

**Development of a Synthetic Multi-Agent System:
The KMITL Cadence 2003 Robotic Soccer Simulation Team,
Intelligent and AI Based Control**

Thunyawat Chitipalungsri, Chawit Jirawatsiwaporn, Thanapon Tangchupong, and Surin Kittitornkun.
Dept of Computer Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang,
Bangkok, Thailand

Abstract: This paper describes the development of a synthetic multi-agent called KMITL Cadence 2003. KMITL Cadence 2003 is a robotic soccer simulation team consisting of eleven autonomous software agents. Each agent operates in a physical soccer simulation model called Robocup Soccer Server which provides fully distributed and real-time multi-agent system environment. All teammates have to cooperate to achieve the common goal of winning the game. The simulation models many aspects of the football field such as noise in ball movements, noisy sensors, unreliable communication channel between teammates and actuators, limited physical abilities and restricted communication. This paper addresses the algorithm to develop the soccer agents to perform basic actions which are scoring, passing ball and blocking the opponents effectively. The result of this development is satisfactory because the successful scoring attempts is increased from 11.1% to 33.3%, successful passing ball attempts is increased from 22.08% to 63.64%, and also, successful intercepting attempts is increased from 88% to 97.73%.

Keywords: Multi-Agent System, Artificial Intelligence, Discrete-Event Simulation, Robocup

1. INTRODUCTION

The stated goal of robotic society is to create a team of robot soccer players, which can beat a human world cup champion team within 2050 [1]. The challenge posed by the goal is enormous and inspires hundreds of researchers every year throughout the world to join *RoboCup* [2]. RoboCup has been used as a research challenge for two main purposes which are a usage for educational purposes, and to stimulate the interest of the public for robotics and artificial intelligence (AI). Each year since 1997, researchers from more than 20 countries, which are dramatically increases, have gathered to play the world cup. The event has drawn an increasing amount of interest from the public, as robotics is still not commonplace.

The RoboCup simulation league is based on the RoboCup simulator [3] called the *soccer server* [4], a physical soccer simulation environment. All games can be seen by displaying the field of the simulator by the soccer monitor on a computer screen. The soccer server is written to support competition among multiple virtual soccer players in an uncertain distributed multi-agent environment, with real-time demands. One of the advantages of the soccer server is the abstraction made, which relieves the researchers from having to handle robot problems at physical layer such as object recognition, communications, and hardware issues. The abstraction enables researchers to focus on higher level concepts such as co-operation, learning and decision-making.

This paper describes the development of a robotic soccer simulation team called *KMITL Cadence 2003*. This paper addresses the algorithm to develop the soccer agents to perform high-level player skills which are scoring, passing ball and blocking the opponents effectively. These algorithms are implemented over the UvA (University of Amsterdam) Trilearn 2001 source code [5] using analysis tool called *Soccer Doctor*. The rest of this paper is organized as follows. The detail of robocup soccer simulation and its implementation as a KMITL Cadence 2003 is elaborated in sections 2 and 3 respectively. Moreover, the pseudo-code for KMITL Cadence 2003 algorithms is illustrated in section 4

and also its results in section 5. Finally, the conclusion ends this paper in section 6.

2. ROBOCUP SOCCER SIMULATION

Robocup Soccer Simulation (<http://sserver.sourceforge.net/>) [3, 4] is one of the Robocup (<http://www.robocup.org>) [2] soccer leagues. Robocup is an international robot competition. The goal of Robocup soccer competition is to have a team of robotic soccer players play against the champion of world cup by the year 2050. In order to achieve this ultimate goal, many leagues of robotic soccer have been initiated including humanoid, 4-legged, middle-size, small-size, and simulation leagues. The competition events of these soccer leagues are held every year.

