Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 102 (2016) 477 – 484

Procedia Computer Science

12th International Conference on Application of Fuzzy Systems and Soft Computing, ICAFS 2016, 29-30 August 2016, Vienna, Austria

Robot soccer control using behaviour trees and fuzzy logic

Rahib H.Abiyev[a*], Irfan Günsel[a], Nurullah Akkaya[a], Ersin Aytac[a], Ahmet Çağman[a], Sanan Abizada[a]

[a]*Applied Artificial Intelligence Research Centre, Near East University, P.O. Box:99138, Nicosia, North Cyprus, Mersin 10, Turkey,*

Abstract

Running time, flexibility and on-line adaptation are important features presenting to the decision making (DM) module of soccer robots. In this paper, the design of DM algorithm based on behaviour trees and fuzzy obstacle avoidance algorithm for control of soccer robots are presented. The basic elements of the behaviour tree (BT) used in decision making process of soccer robots are described. The integration of BT and fuzzy logic is proposed to increase the flexibility of the decision made by DM modules. Given algorithms have been extensively tested in simulation and provided satisfactory results in run time and adaptation to the new situation. Also, the efficient navigation procedure is designed using fuzzy obstacle avoidance algorithm. The obstacle avoidance algorithm uses fuzzy if-then rule base and inference mechanism for finding optimal path for avoidance of obstacles. The presented BT based decision making and fuzzy obstacle avoidance algorithms are used for control of the holonomic 4-wheel-driven soccer robots. The obtained results demonstrate the effectiveness of proposed algorithms in soccer robot control.

Peer-review under responsibility of the Organizing Committee of ICAFS 2016

*Keywords:*Robot Soccer; behaviour trees; fuzzy inference; obstacle avoidance.

1. Introduction

In robot soccer game the environment where soccer robots move is characterised with fast changing dynamic areas with moving dynamic obstacles. Control system of these soccer robots includes a set of algorithms. These are vision module, decision making, path finding, obstacle avoidance and motion control algorithms. Decision making (DM) module of soccer robots is a basic module that analyses the current state of the world model and makes decisions about the new positions of each robot and then plans a motion for these robots[1]. The decision making in a

* Rahib H.Abiyev. Tel.: +090-392-2236464; fax: +090-392-2236622.
*E-mail address:*rahib.abiyev@neu.edu.tr

Peer-review under responsibility of the Organizing Committee of ICAFS 2016
doi:10.1016/j.procs.2016.09.430

short time, flexibility and on-line adaptation are basic features presenting to DM module of soccer robots working in a rapidly changing environment. In this paper, the designs of DM module and fuzzy obstacle avoidance algorithm of soccer robots are presented.

Different approaches have been used to design the DM module of soccer robots. The most used are finite state machines (FSM)[2,3], Petri nets[4], behaviour-based control[5,6,7], topological graph map[8]. FSM approach has been demonstrated good performance in designing decision making mechanism of soccer robots[2,3,9]. The FSM algorithm has been tested in simulation and provided satisfactory results in many research works. But for complex situations, the use of FCM in designing of DM module of soccer robots increases the number of states required to encode the behaviour of the robots, along with the number of transitions between the states and modelling of these situations becomes difficult.

Petri nets have a larger modelling power than FSM and can model the state space with a smaller graph[4]. Petri nets have modularity, as each module can be modelled separately and then composed with others. The composition of Petri nets usually leads to an exponential growth in the state space. But in some cases, the generation of all states with Petri nets lead to the construction of large network which complicates the analysis of the network.

In this paper, BT approach, and also the integration of BT and fuzzy logic is proposed for decision-making. The use of BT simplifies the decision-making process[5,6,10]. Behaviour trees use more restrictive and more structured traversal approach to replace the growing mess of state transitions of FSMs. BTs easily define complex states and it is very easy to see logic in BT. They are fast to execute and easy to maintain. The integration of BT and fuzzy logic is proposed to increase the flexibility of the decision made by DM modules. It was shown that some behaviours of the soccer robots are more efficiently described by integration of fuzzy logic and BT.

