Journal of Applied Sciences 11 (1): 111-117, 2011
ISSN 1812-5654 / DOI: 10.3923/jas.2011.111.117


# The Development of Ball Control Techniques for Robot Soccer based on Predefined Scenarios

[1,2]A.H. Pratomo, [1]A.S. Prabuwono, [1]S.N.H.S. Abdullah,
[1]M.S. Zakaria, [1]M.F. Nasrudin, [1]K. Omar, [1]S. Sahran and [1]M.J. Nordin
[1]Center for Artificial Intelligence Technology, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor D.E., Malaysia
[2]Department of Informatics Engineering, Faculty of Industrial Engineering,
University Pembangunan Nasional Veteran Yogyakarta, Jl, Babarsari No. 2, Jogjakarta, Indonesia

**Abstract:** Robotic soccer is an attractive topic in artificial intelligence and robotics research. However, to develop techniques and algorithms in this domain is a complex task. This study presents the development of ball control techniques and algorithms for robot soccer based on several predefined scenarios. In this study, we study the robot can do ball passing, obstacle avoiding and ball shooting according to certain situations. A vision system is used in this case where it calculates the robot position in x, y coordinates to make sure the robots move to the right direction. The velocity of each robot wheel is manipulated to control the speed of the robots and allow them to make turning and shooting. Algorithm testing was carried out by using a robot soccer simulator. Several techniques in obstacle avoiding and positioning were successfully implemented. The results prove these algorithms can be applied to execute the given tasks.

**Key words:** Ball control, position algorithm, predefined scenario, robot soccer strategy

## INTRODUCTION

Robotic soccer is a common task in artificial intelligence and robotics research. This task permits the evaluation of various theories, the design of algorithms and agent architectures (Pratomo *et al*., 2010). Robot Soccer was introduced in 1994 with the theoretical background to develop multi-robots adaptive, co-operative, autonomous systems solving common tasks (Asada and Kitano, 1999). A group of robots shall interact and self-organize autonomously in order to achieve a common goal. Further technical aspects besides co-operative and coordinated behavior are miniaturization of a complex electro-mechanical system, precise movement and optimal power efficiency (Kim *et al*., 1997a). There are several categories in robot soccer: Simurosot, MiroSot, Narosot and HuroSot, classified by the size of the robots and the number of playing robots. SimuroSot consists of a server which has the soccer game environments (playground, robots, score board, etc.) and two client programs with the game strategies (Kim *et al*., 1997b).

In this study, we implement several algorithms for robot soccer strategies. We create a strategy which makes the robot dribbles a ball while avoids obstacles and moves towards a passing position before makes the ball passing. Simultaneously, another robot avoids an opponent robot to move to the shooting area, receives the ball and shoots the ball to the goal. The robot shooter has to beat the opponent's goal keeper to score a goal.

## MATERIALS AND METHODS

Figure 1 shows the development steps. We develop and test numerous strategies using the prototyping approach in the Laboratory of Computational Intelligence, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia. We adopt the best strategy in the robot soccer game to complete the required task.

**Task:** A pictorial scenario based on several tasks as mentioned earlier, is shown in Fig. 2. We need to analyze

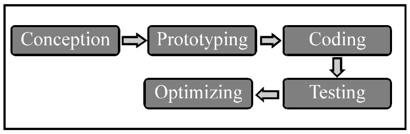


Fig. 1: Development steps

**Corresponding Author:** Anton Satria Prabuwono, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 UKM, Bangi, Selangor D.E., Malaysia

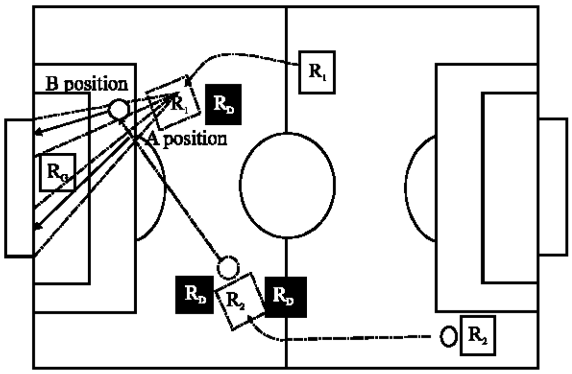


Fig. 2: Scenario of the game

and understand the situation to identify the best movements for the robots as depicted in Fig. 2. Briefly, the $R_1$ robot must avoid the $R_D$ robot that acts as an obstacle, chase and shoot the ball into the goal. At the same time, the $R_2$ robot dribbles the ball along the bottom field and passes the ball through two $R_D$ robots as shown in Fig. 2. The $R_G$ robot must stop the ball from enter the goal. In our scenarios all the $R_D$ robots are the defender in the static positions. These robots act as the obstacles.

