FUZZY CONTROL OF AUTONOMOUS MOBILE ROBOT

Olumide OBE[1], Ioan DUMITRACHE[2]

Navigarea roboților mobili autonomi într-un mediu dinamic, nestructurat şi non-episodic este constrânsă de imprecizia, incertitudinea, inexactitatea, complexitatea şi imprevizibilitatea caracteristică senzorilor şi controlerelor roboților. În această lucrare, sunt investigate mai multe tehnici de navigație pentru robotul mobil Kephera într-un mediu total necunoscut, utilizând tehnica inteligentă fuzzy. Este dezvoltat un controler logic fuzzy (FCL) ce folosește diferite tipuri de funcții de apartenență pentru a naviga robotul. Sunt investigate efectele diferitelor tipuri de funcții membru în ceea ce privește performanța comportamentului robotului în sarcini cum ar fi evitarea obstacolelor, urmărirea unui drum/urmărirea peretelui, explorare şi mersul la țintă. Tehnica Gaussiană a funcției membru se dovedeşte a fi cea mai eficientă pentru navigarea robotului mobil.

Seamless navigation of autonomous mobile robots in an unstructured, non-episodic and dynamic environment is constrain by imprecision, uncertainty, unreliability, complexity and unpredictability issues that characterize robots' sensors and its controller. In this paper, we investigate navigation techniques for Kephera mobile robot using fuzzy logic intelligent technique. We develop Fuzzy logic controller (FCL) using different membership function types to navigate the robot. We investigate the effects of membership functions types on the performance of behaviours such as obstacle avoidance, path tracking/wall following, exploration, and target reaching. Gaussian membership function technique is found to be most efficient for mobile robot navigation.

Keywords: fuzzy logic, mobile robot, obstacle avoidance, sensors

1. Introduction

A considerable research in navigation of mobile robot in an uncertain and complex environment has received a considerable attention lately. Efficient algorithms like adaptive control and behavior-based control are the bedrock of the research in mobile robot navigation. Because of the difficulties involved in building precise path generating equation for unknown and complex environment, an alternative method to adaptive control is indispensable.

[1] PhD student, Faculty of Automatic and Computer Science, University POLITEHNICA of Bucharest, Romania, E-mail: oluobes@gmail.co
[2] Prof., Faculty of Automatic and Computer Science, University POLITEHNICA of Bucharest, Romania, E-mail: idumitrache@ics.pub.ro

Evolutionary computation provides an alternative design method that adapts the robot's behaviour without requiring a precisely specified model of the world. Its adaptive power enables the robot to deal with changes in the environment and to acquire a robust behaviour tolerating noisy behaviour and unreliable sensor information [1].

Soft computing is concern with the design of intelligent and robust systems, which exploit the tolerance for imprecision inherent in many real world problems. Soft computing techniques like fuzzy logic, genetic algorithm and neural network are considered for expressing the subjective uncertainties in human. Humans have a remarkable capability to perform a wide variety of physical and mental task without any explicit measurements or computations [2]. Human uses perceptions in performing various tasks which are described by propositions drawn from a natural language, in which the boundaries of perceived classes are fuzzy.

In this paper, a fuzzy-logic-based approach was selected since it posses the capabilities to deal with uncertainties. It has already been shown that fuzzy rule based systems (FRBS's) [3] are important tools for modeling complex problems.

Fuzzy logic systems represent knowledge in linguistic form, which permits the designer to define a highly abstract behaviour in an intuitive manner [2].Using fuzzy Logic framework, the attributes of human reasoning and decision making can be formulated by a set of simple and intuitive IF (antecedent)-THEN(consequent) rules, coupled with easily understandable and natural linguistic representations.

Fuzzy control approach has been used by many researchers in sensor-based navigational control of mobile robots. Applications ranges from the use of single output Sugeno (e.g. [7]) to multiple outputs Mamdani fuzzy logic, purely reactive fuzzy controller, e.g. [4], to the mixture of ''behaviours" like single-goal directness and reactive collision-avoidance, e.g. [5], [6] and [8]. Harisha et. al. [1] have worked with fuzzy logic reasoning to control Mobile robot on pre-defined Strip Path. The predefined path has varied turns, the fuzzy logic reasoning take care of speed to keep mobile robot in the defined path. The results were proved experimentally. Iwakoshi et. al. [9] have presented an evolutionary learning of robot behaviours by Fuzzy Classifier System (FCS).They have done simulations under eight different conditions to validate their technique. Lee and Wu [10] have proposed a fuzzy algorithm to navigate a mobile robot from an initial position to a target position in an unknown environment filled with many obstacles. Feasibility of their proposed method has been shown experimentally and by simulation. In [11], an extensive fuzzy behavior-based architecture for the control of mobile robots was proposed. The architecture has been implemented on a team of three soccer robots performing different roles interchangeably. The robot behaviors and roles are designed to be complementary to each other for collective behavior.