Robocup soccer simulation is a simulated soccer competition between two teams of eleven software agents. The simulation is carried out by a Robocup Soccer Server. The robot agents are clients to this server communicating via the IP network, the Internet. Each agent can be regarded as a robot with near-perfect physical capabilities. The developers must understand how to control each robot (agent) to run at certain speeds, kick the ball at a particular direction and energy, and so on. In order to play with its teammates against another team, each robot or agent is capable to see where the ball is located, to say something, and to hear what its teammates and rivals say. The figure below shows the simulated soccer competition between TsinghuaAeolus 2002 and its clone.



Fig. 1 The simulated soccer competition between TsinghuAeolus 2002 [7] and its clone

3. The KMITL CADENCE 2003

The main objective in any soccer game is to score as many goals as possible. It is, therefore, important for an agent to have a concrete measure and clear policy about whether it should attempt to score in a given situation, and if so which point in the goal he should aim for. In this part, we briefly describe the algorithm using for the agent to determine the best point in the goal, together with an associated probability of scoring when the ball is shot into this point. It is partly based on an approximation method which we have tried to find the optimum constant relative to the approximation equation.

This big and complicated problem can be broken into two sub-problems, **Finding Scoring Probability Problem and Determining the best Scoring Point Problem.**

3.1 Finding Scoring Probability Problem

An important decision for a robotic soccer agent is which action to choose when it has the control of the ball. Possible options are dribbling the ball, passing the ball to teammates, attempting to score the goal and clearing ball. Especially choosing whether to shoot the ball or not is crucial question. This problem needs concrete measures and clear policy to solve this problem effectively. Ideally, the decision is based on the current environment.

To achieve this, we need to solve a problem which is called Finding Scoring Probability problem which can be stated as follows: find the probability of the given point indicating how much the agent likely to score in a given situation. Before finding the solution, we should find the possibilities that the agent kicks but can't be scored. After considering the various situations, it can be concluded that there are only three possibilities that ball are not entering the goal.

Firstly, the ball is intercepted or blocked before entering the goal. Secondly, the ball direction is not aimed at the goal and as a result the ball goes out of the field. Third, the kicking power is not enough for the ball to travel and entering the opponent's goal. The first possibility needs the concrete and effective measure to determine the probability of intercepting the ball from the opponent. The second possibility can be solved by kicking the right direction which must not too close to the goal pitch. The third possibility can be solved by considering the appropriate range of shooting the ball. This

can be seen that it still has one possibility can not be absolutely solved by concrete measure. So, it should be evaluated in probability domain. So, it can be expected that the probability can be calculated with both the angle to the closest opponent relative to the given point and the distance between the agent and the given point.

$$\chi = A\theta + B / \ell. \quad (1)$$

Where

χ is How difficult for opponent to intercept.

θ is Angle Degree between nearest opponent and the agent.

ℓ is Distance between the agent and goal point.

A is Constant 1.

B is Constant 2.

Eq. (1) is for determining how difficult for the opponent to intercept the ball

3.2 Determining the best Scoring Point Problem

In order to determine the best scoring point in the goal, we specify the goal interval $[-\text{goal_width} / 2, \text{goal_width} / 2]$ and compute the total probability the ball will end up in each discretized bin. This total probability is a bell-shaped function which represents the probability that the ball will enter the goal and which has a valley around the position of the goalkeeper. The best scoring point is determined by the global maximum of this curve.

3.3 Pass Ball Selection Algorithm

Passing Ball is one of the most important actions. Choosing to pass ball or not and choosing whether which teammates to be passing ball to are crucial questions. This requires concrete solution for both.

Firstly, we determine which teammates are in the possible position to pass ball to. If there are some, we choose the one who stands in the best position. If none of them are in the possible position to keep the ball, we choose not to pass the ball. So, the problem can be broken into 2 questions which are described below.

The first question is how we can determine how likely for this agent we should pass ball to. What measure should be taken? When we play soccer, we pass the ball to friends who are not too far from us, not currently blocked and the way to the teammates are not likely to intercept by the opponents. It is easy to determine the appropriate range for passing the ball but it is hard when we determine how much likely to determine the path passing the ball to teammates will be intercepted or not. Moreover, it is hard to determine that the teammates are currently blocked or not.