The soccer robots are moving in unpredictable, cluttered, unknown complex and dynamic environments. In this environment, the avoidance of mobile robots from the obstacles becomes important problem. Many obstacle avoidance algorithms are proposed[11-21]. “Bug” algorithms follow the edges of obstacles without considering the goal[11]. They are time consuming. Artificial potential field is most commonly used method that uses attractive and repulsive fields[12] for the goals and obstacles, respectively. When there are many obstacles in the environment the field may contain a local minima; the robot unable to pass through small openings such as through doors; the robot may exhibit oscillations in its motions[12]. Vector Field Histogram (VFH) uses a two-dimensional Cartesian histogram grid as a world model and the concept of potential fields[13]. The VFH algorithm[13,14] selects a shorter path than bug algorithms but it takes more time to manipulate. Other goal oriented algorithms are dynamic window[15], “agoraphilic”[16], A-star[17] search algorithms which are time consuming. The above methods sometimes cannot solve the problem in reasonable time. Some improvements of path finding algorithms have been given in[17,18,19].

Rapidly-exploring Random Trees (RRTs) algorithm developed by Kuffner and LaValle is a faster algorithm and can be applied for path finding in dynamic environments[20,21]. But frequently the path determined by RRTs may be very long. The RRT-smooth algorithm was proposed to short the RRT length[10]. In the paper fuzzy obstacle avoidance algorithm is proposed to find short path in a short time.

The paper is organized as follows. Section 2 describes the structure of control system of soccer robots playing football game. In section 3 the descriptions of BTs used in decision making have been described. Section 4 presents the design of a fuzzy obstacle avoidance algorithm used for soccer robots. In Section 5, the simulation studies and real life application of the proposed algorithms are presented.

2. Structure of control system of soccer robots

In Fig. 1, the graphical representation of robot soccer field for playing a soccer game, is shown. The environment where soccer robots move in is characterized with fast changing dynamic areas with moving dynamic obstacles. Collision free navigation of soccer robots in such dynamic environment is very important. The current model of the environment, captured by four cameras is sent to the SSL vision system. All objects on the field are tracked by a standardized SSL vision system. SSL-Vision uses a standard Cartesian coordinate system in meters and radians. The centre of the soccer field is accepted as (0,0). x is horizontal axis, y is vertical axis with headings given in radians. The model of the environment includes image information consisting of the coordinates of each robot, the coordinate of the ball and the information about the field. The host computers of both teams receive this information and then analyses for making a decision on new positions of each robot. The host computer determines a strategy finds the paths and plans the motions and the corresponding velocity commands for every soccer robot. The decisions are processed by the computers and control signals through wireless are sent to the microcontroller module of each soccer robot. Each soccer robot receives velocity command from the host computer and regulates the rotational velocities of the forward and rear right and left wheels.

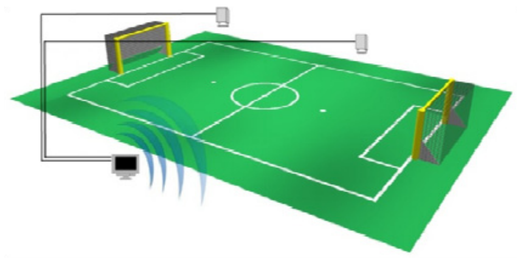


Fig.1. The graphical representation of robot soccer field

The detailed structure of the control system of soccer robots which is designed in this paper is given in Fig.2. The map of the world is captured by the camera and sent to the SSL vision system. SSL vision processes the map and obtains the coordinates of soccer robots and balls then sends them to the control system. From the camera, the data is sent every 1/60 of a sec to tracker module. Tracker module receives this data stream and converts it into a data structure. DM module uses this data structure to make the strategic planning of soccer robots and to define the new position of each robot. DM uses BT based decision making mechanism given in section 2. After selecting certain behaviours, the path finding block starts to look for a path for each robot. During the move, the obstacle avoidance algorithm is run for each robot. After defining the target coordinates of the robots the output control signal determined for each robot and the velocity of the each robot is calculated. The velocities are transformed to the motor speeds of robot's wheels and send to the robot microcontroller through a wireless channel[22-24].