**Approach:** The code of robot soccer strategy is developed using Microsoft Visual C++. We divide the scenario into two parts, Part A and Part B as shown in Fig. 3 (refer to Fig. 4 and Fig. 5 for the detailed version).

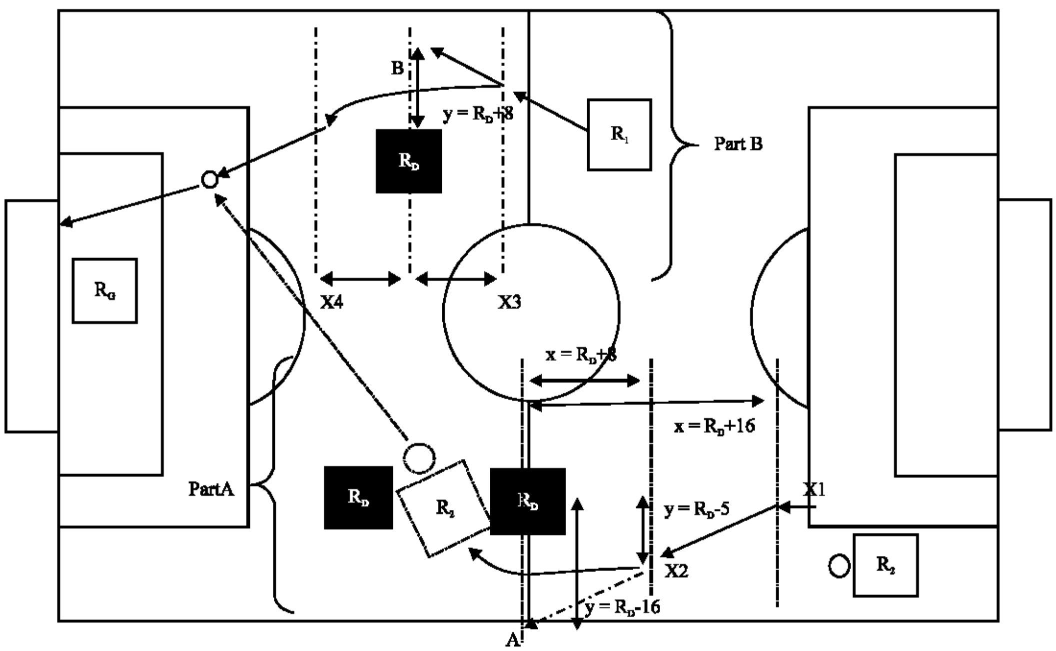


Fig. 3: Strategy plan

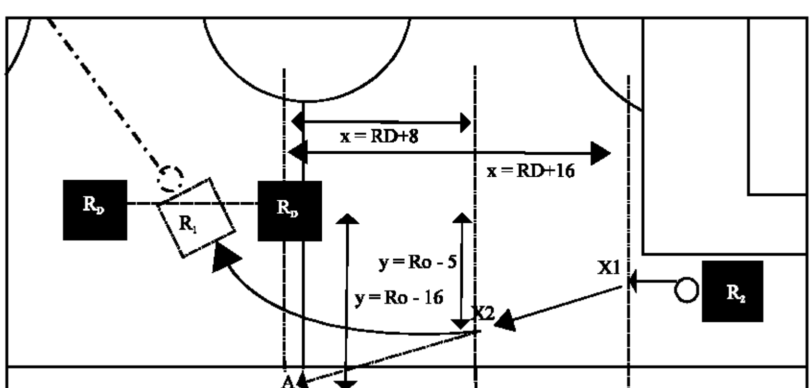


Fig. 4: The part A strategy

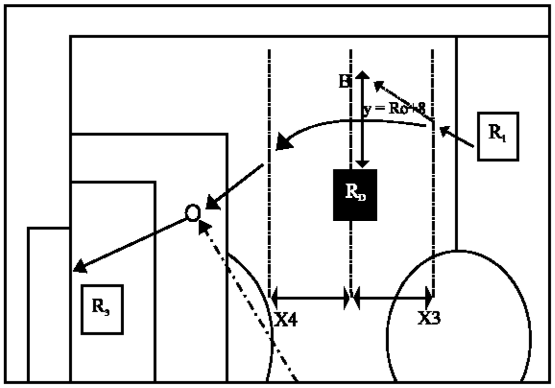


Fig. 5: The part B strategy

The task of part A is to assign movement for the $R_2$ robot to dribble the ball along the bottom field and to shoot the ball through two $R_D$ robots. On the other hand, the part B is to assign movement for the $R_1$ robot to avoid $R_D$ robot, chase the ball that moves towards it and shoot the ball straight to the goal. Those scenes are divided in such away to facilitate the source code development.

**Development of the part A strategy:** First, we assign a movement to the $R_2$ robot to move forwards slowly until $x_1$. Then, the robot changes its direction to point A which is located below the $R_D$ robot. The distance between the $R_2$ robot and the $R_D$ robot is shown in Fig. 4. Our vision system used 640×480 image resolution that makes 2,67 pixels equivalent to 1 cm in real measurement. When the $R_2$ robot arrives at $x_2$, it makes a half circle turn approaching the field obstacles and hits the ball through two $R_D$ robots. We fix the coordinate of $x_1$ and $x_2$ according to the Fig. 4.

**Development of the part B strategy:** We assign the movement for the $R_1$ robot as the followings. The $R_1$ robot avoids the $R_D$ robot and then chases the ball that moves towards it. Then, the $R_1$ robot shoots the ball into the goal. The $R_1$ robot only starts moving when the $R_1$ robot and the $R_2$ robot