Dragoicea et. al. [12] have investigated into multi-behavioural model based autonomous navigation of mobile robots. They presented some aspects of mobile robot control by using fuzzy reactive behaviours in the frame of the so called subsumption architecture. They proposed approach that allows the possibility to implement simple behaviours in a fuzzy reactive multi-control system. Saroj et al. [13] investigated navigation techniques for several mobile robots as many as one thousand robots using fuzzy logic in a totally unknown environment. Fuzzy logic techniques using different membership functions were developed and used to navigate mobile robots. The result was tested in a simulated environment and it was found that fuzzy logic controller with Gaussian membership function is most efficient for multiple mobile robots navigation.

In this paper, we investigate the navigation of Khepera II mobile robot in a simulated KIKs environment with static and moving obstacles using different types of fuzzy logic controller (FCL).

2. Problem formulation

The objective of this work is to implement the fuzzy controller on Khepera II mobile robot [14] in order to control it for wall following, obstacle avoidance, target reaching and exploration behaviours.

The Khepera II robot is widely used especially among the academia around the world as a platform for various robotics experiments and applications.

3. Fuzzy logic and fuzzy logic control approach

In robotic design, it is more desirable to have a robot act more like a human being than a machine. Traditional logic is crisp and therefore inadequate for approximating the human decision making process. Fuzzy logic mimic human thinking process by using the entire interval between zero and one, and can therefore be used to closely represent human reasoning [1, 16]. The design process of a fuzzy logic system can generally be separated into the following stages: Fuzzification, Rule evaluation and deffuzification.

Figure 1 depicts the block diagram of Mamdani fuzzy logic based controller for the control of our mobile robot.

In fig. 2, the eight infra-red sensors on the Khepera II robot are shown according to our input design. The sensor readings are integer values in the range of [0, 1023]. A sensor value of 1023 indicates that the robot is very close to the object while a sensor value of 0 indicates that the robot does not receive any reflection of the infra-red signal. The readings are normalized to the range of [0, 1].

We define three linguistic variables representing relative distance between the robot and an object in its environment. These include: Left Distance LD, Right

Distance RD, and Front Distance FD. Since the robot rarely uses the back sensors because, the robot usually moves forward, we exclude back sensor readings in our design. The physical domains over which these linguistic values are defined are determined as:

LD = Max (S0, S1);

FD = Max(S2,S3);

RD = Max(S4,S5).

Where Si are the sensor values normalized within the interval [0, 1]. All these input linguistic variables have the same base variable length.