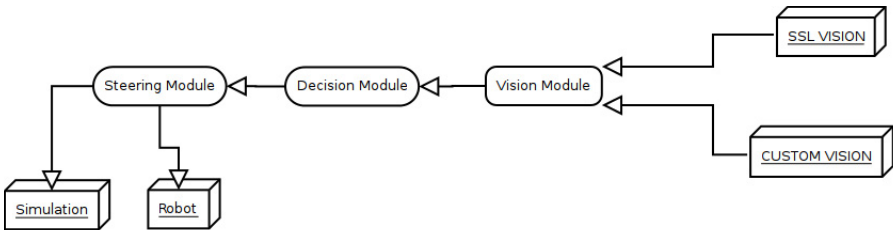


Fig. 2. Structure of the control system

3. Decision making based on behaviour trees

3.1. BT nodes.

All the nodes used in BT can be classified into leaf and non-leaf nodes. The Leaf nodes are action nodes that change the state of the robot such as calculating a new path or kicking the ball. The non-leaf nodes may have finite number of children. In the paper, we use root, selector, sequence and decorator non-leaf nodes are used to control the flow within the three.

Selectors and sequences are basic flow blocks of BT. The selector node (Fig. 3) will sequentially try to execute its child nodes from left to right until it receives a successful response. The selector node successes if one of its children will succeed. If all child nodes respond with failure the selector node will return a failure. Selector start executing its children from left to right beginning with "Shoot Goal" sequence. The success of "Shoot Goal" sequence will cause the selector to be succeeded. If "Shoot Goal" sequence fails the selector will try to execute the next "Pass" sequence. If "Pass" will succeed the selector will succeed. In another case, the selector will fail.

A sequence includes a series of behaviours that needs to be accomplished. The sequence will try to execute all its children from left to right. If all of its children succeed, the sequence will also succeed. If one of its children fails, the sequence will stop and return failure. Fig.4 describes sequence node "Pass" that has three child action nodes.

"Pass" sequence will start executing its children from left to right. If the conditions are true, then it will execute the pass and return success up the tree.

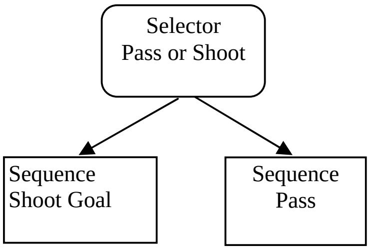


Fig.3. Implementation of selector node.

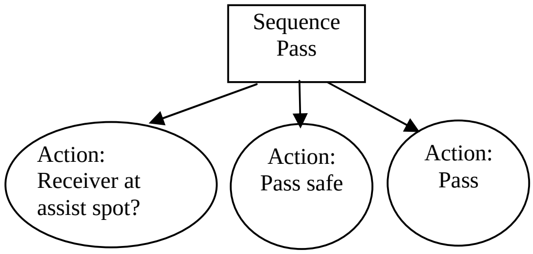


Fig.4. Implementation of sequence node.

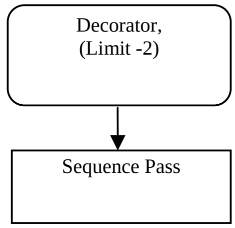


Fig.5. Decorator's descriptions.

Decorators are nodes with a single child are added to the BT to provide more flexibility and control the flow within the three. Commonly decorators are used like filters with conditions such as, execute once, execute with some probability, etc. Besides filtering, decorator can modify the behaviour of the branch. Fig.5 shows the same "Pass" sequence with a decorator as its root. The decorator will try to play the pass. If a pass is made it will return success up the tree, if "Pass" sequence will fail decorator will try it n more times to execute a pass, if a pass can't be made within n times it will return fail up the tree. In the research, the BT is extended and used for the composing tree. Brief definition of these nodes are given in[13,22-24].