We consider that heuristic algorithm should be taken to determine that the path passing the ball to the teammate is safe or not. The fitness from heuristic algorithm is computed by the equation 1 as described in section 3.1. There is only one different when using the equation which ℓ is Distance between the agent and teammate instead. So, the fitness is a value determining how difficult for the opponent to intercept the ball.

The second question is which agent we should pass the ball to. Since we have the algorithm for determining how likely to pass the ball to a teammate resulting in a fitness value. So we find the teammate who has the maximum fitness

value. If the maximum fitness value agent has fitness value below than the threshold, it can be concluded that there is no appropriate teammate to pass ball to.

3.4 Mark Selection Algorithm

Firstly, it must be stated how aggressive the team have to mark the opponent in different situation. Situation 1: If the ball is in the opponent's control and near my goal, it must be aggressively block every opponent in the specific zone especially the one who controls the ball. Situation 2: If the ball is in the opponent's control but it's quite far from my goal, the opponent who stands quite near to my goal must be blocked. Situation 3: If the ball is my control, the opponents may not be block because we have to make the space for agents to play the ball. Still, there are many situations to be considered. We have to choose the appropriate criteria to distinguish the problem into levels and solve them one by one. We discriminate this problem using marking zone. Marking zone is created by distance in x-axis from my goal.

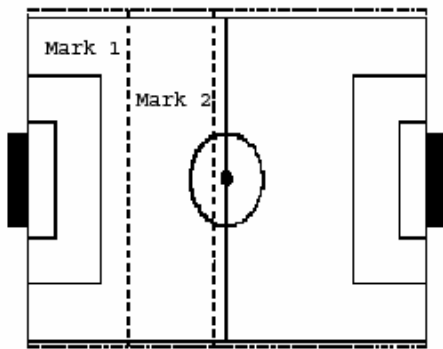


Fig. 2: Marking Zone description
(Left side is my goal and also, right size is opponent goal).

As we can see from figure 2, there is 2 major marking zone distinguished by distance in x-axis as stated above; Mark 1 and Mark 2. Mark 1 is the situation that must be aggressively block and try to intercept ball from the opponents. In this situation, agents will try to block the opponent not to kick to score easily. Mark 2 is the marking zone that requires quite urgent intention to take care of this situation. In this situation, agents try to block opponents in two intentions. First intention is not to let the opponent kick to score. Moreover, blocking opponent for passing ball to their teammates is a second priority.

Furthermore, there must be stated that how each of our agents can block every opponents in the marking zone. The first algorithm in mind may be letting every agent in the zone blocks the nearest opponent agents. So what if three of our agents is the nearest to the only one opponent agent who do not have the control over the ball. So, this is not good algorithm. Our proposed algorithm is marking rendering selection algorithm which is described below.

For every opponent after sorted by Scoring Probability and Fitness

Find nearest teammate to mark the opponent
If nearest teammate has already marked the other opponent then
Find second nearest teammate to mark the opponent
End if

Mark Opponent by the Mark rule-based algorithm
End for

Algorithm 1: Marking Rendering Selection Algorithm

This algorithm requires agents to block the opponent who has the maximum probability to score using Scoring Policies first and so on. Moreover, using the nearest of our teammates to block the selected opponent guarantee that the strategy does not waste many teammate agents to block only one opponent.

As a result, every opponent in the marking zone especially the one who can score easily will be blocked one by one. Using this algorithm can help the agents to have a better chance to intercept ball from the opponents. The details in implementing the algorithms are in the Algorithms section. Also, the statistics of this algorithm is stated in the Experiment Result and Discussion section.