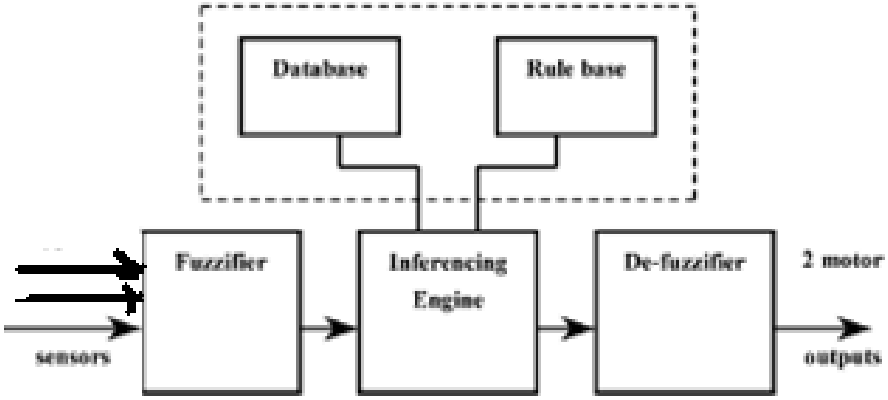

Fig. 1 Block diagram of FLC

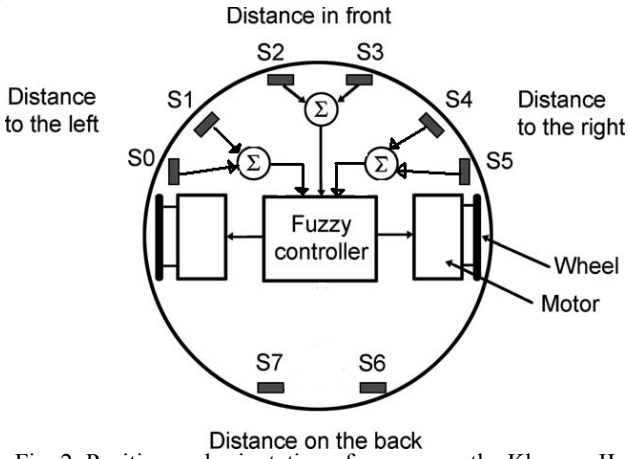

Fig. 2. Position and orientation of sensors on the Khepera II

We define some linguistic variables that represent the relative distance of the robot from various objects in its environment as measured by the infra-red sensors. These include:
Very Far-VF; Far-F; Medium-M; Near-N; Very Near-VN

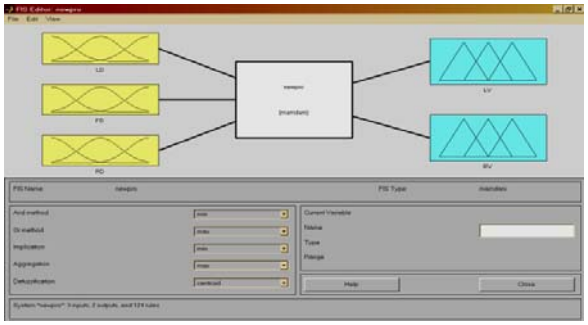

Fig. 3. Fuzzy inference table for Five Input-Five Output controller

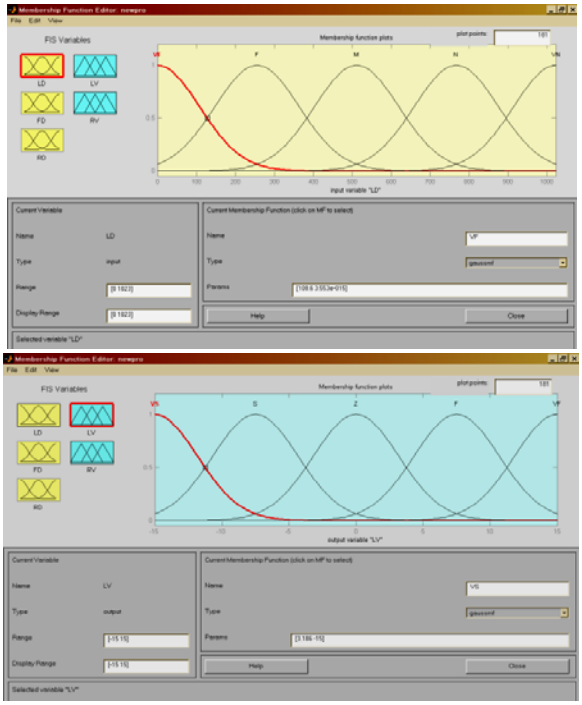

Fig. 4. Input and Output Gaussian membership function for Five Input-Five Output obstacle distance respectively

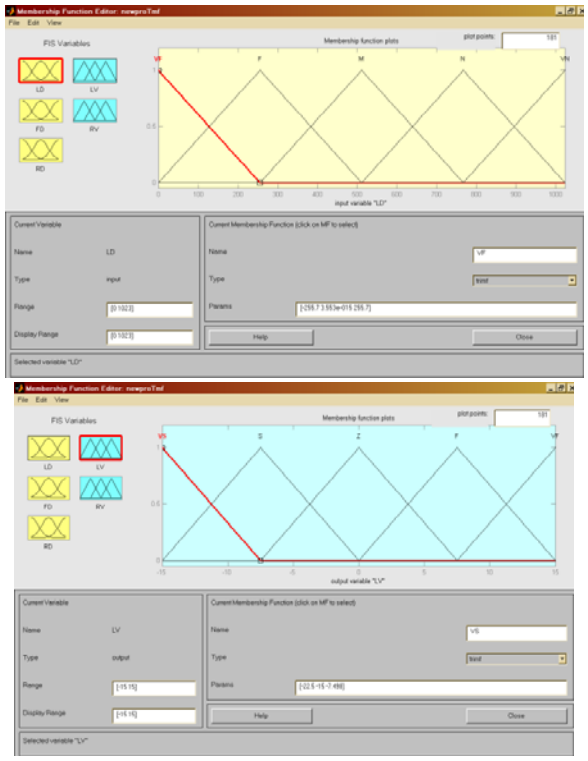

Fig. 5. Input and Output Triangular membership function for Five Input-Five Output obstacle distance respectively