3.2. Fuzzy implementation of BT nodes

Soccer robot is operating in unpredictable, uncertain and dynamic environments. Making decision against desired outcome needs the accurate evaluation of the environment. In the soccer robot control, fuzzy logic includes a range of degrees for decisions denoted as a membership function. Here integration of fuzzy inference and BT is described for the selector behaviour. Membership functions are defined for the child of the selector. For example, the deterministic Move behaviour can be implemented using Run, Walk and Turn (Fig.6(a)). Implementation one of these operations depends on a set of conditions. Depending on the input data obtained from the environment the membership degree is determined for the each of the decision. The selection of each block is determined by the condition defined in the body of BT. The fragment of the rule is given below.

If ball is near and ball speed is low then execute sequence walk

If ball is far away and ball speed is high then execute sequence run

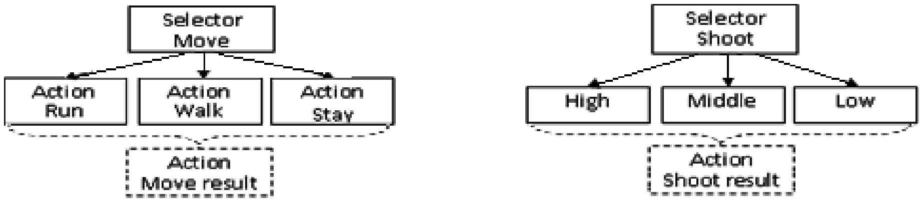
If ball is far away and ball speed is very low then execute sequence stay

On the base of experimental data and expert knowledge, the fuzzy rule bases (RB) are designed for given processes. The implementation of BT is carried out on the base of formulas given below. The output of the fuzzy BT will have continues value. For example in the deterministic case for the tree given in figure 6(a) the output of tree is one of the children- Run, Walk or Stay. Each of these classes has characterised by certain crisp value. In fuzzy case, there is no crisp restriction between the bound of these values. The value of these classes are the fuzzy interval, they will not have so strict ground. In the result of fuzzy inference, the calculated final speed of the soccer robot may not belong to the deterministic Run, Walk or Stay. It may have continues value that will belong to the one of the value between zero and a maximum speed of the robot. Another BT is about Shoot Ball fuzzy selector node that could be implemented in three different ways are given in Fig.6(b).

Using the rule base the output of the BT is determined. The inference engine in BT is implemented using max-min composition. Inference of fuzzy system is performed by the following formula

$$\tilde{X}1, \tilde{X}2 \rightarrow \tilde{Y} \tag{1}$$

where $\tilde{Y}$ is a speed of the robot characterising one of the output Run, Walk, Stay, $X\tilde{1}, X\tilde{2}$ are distance to the ball and speed of the ball respectively. After defining the membership degrees of the input signals for each active rules in rule base the fuzzy logic inference is performed using max-min composition.



(a) (b)

Fig.6. Selector BTs. (a) Move, (b) Shoot

4. Fuzzy obstacle avoidance

In mobile robot navigation, next important problem is the designing of obstacle avoidance algorithm. Robot directly follows to the goal when no obstacles are detected. In more research the implementations of these operations are carried out by measuring left, direct and right distances from the obstacles. However, this approach does not find the optimal path, every time. In the paper, we use left and right angles to implement obstacle avoidance algorithm. These angles are used to design fuzzy knowledge base of the mobile robot. For designing knowledge base, the left and right angles between the line connecting robot and goal, and the line connecting robot and left and right sides of the obstacle are used. These angles are used in the knowledge base for making a decision by a robot. An example scenario that demonstrates obstacle avoidance is shown in Fig.7. In robot soccer game the obstacles will be the soccer robots of another team. During avoidance of obstacle the boundary of the obstacle is further enlarged to ensure the safety of the robot. This expanded boundary is called the “safe boundary”. This ensures the safety of the robot during avoidance of obstacles. Each obstacle will have two different boundaries, the “real boundary” and the “safe boundary”, as shown in Fig. 7. After detecting obstacle the real boundary and then the safe boundaries of obstacles are determined using the navigation algorithm.