are in parallel of each other in its way to x4-axis. The $R_1$ robot will approach x3-axis and B point. At x3-axis, the $R_1$ robot makes a turn slightly to avoid the third $R_D$ robot, and gradually approaches the x4-axis. From x4-axis, the $R_1$ robot chases the ball using a predict ball function and shoots it into the goal. We fix the x3 and x4 -coordinates according to Fig. 5.

**Algorithms:** We develop algorithms of strategy part A and B based on flow chart as depicted in Fig. 6 and 7 subsequently.

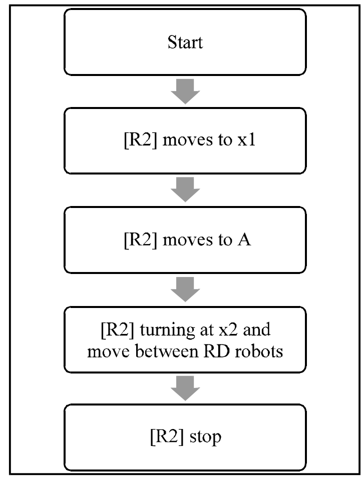


Fig. 6: The part A plan

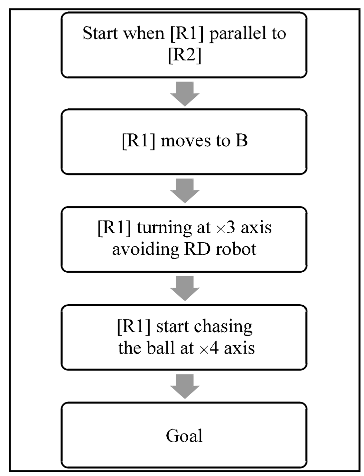


Fig. 7: The part B plan

**Testing:** We conducted several tests and then we recorded the number of success goal and the percentage of successfully performed the required tasks. We checked the unsuccessful task by replaying the simulator. We noted down the problems and improved them by changing the algorithms and setting parameters such as robots position, turning point, speed of the robot and velocity for turning.

**Optimization:** To optimize the performance and to produce the acceptable results in this task, we try out many methods and change the setting parameters to increase the number of success goal. First, we adjusted the constant value that indicates the robot passing time. Second, we initialized the position of the three robots in this task. The position must be exactly the same as the given scenario in order to get more accurate movement.

## RESULTS

In this task, we used fuzzy algorithm. The position and predict algorithms are included in mathematical functions such as method to calculate angle, distance and position prediction of the ball and desired position of the robot (Egly *et al.*, 2005; Huang, 2009).