3.1. Fuzzification

In the fuzzification stage, fuzzy inputs are sent into the fuzzy controller. The inputs are fuzzified using membership functions. The membership functions maps input space [0, 1023] of the sensor values into degree values of membership for each function [0, 1]. Two types of fuzzy membership functions are used to fuzzify the input data. These include Gaussian membership function as shown in Fig. 4, and Triangular membership functions as shown in Fig. 5. The Gaussian function is of the form:

$$\mu_{ij} = e^{-(x_j - a_{ij})^2/2b_{ij}^2} \tag{1}$$

where x_j is the input sensor value (normalized), a_{ij} is the center value and b_{ij} is the width of the corresponding membership function curve.

Once the inputs are fuzzified, the rules are applied to determine a response to the inputs.

3.2. Rule Evaluation

Based on the subsets, the fuzzy control rules are defined as follows using annotations in [2]:

$$\text{If}(LD \text{ is } LD_l \wedge FD \text{ is } FD_m \wedge RD \text{ is } RD_n)\text{Then } LV \text{ is } LV_{lmn} \wedge RV \text{ is } RV_{lmn} \quad (2)$$

Where LD, FD, and RD represents Left Distance, Front Distance and Right Distance respectively; and l=1-5, m=1-5, n=1-5 because LD, FD, and RD have five-membership function each.

From expression (2), two rules are derivable:

$$\text{If}(LD \text{ is } LD_l \wedge FD \text{ is } FD_m \wedge RD \text{ is } RD_n)\text{Then } LV \text{ is } LV_{lmn} \quad (3a)$$

and

$$\text{If}(LD \text{ is } LD_l \wedge FD \text{ is } FD_m \wedge RD \text{ is } RD_n) \text{ Then, } RV \text{ is } RV_{lmn} \quad (3b)$$

A factor W_{lmn} is defined for the rules as follows:

$$W_{lmn}=\mu_{LD_l} \wedge \mu_{FD_m} \wedge \mu_{RD_n} \quad (4)$$

The membership values of the left and right wheels velocities are given by:

$$\mu_{LV_{lmn}}(vel) = W_{lmn} \wedge \mu_{LV_{lmn}}(vel_{LV}) \quad \forall_{vel} \in LV \quad (5a)$$

$$\mu_{RV_{lmn}}(vel) = W_{lmn} \wedge \mu_{RV_{lmn}}(vel_{RV}) \quad \forall_{vel} \in RV \quad (5b)$$

The overall conclusion by combining the outputs of all the fuzzy rules for 5input-5output membership function can be written as follows:

$$\mu_{LV_{lmn}}(vel)=\mu_{LV_{111}}(vel_{LV}) \vee \ldots \vee \mu_{LV_{lmn}}(vel_{LV}) \vee \ldots \quad (vel_{LV}) \quad (6a)$$

$$\mu_{RV_{lmn}}(vel)=\mu_{RV_{111}}(vel_{RV}) \vee \ldots \vee \mu_{RV_{lmn}}(vel_{RV}) \vee \ldots \quad (vel_{RV}) \quad (6b)$$

3.3. The Fuzzy Rule Table

The fuzzy rules in this work are obtained heuristically bearing in mind that Khepera II mobile robot is a differential wheeled robot. There are 124 rules considered for the controllers. Some of the rules are shown in Table1 below.