During navigation of robot the angle between the direct line connecting robot and target, and the line connecting robot and safe obstacle boundary are determined for the left and right side of the robot. In the paper, this is done by inspecting left and right side of the obstacle in $\theta_s=30^o$. After detecting the left and right boundaries of obstacles, the corresponding left $\theta_l(k)$ and right $\theta_r(k)$ angles are determined. Using these angles a decision is made by the robot. The direction that has the smallest angle to the goal direction is selected. The calculated new angle to avoid an obstacle will be

$$\theta(k)=F(\theta_l(k),\theta_r(k),\theta_s(k)) \tag{2}$$

$\theta_l(k)$ and $\theta_r(k)$ are angles based on the real boundary and the goal directions for left and right sides, respectively, and $\theta_s(k)$ is the safe angle. Table 1 shows the selection rule. In designing of the fuzzy navigation algorithm the Gaussian membership functions are used. Meanwhile, it can also be assumed that there are five membership functions for each of the input linguistic variables θ_l and θ_r , respectively. The membership functions will be denoted as *VS* (Very Small), *S* (Small), *M* (Medium), *L* (Large), and *VL* (Very Large), respectively.

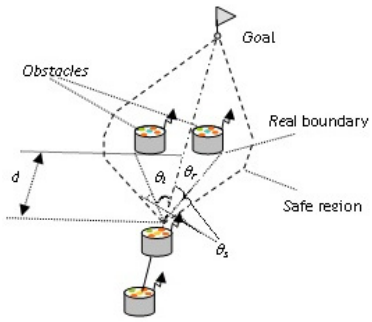


Fig. 7. Mobile robot avoiding obstacle

Table 1. The rule base

$\theta(k)$		$\theta_l(k)$				
		Very Small	Small	Mediu m	Large	Very Large
$\theta_r(k)$	Very Small	$\theta_{ls}(k)$	$\theta_{rs}(k)$	$\theta_{rs}(k)$	$\theta_{rs}(k)$	$\theta_{rs}(k)$
	Small	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{rs}(k)$	$\theta_{rs}(k)$	$\theta_{rs}(k)$
	Medium	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{ls}(k))$	$\theta_{rs}(k)$	$\theta_{rs}(k)$
	Large	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{rs}(k)$
	Very Large	$\theta_{ls}(k))$	$\theta_{ls}(k))$	$\theta_{ls}(k)$	$\theta_{ls}(k)$	$\theta_{ls}(k)$

With reference to Table 1, $\theta_{ls}(k)= \theta_l(k)+ \theta_s(k)$ and $\theta_{rs}(k)= \theta_r(k)+ \theta_s(k)$. Using this table the TSK (Takagi-

Sugeno-Kang) fuzzy rule base is developed. Since there are five membership functions for $\theta_l(k)$ and $\theta_r(k)$ respectively, there will be 25 fuzzy rules. The values of the left and right angels $\theta_l(k)$ and $\theta_r(k)$ are defined with fuzzy linguistic values. Fig. 8 presents the linguistic values defined for the left and right angles. The form of membership functions are accepted in triangle form.

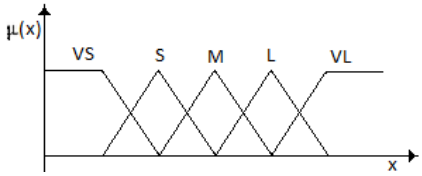


Fig. 8. Linguistic values

$$\mu(x) = \begin{cases} 1 - \dfrac{\bar{x} - x}{\alpha}, & \bar{x} - \alpha \le x \le \bar{x} \\ 1 - \dfrac{x - \bar{x}}{\beta}, & \bar{x} < x \le \bar{x} + \beta \\ 0, & \text{in other case} \end{cases} \tag{3}$$

The above given formulas (1) and (3) are adopted to obstacle avoidance using input variables left and right angles in order to determine output turn angle of soccer robot by fuzzy logic system.

