RCMasters 2D Soccer Simulation
Team Description Paper
RoboCup 2013

Saman Ismael$_1$ and Mohammadali Kamkar$_1$

Faculty of Engineering
Islamic Azad University South Tehran Branch$_1$,Tehran, Iran
ismael@programmer.net, kamkar@live.com

Abstract. This paper describes main features of RCMasters 2D soccer team that has been submitted to qualification for the competition of the soccer 2D simulation league of RoboCup 2013. The most important issue that is demonstrated in this paper is a new approach of decision making and an analyzing scenario-based system.

1 Introduction

RCMasters project consists of 2 stundets of Islamic Azad University of South Tehran Branch and established in 2012. Members of this team have started their activities in this field since 2008 and they achived places as championship of National Robotic Competitions 2010 (Khwarizmi 1389) in 2D soccer simulation league and they hold 2nd place in National Robotic Competitions 2012 (Khwarizmi 1391) and AUTCup 2012 in Urban Robots League and they had the 3rd place in Iran Open 2011 and 2012, 4th place in China Open 2009 and so on. We have released 3 source codes of our teams Radian 2008 , CmorQ and RCMasters based on UVAtrilearn to help newbies in this league on sourceforge [1][2][3].
In this following years we change our base code from UVAtrilearn to agent2d [4], improved pass algorithm ,focused on a opponent analyzing system and developed our researches on a new dribble algorithm that is describes in continue of this paper. In first section a new approach for decision making is presented in and second section describes a scenario-based strategy generator system.

2 Decision Making

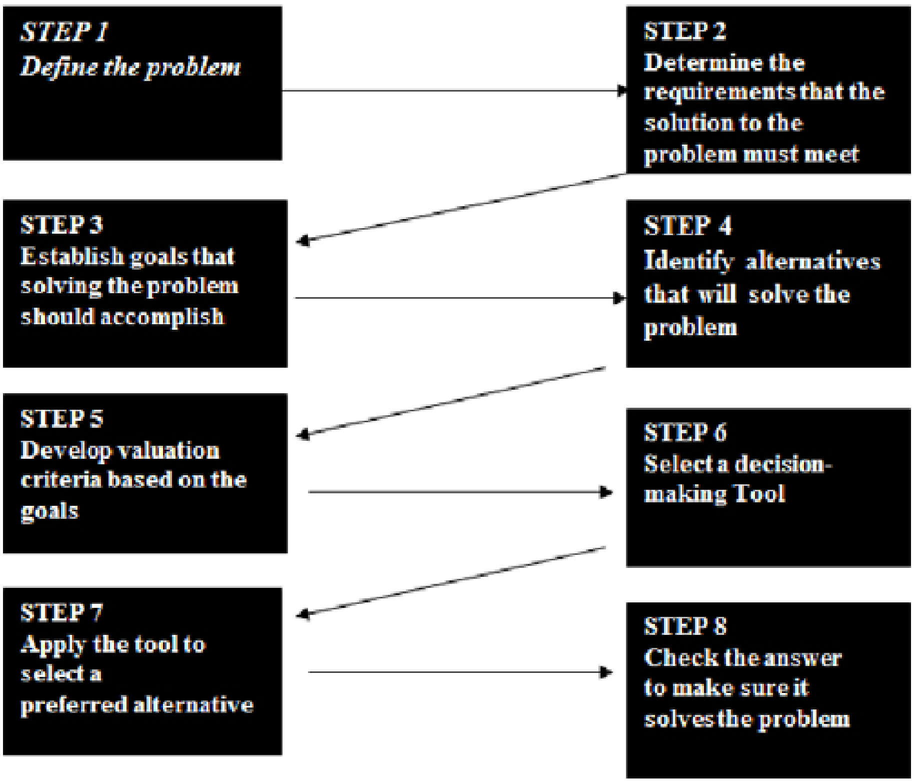
Decision Making is one of important parts of multi-agent system. In this paper we are using a new approach to build a new decision making system based on a real game situation and an online trainer that train decision neurons.

2.1 Decision Making Process

A decision making made up of a composite of information data, facts and belief. Data by itself does not constitute useful information unless it is analyzed and processed.