Table 1

Fuzzy Rule Table

	INPUT			OUTPUT	
	SENSORS IF			MOTOR WHEELS THEN	
Rule no.	***LD***	***FD***	***RD***	**LV**	**RV**
1	VN	VN	VN	VS	VF
2	VF	VN	N	S	F
3	VN	VN	M	VF	Z
4	VN	VN	F	F	S
5	VN	VN	VF	VF	VS
6	VN	N	VN	VF	VS
7	VN	N	N	F	S
8	VN	N	M	F	Z
9	VN	N	F	VF	S
10	VN	N	VF	VF	VS
11	VN	F	N	VF	S
12	VN	M	VF	VF	Z
13	N	F	F	F	Z
14	M	F	N	Z	Z
15	F	M	N	Z	F
16	F	VF	N	Z	VF
17	N	VN	VN	S	VF
18	N	VN	N	F	S
19	N	VN	M	F	Z
20	N	VN	F	F	S
21	N	VN	VF	F	VS

3.4. Deffuzification

As shown in Fig. 3, we employ the Mamdani Fuzzy Logic reasoning method to compute the output of the fuzzy controller. The output linguistic variables for each fuzzy controller are:
VS - Very Slow; S - Slow; Z - Zero; F – Fast, and VF - Very Fast.

In the deffuzification stage, the results of the rule evaluation are translated into discrete outputs. This can be done in various ways, using different kind of algorithms. A common and easy approach is Center-of-Gravity (COG) method.

We compute the crisp values of the left and right wheels velocities of our robot using center of gravity (COG) method.

The general formula for the center of gravity method is:

$$\Delta U = \frac{\sum_{j}^{m} \mu_j \Delta u_j}{\sum_{j}^{m} \mu_j} \tag{7}$$

ΔU-Output varaiable, m-Number of rules, μ_j-is the grade of the j_{th} output of MF

ΔU_j-Output label for the value contributed by the j_{th} MF

4. Behaviours implementation

The approach in this work is behaviour-based and Fuzzy Logic is used as a reliable technique for behaviour fusion. In this work, the robot alternates among the following behaviours: Target reaching, Obstacle avoidance, Exploration, and Wall following. The fuzzy rules are designed to implement these behaviours. The rule base is designed to avoid dead end situation. Once the robot has received a command to start searching for the target position, the robot will try to locate the target while avoiding any obstacle along its path. On the other hand, the robot is set to explore its environment based on the time set and avoiding any obstacle on its path.

5. Robot simulation environment

KIKs simulation environment is used for this work. KiKS stands for "**K**iKS **i**s a **K**hepera **S**imulator". As the name implies, KiKS simulates one or more Khepera robots connected to the computer in a very realistic way, and runs under Matlab. Since KiKS accurately simulates motors, proximity/light sensors and even baud rate, we developed behaviours in the simulator (which usually runs at 200-300% of wall clock time) and expect the same behaviour once we implement it on a real Khepera II robot.

5.1. Simulator engine

KiKs simulates a Khepera or Khepera II robot connected via serial link to computer in a very realistic way. The simulation is based on publically available data from K-team [14], and performance of all simulated robot components (motors, sensors, camera) have been fine-tuned for countless hours to match those of real Kheperas.

6. Simulation results and discussions

The proposed fuzzy logic techniques have been simulated in different designed environments. Simulations were conducted using KIKs simulator. We investigate the performances of each fuzzy controller for four behaviours such as wall following, target reaching, obstacle avoidance and exploration (or wandering) as shown in sections 6.1 below:

6.1. Experimental Results

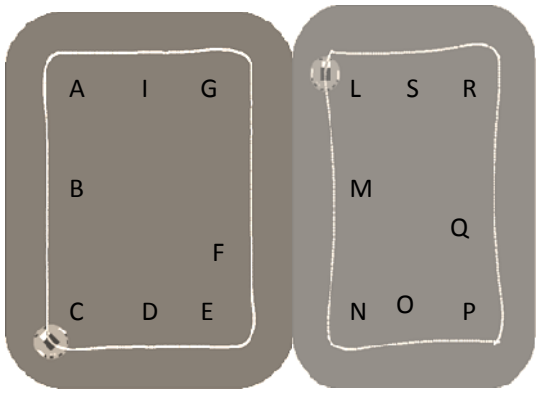

Fig. 7a. Wall Following and Obstacles Avoidance behaviours: Gaussian MF based and Triangular MF based controllers simulated performance results respectively