5. Experimental studies

BT based control with obstacle avoidance algorithms are applied for the soccer robots which are designed in our research laboratory. The designed robots are used in RoboCup competition. Here the problem is to make strategic planning and implement control of each robot on the base of coordinates of the robots and the goal. Design includes two stages: the simulation of fuzzy obstacle avoidance algorithm and real life implementation of BT based decision making and fuzzy obstacle avoidance algorithm in robot soccer game.

5.1. Simulation of fuzzy obstacle avoidance algorithm

Simulation of fuzzy obstacle avoidance algorithm has been done. When the algorithm is run the input position of goal, start position of the robot, positions of obstacles, the threshold distance to the obstacle are entered. After entering these values the robot starts to move towards the goal. For avoidance from the obstacle the angle between the line connecting the robot to the goal and the left and right furthest points of the robot is measured. The safe angle is added to the determined angle in order to find the left and right angles for the obstacle avoidance. These left and right angles were used as input for rule base. Using rule base and fuzzy inference engine the robot is making a decision about turn angle of the robot for avoiding an obstacle. According to the turn angle, the robot changes direction in order to avoid from the obstacle. After avoiding obstacle the robot continues to move to the goal while checking the presence of an obstacle in the certain small distance. Fig.9 depicts the graphical simulation result of robot navigation. Table 2 demonstrate the simulation results using A*, APF, RRT-Plan, RRT-Smooth[13] and fuzzy obstacle avoidance algorithms. Here the results are obtained for 1000 runs. As shown the time and distance results of the fuzzy obstacle avoidance algorithm are better than other ones.

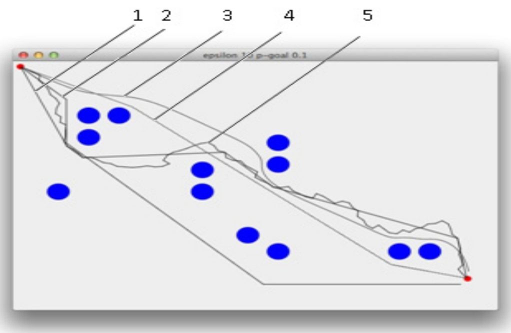


Fig.9. Obstacle avoidance, 1- A star, 2- RRT-Smooth, 3- APF, 4- Fuzzy, 5 - RRT

Table 2. Simulation results of different algorithms

Methods	Time	length
A* (Grid Size=10)	22.5347799	792.5483399
APF	102.477935	732.0000
RRT Plan	8.2619800	849.9
RRT Smooth	14.870870	748.336417
Fuzzy	0.9072543	703.25468

5.2. Real life implementation

The soccer robots and their control system are designed and manufactured in our research laboratory. The holonomic wheels with 3 degrees of freedom are used to design soccer robots[22-24]. Fig.10 depicts the designed holonomic robot soccer (a) and its control structure (b). The soccer robots have four omniwheels that are controlled with the brushless DC motors through gear mechanisms. The driver of each motor is driven by a microcontroller. Each robot has a micro-controller board that controls the control circuits of forward and rear motors of wheels, a control circuit of dribbler and kicker mechanisms.

The above given decision making mechanism and obstacle avoidance algorithm are used for designing the control system of soccer robots and applied 6-vs-6 robot soccer game. In the results of the decision making, the new coordinates of the soccer robots are computed. Using these coordinates soccer robot will move to its target locations. These coordinates are used to calculate and update the velocity of the robot that guides the robot along the path. Using this velocity a set of four motor speeds one for each wheel is calculated and sent to the soccer robots. The control of direction and rotation of the soccer robot is implemented by changing the speed of the individual omni-wheels[10]. A PID controller is used to control the rotational speed of the soccer robot which keeps it pointing to an angle determined by the DM block. The computer sends the values of speeds to the soccer robots through wireless. Once a velocity is calculated in meters/sec that will guide the robot towards a waypoint on the path, this velocity needs to be converted to individual motor speeds in radians/sec.