1. Is only as good as the data that informed it
2. Is only as good as it is an informed one
3. Is only as good as the system which exists to implement
4. Is only good if you have the means to implement it
5. Is only good if other people understand it and what it means

Below describes decision making process steps:


STEP 1
Define the problem
STEP 2
Determine the requirements that the solution to the problem must meet
STEP 3
Establish goals that solving the problem should accomplish
STEP 4
Identify alternatives that will solve the problem
STEP 5
Develop valuation criteria based on the goals
STEP 6
Select a decision-making Tool
STEP 7
Apply the tool to select a preferred alternative
STEP 8
Check the answer to make sure it solves the problem

We split our decision making process to two main sections that explained below: Decision Making:

1. Action Selector
2. Strategy Selector

2.2 Action Selector

There is where we are using QMDMS (Qualitative Matrix Decision Making System); the agent decision selector. Based on real game situation and strategy selected by strategy selector.

2.3 Strategy Selector

Strategy is a long term plan of action design to archive a particular goal. According to this definition strategy selector try to set a strategy that help agents to action better. This strategy can be determined using other agent action and situation or global situation such as goal different and game mode.

2.4 Qualitative Matrix Decision Making System (QMDSM)

QMDSM is a qualitative decision making system that using dynamic weights to select proper action. It is like an artificial neural network (ANN) in every aspect, but we add some transparency, so it is very easier to learn and some ANN weak points are restored. In this implementation we use three tables for store weights and training data[5].

Param-Action Table Each agent has a particular Param-Action table that store weights for fuzzy action selecting. The table value show coefficient for each action per parameter. The proper action selected from maximum SOP parameter using this coefficient.

Matrix X 1	Action 1	Action 2	Action 3
Param1	A1	B1	C1
Param2	A2	B2	C2
Param3	A3	B3	C3

Situation-Strategy Table Table 2 is a sample table that is not trained now. This table shows the situation coefficient per strategy that shows how much a situation is important is particular strategy. Just after an action performed; ball and agent situation extracted and using this weights; strategy point for action calculated.

Matrix X 1	Stretegy 1	Strategy 2	Strategy 3
Situation1	A1	B1	C1
Situation2	A2	B2	C2
Situation3	A3	B3	C3

Action-Strategy Point Table As this is somehow a memory based learning system. Some have a table to keeps recent of results. Proper action weights are extracted from this table using average if max values. This table stores this data for each action, weights of performed { W_1, W_2, W_3, W_4, }, strategy points and etc.

Implementation In this implementation we are using a wide range of various algorithms to optimize and speed up whole system. We are using a new data structure for store these tables that are tasted both Euclidean and Pearson methods for selecting final weights from Matrix 3 and got good results that does not fit in this article. Anyway we are using Euclidean algorithms for now. We are using a very simple strategy selector for coach that select strategy in according of game mode and goal difference. However if Scenario Searching will complete it will used instead.

3 Scenario-Based Strategy Generator System (SBSGS)

One of most important points in even competition matches is analyzing and detect opponent behaviors. In RCMasters we used a new idea for finding opponent attacking scenario. This algorithm used a classified data table which collected from each scene of match and two similarity algorithm for finding best pattern to the opponent.

3.1 Data Collecting

Each scenario made of some scenes, for collecting each scene we need a low-noise observer, coach is the best choice for collecting data from match. Coach extract data and classify and after using them for finding the scenario.

1. Position of each player : consists of X and Y of each effective player in attacking position.
2. Position of ball : consists of X and Y of ball
3. Action of each player : in this system there are 5 type of action for opponent and 3 type of action for teammates.
   (a) Opponents:
      i. Dash without ball (DASH):
      ii. Dash and kick with ball (DRIBBLE) : defines with comparing of ball owner and last action on ball in current cycle and last cycles. In last cycles if the last owner kicked the ball to relative angle from itself and in current cycle still the agent is the owner, it defines as DRIBBLE.
      iii. Kick to teammate (PASS) : defines with comparing of ball owner and action in current cycle and last cycles. In last cycles, if the last owner kicked the ball in specific relative angle to its teammate, and current cycle the teammate is the owner, this action defines as PASS.
   (b) Teammates:
      i. Block the opponent player (BLOCK) : defines when teammate player going to block the ball owner.
      ii. Going to intercept the ball (INTERCEPT): defines when teammate player going to cut the pass route between opponent players.
      iii. Marking opponent players (MARK): defines when teammates are in man-marking position.

iv. Going to default position which defines in formation (HOMEPOS)

When opponents are on attack and we are on defense position, coach starts to save each scene. Each scenario consists of n number of scene which n equals start cycle of being in defensive position and end cycle of outgoing of defensive position.