**The $R_2$ dribbles the ball:** The ball dribbling is a technique to control the ball and to bring it to the desired location (Liu *et al.*, 2004). We used the best method and strategy to control the ball. In the robot soccer, each player has its own velocity coefficient to chase or shoot the ball. From the algorithm, we set the velocity of the robot into a consistent value during ball dribbling. We also set the robot's speed into a constant value to ensure that both wheels are in balance. The speed will only change when it meets the $R_D$ obstacle. The $R_2$ robot changes its position when it meets the $R_D$ obstacle and automatically tries to pass the ball to the $R_1$.

**The $R_2$ avoids the obstacles to the $R'_2$ position and then passes the ball to the shooting area:** Figure 8 shows a scenario when the $R_2$ dribbles the ball and changes its action after it meets the obstacles and passes through the center line.

Here, we employ the two techniques. First, we use the Cartesian plane (coordinates) concept when the ball is near to the $R_D$ obstacles. Second, we use obstacle avoiding strategy when the $R_1$ faces both the $R_D$ robots. By using these two strategies, the $R_2$ can pass the ball through the two $R_D$ robots and move forward to the shooting area as shown in Fig. 9.

The $R_1$ moves to the $R'_1$ position and avoids the robot opponent defender $R_D$: As shown in the Fig. 10, after the $R_1$ meets the $R_D$ obstacles, it will shoot the ball to the shooting area (refers to the circle in the Fig. 10).

In this situation we provide two attacking strategies. In the first strategy, the simulator has FBOT (field bottom) and FTOP (field top) values to determine the center line. After the ball crosses the center line, the $R_1$ starts to move forward and will try to avoid the $R_G$ obstacle. In the second strategy, after the ball moves through the center line and meets the $R_D$ obstacle, the $R_2$ takes the action to shoot the ball. We combine both of the strategies to make the $R_1$ and the $R_2$ move simultaneously. The $R_1$ and $R_2$ share their own positions values in order to move to the shooting area and finally to make a goal.

**The $R_1$ receives or chases the ball:** Figure 11 shows the robot receives and chases the ball. In this situation, we try to find the best strategy. Based on the previous scenario strategy, the basic concept of the ball chasing as well as the obstacles avoidance is known. We create the codes to detect how far is the $R_G$'s obstacles. The $R_1$ will try to catch the ball and kick the ball towards goal regardless of the ball position. The ball might not exactly at the desired location because of human errors during the ball and robots setting before the game.

**The $R_1$ shoots the ball to the goal in A or B position:** Basically, the attacker will try to make a goal as fast as possible with a high kicking speed. It is the same situation as shown in Fig. 12. The $R_1$ will shoot the ball immediately after the ball is received. When the attacker reached the goalie area, we increase the $R_1$ speed