(a)

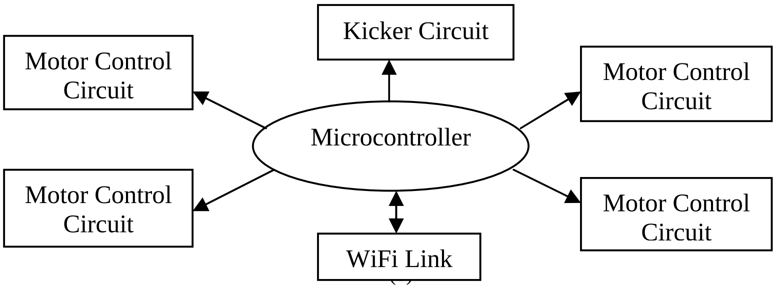


Fig.10. (a) Soccer Robots, (b) control structure of the robot-soccer

At first, the designed algorithms are tested using a simulation software package grSim. Fig.11 is the snapshot of robot soccer games, where the ‘team 1’ is in charge of defending the left goal and attacking the right goal while the ‘team 2’ has the reverse duties as ‘team 1’. Effective offence and defence were observed in each team. Each team is equipped with the same decision-making system. The designed DM and obstacle avoidance algorithms are also tested on real soccer robots of NEUIslanders team, in real life application. The average time for making a decision was 0.02 ms. In the case of the simple tree the running time was 0.003 ms. Run time for the more complicated BT having many levels of sequence and selection nodes was 0.2 ms. In different complex situations, the robots have done different decisions. Fig.12 depicts the robot soccer team of NEUIslanders team playing football game using above-presented behaviour based DM and fuzzy obstacle avoidance algorithms in RoboCup competition.

6. Conclusions

The behaviour tree based decision making and obstacle avoidance algorithms have been integrated with a fuzzy logic theory for efficient navigation of soccer robots which are designed and manufactured in our research laboratory. The modularity of the behaviour tree based approach allows easily extend the decision-making mechanism for complex states. The integration of BT with fuzzy logic has been done to increase the flexibility of decision. The proposed algorithm has been tested in simulation and provided satisfactory results. Also, the proposed fuzzy obstacle avoidance algorithm has shown satisfactory results in avoidance of obstacles. In the result of the application of BT based decision-making mechanism and fuzzy obstacle avoidance algorithm, the required time for reaching the new position is decreased. It was shown that the proposed algorithm efficiently finds the new decision in short time and shortens the path length during avoidance of the obstacles.

Fig. 11. Snapshot robot soccer game.

Fig. 12. The real game in RoboCup competition: NEUIslanders Robot soccer team playing football game

