opponent team. Finally a strategy is resulted in leaf. This strategy is sent to next phase, companying information and the needed skills for creating advice.

For example assuming that our team has scored more goals than the opponent team, two cases are considerable:

If game cycle is subsequent to Risk Time (This particular time is calculated based on experience and the environment and during this time there is opportunity for scoring goals for the team), in this case coach suggests absolute defensive strategy to keep the result of the game to our side regardless of abilities of opponent team.

If game cycle is prior to Risk Time, coach calculates abilities of opponent team. If the ability is estimated strong, to maintain the attained result, defensive mode is suggested otherwise if the opponent team is weak, the formation of team stays unchanged.

Message Generation. After learning strategy, coach attempts to generate proper advice. The advice is categorized into the sorts below:

Formation: The first step is creating a suitable formation. This formation involves two offensive and defensive modes in which each of them includes predefined templates suiting that state. Each offensive and defensive mode has absolute and ordinary modes as two sub modes.

Let the suggested strategy for team be absolute defensive mode, the arrangement is as follows (our recommended idea as a rule): 2 defenders more than the opponent offenders, 1 middle player of team more than the opponent middle players and other players are put as offenders.

$$\begin{aligned} &\textit{No (our\_defenders)} = 2 + \textit{No (opp\_offenders)} \\ &\textit{No (our\_middles)} = 1 + \textit{No (opp\_middles)} \\ &\textit{No (our\_offenders)} = 10 - [\textit{No (our\_defenders)} + \textit{No (our\_middles)}] \end{aligned} \tag{2}$$

Or in ordinary offensive mode, we put one player of our team for each player of the opponent in triple parts of the field.

And this process is applied to any other strategies.

For example assume that the formation of opponent team is detected as 433 and the suggested formation from expert system is absolute defensive, i.e. the opponent team has more strength than ours. In this case with attention to the predefined templates, formation of the team is calculated as follows:

No (our_defenders) = 2 + 3 = 5
No (our_middles) = 1 + 3 = 4
No (our_offenders) = 10 – [5 + 4] = 1
So the final formation would be 541.

Positioning: By learning formation of team, the positions of players are calculated. Positioning is divided into two modes (1) Marking the offenders of opponent team by our defenders (2) Positioning with ball for rest of the players. The following is the way each is calculated:

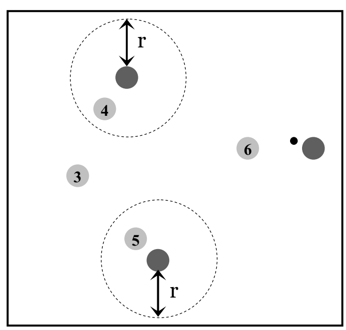


Fig. 4. Marking opponent

1) Mark: To generate the position, the activity area of offenders of the opponent team is calculated, then for marking (i.e. trapping) each offender one defender is summoned. To do this a circle with radius 'r' is assumed around the offender of opponent team, when each of defenders must mark this region. Regarding Figure 4

For example assume ball owner is one of the opponent players, related CLang rule which our player 5 marks offender is shown below:

```
(define (definerule Mark5 direc ((and (bowner opp {X})
       (bpos (rec (pt -52.5 -34) (pt 0 34))))
       (do our {5} (markl (arc (pt opp {$P}) 0 r 0 360)))) ))
```

2) Positioning with ball: In accordance with the calculated formation and activity area of the opponent players, our players' activity area is deduced. Now we segment the field into distinct areas as shown in Figure 5. We define the positions of our players in their activity area, according to the position of ball in the segments.

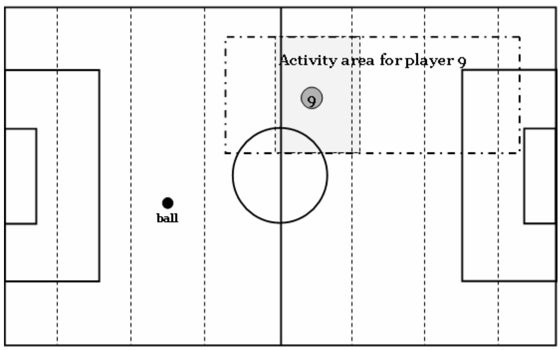


Fig. 5. Positioning player 9 with ball

For example the activity area of player number 9 is shown in Figure 5. With respect to the position of ball, the position of player would be the dark rectangle.

In CLang:

```
(define (definerule Pos4_P9 direc ((and (bowner opp {X})
       (bpos (rec(pt -13 -34)(pt -25 34))))
       (do our {9} (pos (rec(pt 0 -34)(pt 15 34))))) ))
```

Pass Graph: Subsequent to learning the formation, the coach models pass graph of team. For each player we have a decision-making tree to create pass rule. The pass rule is calculated based on the positions of the other players. For player 'i' the pass rule is declared as follow, [6], [7]:

Pass (k):
Pass to teammate with uniform number k Є{X} – {i}
X is set of players whose x-coordinate is greater than x-coordinate of player 'i'.