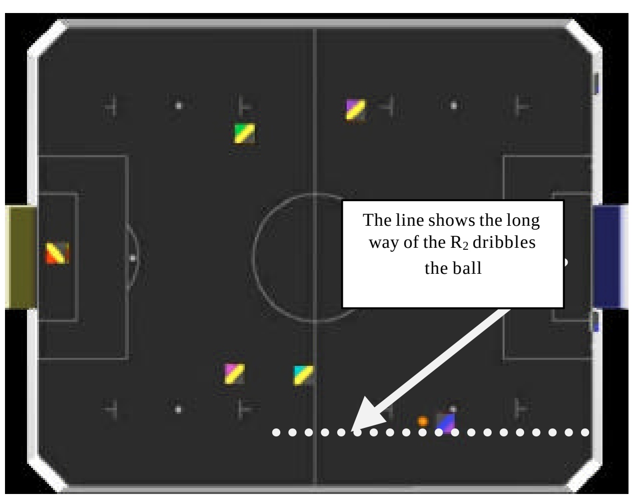


Fig. 8: The $R_2$ dribbles the ball


Fig. 9: The $R_2$ dribbles the ball and changes its action

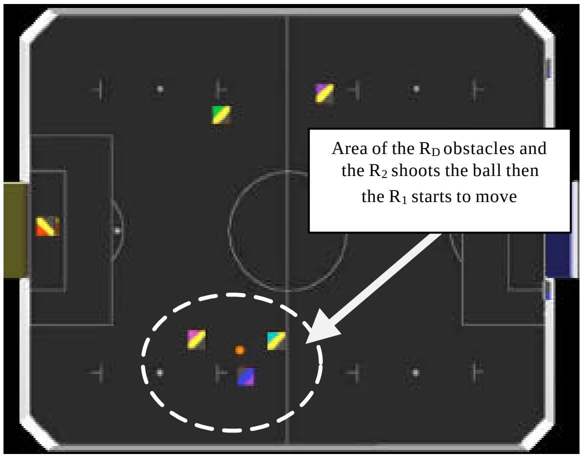


Fig. 10: The $R_2$ shoots the ball and the $R_1$ starts to move

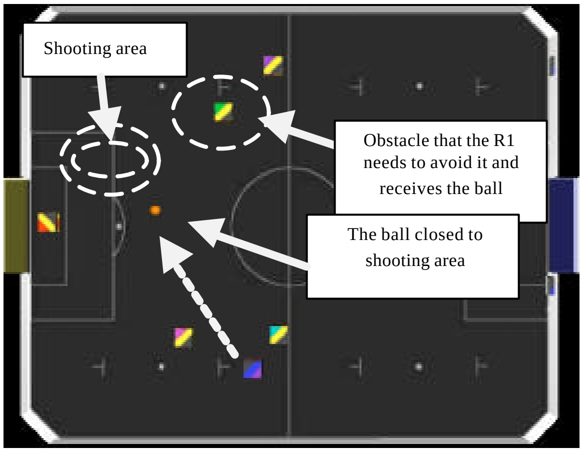


Fig. 11: The $R_1$ avoids the $R_G$ obstacles and the ball closed to shooting area

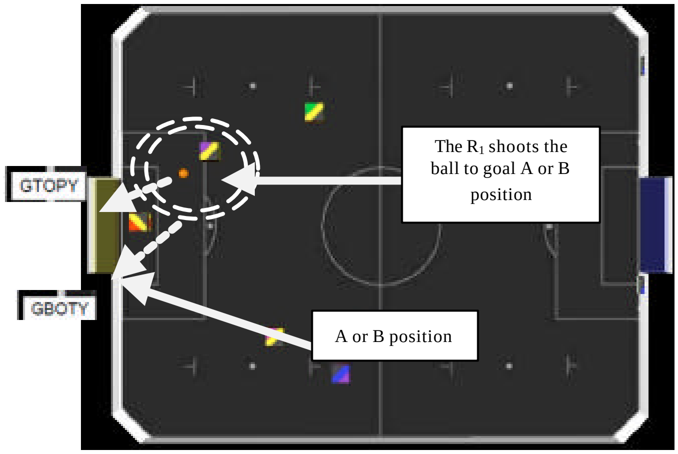


Fig. 12: The $R_1$ shoots the ball in A or B position

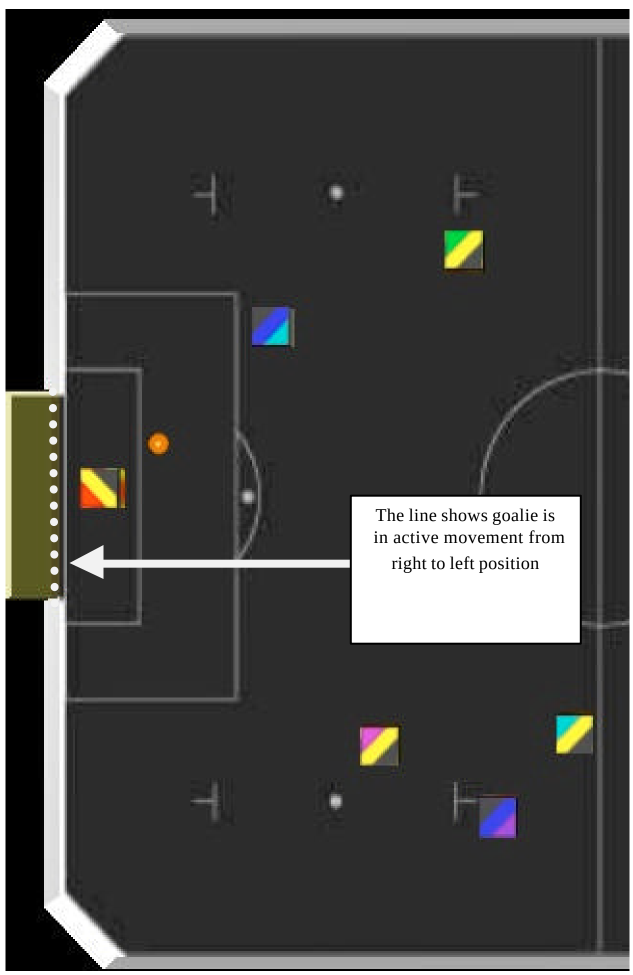


Fig. 13: Goalie defends the goal

Table 1: Tasks definition

| Task | Definition |
|---|---|
| $T_1$ | The R2 robot dribbles, avoids the obstacles ($R_D$) and passes the ball |
| $T_2$ | The R1 robot moves and avoids an obstacle ($R_D$) |
| $T_3$ | The R1 robot is in the right position and shoots the ball |
| $T_4$ | The ball enter the goal |

immediately. If the ball is very near to GTOPY or GBOTY corner (goal line or goal box), we set or give more velocity to the $R_1$.