References

1. Kalyanakrishnan S, Stone P. Learning Complementary Multiagent Behaviors: A Case Study. In Baltes J, Lagoudakis MG, Naruse T, Ghidary SS, editors, *RoboCup 2009: Robot Soccer World Cup XIII*, Springer Verlag; 2010, p.153–165.
2. Damas B, Lima P, Tecnologia ESD. Stochastic discrete event model of a multi-robot team playing an adversarial game. In Proc. of 5th IFAC/EURON Symposium on *Intelligent Autonomous Vehicles - IAV2004*, 2004.
3. Dadios EP, Park SH. Real time robot soccer game event detection using finite state machines with multiple fuzzy logic probability evaluators. *International Journal of Computer Games Technology* 2009;
4. Costelha H, Lima P. Modelling, analysis and execution of robotic tasks using Petri nets. In Proc. of Int. Conference on *Intelligent Robots and Systems IEEE/RSJ*; 2007. p.1449 –1454.
5. Mo H, Tang Q, Meng L. Behavior-Based Fuzzy Control for Mobile Robot Navigation. *Mathematical Problems in Engineering* 2013.
6. Lim CU, Baumgarten R, Colton S. Evolving behaviour trees for the commercial game DEFCON. In Proc. of the Int. conference on *Applications of Evolutionary Computation, Part I*, Lecture *Notes in Computer Science*, Springer-Verlag; 2010, p.100–110.
7. Vadakkepat P, Miin OC, Peng X,, Lee TH. Fuzzy Behavior-Based Control of Mobile Robots. *IEEE Transactions on Fuzzy Systems* 2004; 12(4):559-565.
8. Neto G, Costelha H, Lima P. Topological navigation in configuration space applied to soccer robots. Robocup 2003, *Lecture Notes in Artificial Intelligence*, Springer-Verlag Berlin Heidelberg; 2004. p.551-558.
9. Kurihara N, Hayashi R, Fujii H, Sakai D, Yoshida K. Intelligent control of autonomous Mobile Soccer Robot adapting to dynamic environment. In: Polani D. et al., editors. RoboCup 2003, *Lecture Notes in Artificial Intelligence*, Springer-Verlag Berlin Heidelberg; 2004, p.568-575.
10. Abiyev RH, Akkaya N, Aytac E, Ibrahim D. Behaviour Tree Based Control For Efficient Navigation of Holonomic Robots. *International Journal of Robotics and Automation* 2014; 29(1):44-57.
11. Sezer, V, Gokasan M. A novel obstacle avoidance algorithm: Follow the Gap Method. *Robotics and Autonomous Systems* 2012; 60(9): 1123-1134.
12. Borenstein J, Koren Y. Real-time obstacle avoidance for fast mobile robots in cluttered environments. In Proceedings of *IEEE* Int. Conference on *Robotics and Automation*; 1990. p.572 –577.
13. Borenstein J, Koren Y. The vector field histogram - fast obstacle avoidance for mobile robots. *IEEE Journal of Robotics and Automation* 1991; 7(3):278–288.
14. Ulrich I, Borenstein J. Vfh+: Reliable obstacle avoidance for fast mobile robots. In Proceedings of the IEEE Int. Conference on *Robotics and Automation*; 1998. p.1572-1577.
15. Fox D, Burgard W, Thrun S. The dynamic window approach to collision avoidance. *IEEE Robotics Automation Magazine* 1997;4(1):23-33.
16. Ibrahim MY. Mobile robot navigation in a cluttered environment using free space attraction ”agoraphilic” algorithm, In Proc. of the 9th Int. Conference on *Computers and Industrial Engineering*; 2002. V.1 p.377–382.
17. Yan Z, Zhao Y, Hou S, Zhang H, Zheng Y. Obstacle Avoidance for Unmanned Undersea Vehicle in Unknown Unstructured Environment. *Mathematical Problems in Engineering* 2013.
18. Abiyev R, Ibrahim D, Erin B. EDUrobot: An educational computer simulation program for navigation of mobile robots in the presence of obstacles. *International Journal of Engineering Education* 2010; 26(1):18–29.
19. Abiyev R, Ibrahim D, Erin B. Navigation of mobile robots in the presence of obstacles. *Advanced Engineering Software* 2010; 41:1179–1186.
20. LaValle SM, Kuffner JJ. Randomized kinodynamic planning. *International Journal of Robotics Research* 2001; 2(5):378–400.
21. S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.
22. Abiyev RH, Akkaya N, Aytac E. Control of Soccer Robots using Behaviour Trees. In Proceedings of the 9th *Asian Control Conference (ASCC)*, 2013, p.1-6.
23. Abiyev RH, Akkaya N, Aytac E. Navigation of Mobile Robot in Dynamic Environments. In Proceedings of the IEEE International Conference on *Computer Science and Automation Engineering (CSAE)*; 2012, v.3, 480-484.
24. Abiyev RH, Akkaya N, Aytac E, Günsel I, Çağman A. Improved Path-Finding Algorithm for Robot Soccers. *Journal of Automation and Control Engineering* 2015; 3(5):398-402.