Assume player number 5 is ball owner and players 6, 7 and 8 are the nearest players to it. This is shown in Figure 6.

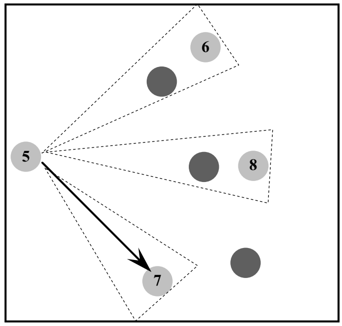


Fig. 6. If there are no opponents in the corresponding passing lane, pass to player k

Pass rule for player number 5 is generated as follows:
if (there aren't opponents in $lane_8$) pass (8)
else if (there aren't opponents in $lane_7$) pass (7)
else if (there aren't opponents in $lane_6$) pass (6)
else do *another action*

In CLang:

```
(define (definerule Pass5 direc ((and (bowner our {5})
      (not (ppos opp {X} 1 11 (tri (pt our 5)
      ((pt our 7) + (pt 0 5)) ((pt our 7) + (pt 0 -5))))))
      (do our {5} (pass {7}))) ))
```

"Another action" could be one of the actions shoot, clear, pass to region and dribble with respect to the position of player. For instance, in offense lane "shoot", in defensive lane "clear", and in the middle of field "pass to region" or "dribble" is chosen.

Act Phase. The act phase contains a queue in which the generated advice of previous phase is put in. Then they are sent to the players as commands.

3 Experimental Result

The MRL coach came in first place out of 12 entries in the 2004 RoboCup Coach competition. The competition consisted of three rounds. In each round, the coached team played three ten-minute games against a fixed opponent. Coaches were evaluated based on goal difference: the number of goals scored by the coachable team minus the number of goals scored by the opponent.

The fixed opponents were all teams that competed in the main simulator competition: Raic2004 in round 1, Hana in round 2, and Kshitij in round 3.

The score differences and rankings for the top four finishing teams are shown in Table 1. Our coach was ranked 1st place in all three rounds. MRL along with FC Portugal, Caspian and Sepanta progressed to the final round with MRL coming out on top.

Table 1. Total scores and rankings for the top four finishing teams in the 2004 RoboCup coach competition. The score consists of the number of goals scored by the coached team followed by the number scored by the fixed opponent.

Coach	*1st Round (RaiC2004)*		*2nd Round (Hana)*		*3rd Round (Kshitij)*	
MRL	0:11	1st	2:6	1st	1:12	1st
FC Portugal	1:13	2nd	2:9	3rd	0:11	2nd
Caspian	1:15	3rd	1:7	2nd	2:15	3rd
Sepanta	0:18	4th	0:13	4th	0:18	4th

4 Conclusion and Future Work

Our prospective effort is to develop a learning system which is able to generate advice in online mode with usage of algorithm for modeling the opponent team. We also plan and try to improve and develop learning algorithms and find out better solutions for generating advice.

References

1. Murray, J., Noda, I., Obst, O., Riley, P., Stiffens, T., Wang, Y., Yin, X., RoboCup Soccer Server User Manual: for Soccer Server 7.07 and later, 2002.H. Simpson, Dumb Robots, 3rd ed., Springfield: UOS Press, 2004, pp.6-9.
2. Patrick Riley and Manuela Veloso, An Overview of Coaching with Limitations. In Proceedings of the Second Autonomous Agents and Multi-Agent Systems Conference, pp. 1110–1111, 2003.
3. Patrick Riley and Manuela Veloso, 2004. Coaching Advice and Adaptation. In Daniel Polani,Andrea Bonarini, Brett Browning, and Kazuo Yoshida, editors, RoboCup-2003: The Sixth RoboCup Competitions and Conferences, Springer Verlag, Berlin.
4. Keith Darlington, the Essence of Expert Systems, England: PERNTICE HALL, 2000.
5. Peter Jackson, Introduction to Expert Systems, Second Edition, England: ADDISON WESLEY, 1990.
6. Gregory Kuhlmann, Peter Stone and Justin Lallinger, The Champion UT Austin Villa 2003 Simulator Online Coach Team. In Daniel Polani, Brett Browning, Andrea Bonarini, and Kazuo Yoshida, editors, RoboCup-2003: Robot Soccer World Cup VII, Springer Verlag, Berlin, 2004.
7. Peter Stone, Layered Learning in Multiagent Systems: A Winning Approach to RoboticSoccer. Intelligent Robotics and Autonomous Agents. MIT Press, 2000.