4. ALGORITHMS

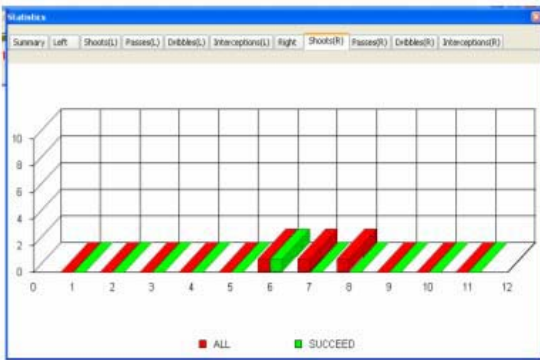
If Agent not facing the opponent's goal
Not kick
Else if the distance between the agent and the opponent's goal more than the threshold
Not kick
Else
For every Point in Goal
If the point at goal and Agent is blocked
Not kick
Else if distance between agent and the point is more than the threshold
Not kick
Else
Fitness how difficult for opponent to intercept =
*Angle Degree between nearest opponent and the agent * W1 + W2 / Distance between the agent and goal point;*
End if
End for
Find maximum of total Distance from Opponent to the point and Fitness how easy pass ball
If the maximum value less than the threshold
Not kick
Else
Kick to the point which has maximum total fitness
End if
End if
End if

Algorithm 2: Scoring Policies.

For each teammate who stands in the range
If the teammate and Agent is blocked
Not pass
Else if distance between agent and teammate is more than the threshold



(a)



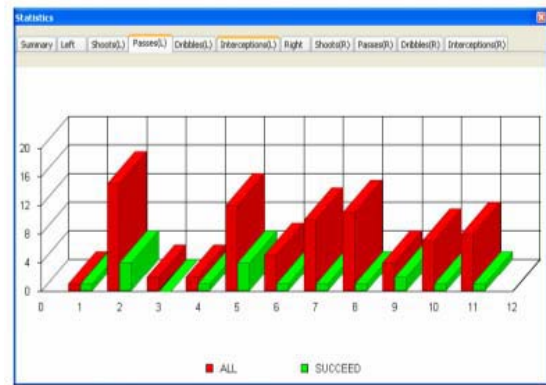
(b)

Fig. 4 Graph representing the successful scoring attempts for UvA Trilearn 2001 (a) and KMITL Cadence 2003 (b) (x-axis represents scoring attempts for each agents in specific team number which red shows all attempts and green shows successful attempts. In the other hand, y-axis represents number of attempts)

We can see from figure 4 (a) and 4 (b) that show successful intercept attempts for UvA Trilearn 2001 and KMITL Cadence 2003, respectively and also Table 1 that there is a significant improvement for scoring accuracy since the scoring attempts decreases but the successful attempts remains the same. So, it increases the percentage of successful attempts from 11.1% to 33.3%. However, we should continue developing the scoring tactics and strategies in order to create more chance to score since the scoring policy does only verify that it should score or not.

Table 2: Pass Ball Result Statistics of UvA Trilearn 2001 (before developed the pass ball selection algorithm) and KMITL Cadence 2003 (after developed the pass ball selection algorithm).

	UvA Trilearn 2001	KMITL Cadence 2003
Pass Ball Attempts	77	132
Successful Attempts	17	84
Percentage of successful attempts	22.08%	63.64%



(a)



(b)

Fig. 5 Graph representing the successful pass ball attempts for UvA Trilearn 2001 (a) and KMITL Cadence 2003 (b)

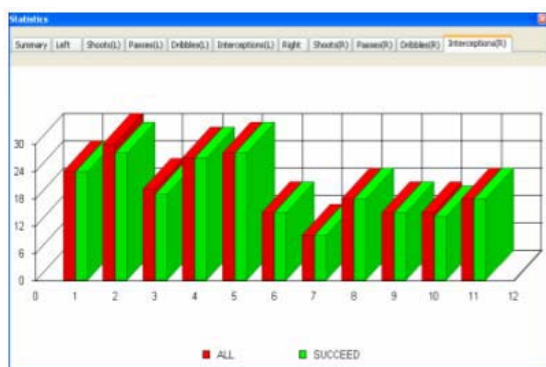
After the improvement in passing ball using passing ball selection algorithm, we can see from figure 5 (a) and 5 (b) that show successful intercept attempts for UvA Trilearn 2001 and KMITL Cadence 2003, respectively and also Table 2 that the percentage of successful attempts increases dramatically. Moreover, the pass ball attempts increases significantly too. This is a supreme improvement in passing ball skills for the simulation team.