**Goalkeeper robot, $R_G$, defends the goal:** The attacker will always try to make a goal and the goalie will defend the goal area. In this situation, we increase the attacker velocity to shoot the ball regardless whether the goalie is ready or not. The implementation of the idea into the robot soccer simulator is shown in Fig. 13.

## DISCUSSION

The developed ball control techniques based on predefined scenarios were successfully tested. The tasks are defined as shown in Table 1.

The experiment shows that almost all $T_1$ tasks are successfully implemented. It can be observed that $T_2$ task has correlation to the achievement of $T_3$ and $T_4$ tasks. The shooting accuracy in $T_3$ task depends on the performance of $T_1$ task. Besides, $T_3$ task determined by the accuracy of the ball prediction as well. Furthermore, $T_4$ task will be achieved when the $T_3$ task is successfully done.

## CONCLUSION

Many possible strategies and techniques can be applied to the robot soccer. The programmer can implement their own idea into the code regardless of the code structure. The important idea is the programmers must have a basic knowledge in robot soccer and know the related strategies and techniques in robot soccer. In this case, we applied our ideas into robot soccer game simulation. The first idea is using the Cartesian concept. Second, we calculated the area in simulator. Third, we used an existing strategy of robot soccer. Finally, we modified the robot soccer parameters such as increases the attacker speed, obstacles avoiding in near position, and dribbling with controlled velocity. Furthermore, we combined all the strategies and techniques to get the optimum result.

After the task completed, we then applied a variety of interesting ideas of the robot soccer movement. We combined all possible techniques such as dribbling, chasing, and shooting. The results show the strategies and techniques had been successfully implemented. The position and prediction algorithms have proved can be applied in the real robots.

## ACKNOWLEDGMENTS

The authors would like to thank Faculty of Information Science and Technology, University Kebangsaan Malaysia for providing facilities and financial support under Research University Operation Project No. UKM-OUP-ICT-36-186/2010 and Fundamental Research Grant Scheme No. UKM-TT-03-FRGS0131-2010. Besides, thanks to Muhammad Nuruddin Sudin, Muhamad Syafiq Shohaimi, and Dwi Yanuar Panji Tresna for their contribution in this research.

## REFERENCES

Asada, M. and H. Kitano, 1999. The robocup challenge. Robotics Autonomous Syst., 29: 3-12.

Egly, U., G. Novak and D. Weber, 2005. Decision making for Mirosot soccer playing robots. Proceeings in the 1st CLAWAR/EURON/IARP Workshop on Robots in Entertainment, Leisure and Hobby, April 2005, Viena, Austria, pp: 69-72.

Huang, L., 2009. Velocity planning for a mobile robot to track a moving target-a potential field approach. Robotics Autonomous Syst., 57: 55-63.

Kim, J.H., H.S. Shim, H.S. Kim, M.J. Jung, I.H. Choi and J.O. Kim, 1997a. A cooperative multi-agent system and its real time application to robot soccer. Proceedings of the IEEE International Conference on Robotics and Automation, April 25-27, Albuquerque, New Mexico, pp: 638-643.

Kim, J.H., J.J. Lee and T. Fukuda, 1997b. Evolutionary robotic system. Proceedings of Micro-Robot World Cup Soccer Tournament, June 1-5, Kaist, Taejon, Korea, pp: 29-37.

Liu, J.S., T.C. Liang and Y.A. Lin, 2004. Realization of a ball passing strategy for a robot soccer game: A case study of integrated planning and control. Robotica, 22: 329-338.

Pratomo, A.H., A.S. Prabuwono, M.S. Zakaria, K. Omar and M.J. Nordin *et al*., 2010. Position and obstacle avoidance algorithm in robot soccer. J. Comput. Sci., 6: 173-179.