<table>
<tr><th colspan="4">Sence Information - Cycle 674</th></tr>
<tr><td>Player</td><td>X</td><td>Y</td><td>Action</td></tr>
<tr><td>Opp7</td><td>-22.437</td><td>5.096</td><td>DASH</td></tr>
<tr><td>Opp8</td><td>-33.822</td><td>12.882</td><td>DASH</td></tr>
<tr><td>Opp9</td><td>-29.325</td><td>-6.593</td><td>DASH</td></tr>
<tr><td>Opp10</td><td>-38.105</td><td>16.907</td><td>DASH</td></tr>
<tr><td>Opp11</td><td>-36.029</td><td>-15.089</td><td>DRIBBLE</td></tr>
<tr><td>Tmm2</td><td>-37.401</td><td>-10.239</td><td>BLOCK</td></tr>
<tr><td>Tmm3</td><td>-38.991</td><td>-3.629</td><td>MARK9</td></tr>
<tr><td>Tmm4</td><td>-38.031</td><td>2.463</td><td>HOMEPOS</td></tr>
<tr><td>Tmm5</td><td>-37.949</td><td>9.329</td><td>MARK8</td></tr>
<tr><td>Tmm6</td><td>-35.826</td><td>-10.393</td><td>MARK10</td></tr>
<tr><td>Ball</td><td>-36.189</td><td>-15.329</td><td>OPP11</td></tr>
<tr><td colspan="3">Attacker Numbers (oppNo)</td><td>5</td></tr>
<tr><td colspan="3">Defender Number (defNo)</td><td>5</td></tr>
<tr><td colspan="3">Attacker X avg (oppXAvg)</td><td>-29.9458</td></tr>
<tr><td colspan="3">Ball angle (ballAng)</td><td>63.213</td></tr>
<tr><td colspan="3">Ball Last Owner (ballLOwn)</td><td>OPP11</td></tr>
<tr><td colspan="3">Ball Last owner update cycle (ballLOwnCycle)</td><td>673</td></tr>
<tr><td colspan="3">Type (type)</td><td>NORMAL</td></tr>
</table>

$$\mathrm{E}_1 = \{\ S_{669},\ S_{670},\ S_{671},\ S_{672},\ S_{673},\ S_{674},\ S_{675},\ S_{676}\ \}$$

3.2 Scenario Searching

As it presented in last section each scenario consists of classified tables. Now coach need to find the similarity of collected datasets and the pattern which has been produced in offline mode.This system is under developing and we want to try to use 2 algorithms on this section, first is a greedy decision tree and second is a statistical test called Kolmogorov Smirnov test.

3.3 Decision Tree

In every cycle coach needs to use this algorithm for finding scene and classify them to 4 types. It means when a scene extracted from the match in cycle n it tags by one of these types. Each type shows the type of defensive form needs to be created against opponent team. If system finds repeated types, for example if cycle 669 to 674 will tagged Max Defensive. Overall defensive model will change to a new formation. Below list is defined type for tagging the table :

1. Normal
2. Wide Defensive
3. Max Defensive
4. Pressed Defensive

So after all we will have some scenes from start cycle of attack and end cycle of attack. System will collect and classify all data and depend on this tables it finds the type of each scene. After that it find number of repeated scene and change the defensive model.

4 Conclusion

In this paper, we described a few aspects of our team. The main focus of paper is on opponent analyzing and desicion making. However we developed all of needed skills like Direct Pass, Through Pass, Clear Ball, Dribble, Block, Mark, Penalty and a monitor for analyzing opponent team. Also we are on developing a dribble with reinforcment and it will complete in near future and we hope to complete SBSGS soon.

References

1. CmorQ 2D Soccer Simulation Teamhttp://cmorq.sf.net/
2. RCMasters 2D Soccer Simulation Team, http://rcmasters2d.sf.net/
3. Radian 2D Soccer Simulation Team, http://radian2d.sf.net
4. http://rctools.sourceforge.jp
5. S. Ismael, M.A. Kamkar, Hephaestos 2D Soccer Simulation Team Description Paper, Iran Open 2012, Tehran, Iran.