Table 3: Intercept Result Statistics of UvA Trilearn 2001 (before developed the mark selection algorithm) and KMITL Cadence 2003 (after developed the mark selection algorithm).

	UvA Trilearn 2001	KMITL Cadence 2003
Intercept Attempts	172	220
Successful Attempts	152	215
Percentage of successful attempts	88%	97.73%



(a)



(b)

Fig. 6 Graph representing the successful intercept attempts for UvA Trilearn 2001 (a) and KMITL Cadence 2003 (b)

We can see from figure 6 (a) and 6 (b) that show successful intercept attempts for UvA Trilearn 2001 and KMITL Cadence 2003, respectively and also Table 3 that there is significant improvement in intercepting result. Both of the intercept attempts and percentage of successful attempts increases dramatically. This is a concrete success of the mark selection algorithm. This not only causes the increment of successful attempts but also decreases both scoring attempts and percentage of successful attempts of scoring of the opponents. This causes a major development in the defensive strategy in the simulation team. The result of this development is satisfactory because the successful scoring attempts is increased from 11.1% to 33.3%, successful passing ball attempts is increased from 22.08% to 63.64%, and also, successful intercepting attempts is increased from 88% to 97.73%.

6. CONCLUSIONS

This paper addresses the development of KMITL Cadence 2003 Robocup Soccer Simulation team. Various algorithms that operate in high-level player skills are provided. Our main contributions include scoring policies, passing the ball policies, and marking the opponent policies. The result of this development is satisfactory because the successful scoring attempts is increased from 11.1% to 33.3%, successful passing ball attempts is increased from 22.08% to

63.64%, and also, successful intercepting attempts is increased from 88% to 97.73%.

ACKNOWLEDGEMENTS

My sincere thanks to Mr. Watchara Chatwiriya, Ph.D., our head of the department, for his comments and suggestions regarding our research and enables us to make this paper possible. Here we need to mention that Mr. Watchara Chatwiriya deserves credit for initiating and supporting the project.

My gratitude to my family for their confidence in me and equal support and encouragement

REFERENCES

- [1] The Road to Robocup 2050, Hans-Dieter Burkhard, Hans-Dieter Burkhard, Dominique Duhaut, Masahiro Fujita, Pedro Lima, Robin Murphy, and Raul Rojas, IEEE Robotics & Automation Magazine, 1070-9932/02, June 2002
- [2] Robocup's Official Website, The Robocup Federation.
At <http://www.robocup.org>.
- [3] User Manual Robocup Soccer Server version 7.07 and later, Mao Chen, Ehsan Foroughi, Fredrik Heintz, ZhanXiang Huang, Spiros Kapetanakis, Kostas Kostiadis, Johan Kummeneje, Itsuki Noda, Oliver Obst, Pat Riley, Timo Steffens, Yi Wang and Xiang Yin, 2001, The Robocup Federation.
- [4] Robocup Soccer Simulation, the Robocup Soccer Simulator Maintenance Group.
At <http://sserver.sourceforge.net/>.
- [5] UvA Trilearn Simulation Soccer Team, Jelle Kok and Remco de Beor.
At <http://carol.wins.uva.nl/~jellekok/robocup/>.
- [6] The Incremental Development of a Synthetic Multi-Agent System: The UvA Trilearn 2001 Robotic Soccer Simulation team, Remco de Beor & Jelle Kok, Master's Thesis for Artificial Intelligence and Computer Science, Faculty of Science, University of Amsterdam, February 2002.
- [7] TsingHua Simulation Soccer Team.
At <http://robocup.lits.tsinghua.edu.cn/>.