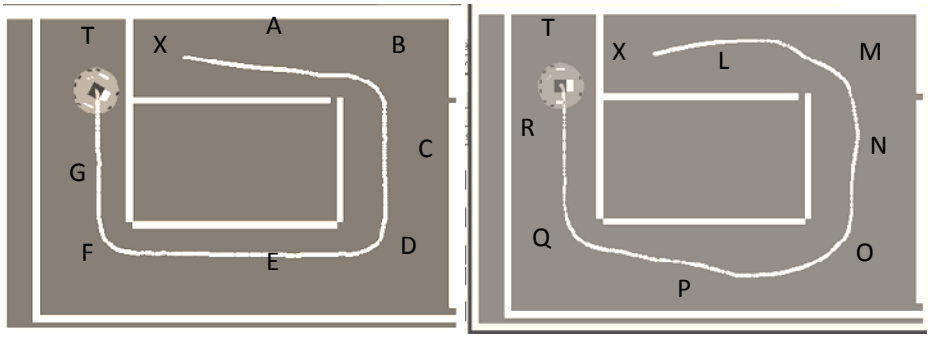

Fig.7b. Wall Following and Target Reaching behaviours: Gaussian MF based and Triangular MF based controllers simulated performance results respectively

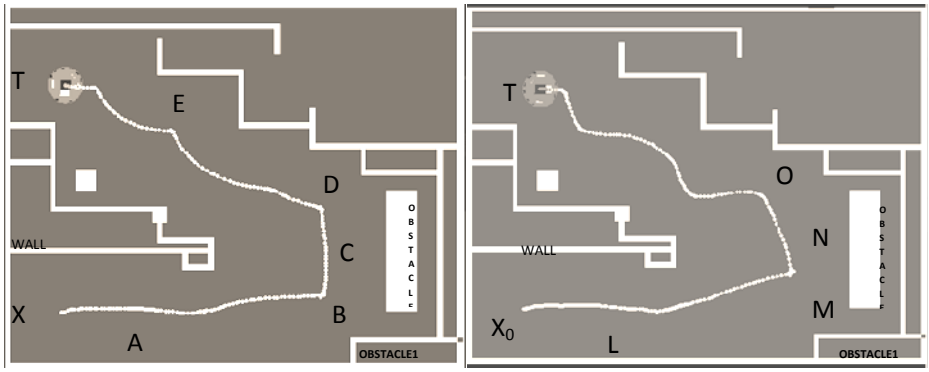

Fig. 8. Obstacle avoidance, target reaching & wall following behaviours: Path traced for Gaussian MF (28s) and Triangular MF (60s) respectively

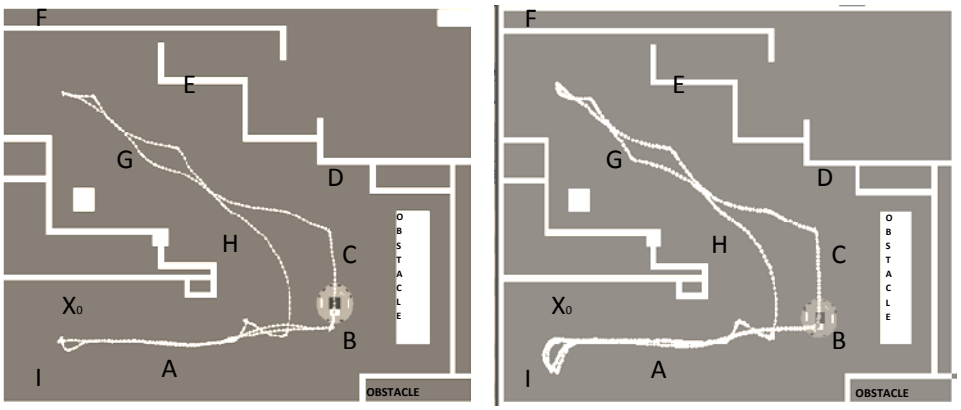

Fig. 9. Exploration Behaviour: Path traced by robot for Gaussian MF and Triangular MF based controllers respectively

6.2. Discussions and comparison between the two fuzzy controllers

Figs.7 (a, and b), 8, and 9 illustrates the simulated results of the path traced by the robot in various KIKs simulator environments. The two fuzzy controllers exhibited wall following, target reaching, obstacles avoidance and exploration behaviours successfully. The experiment validates the sufficiency of sensor-based fuzzy controller model in robot navigation.

In fig.7a, the first figure illustrates the path traced by the Gaussian based Fuzzy controller from the starting point A. Wall following behaviour was elicited from the initial point A up to point C where the robot needs to avoid colliding with the wall at right angle. At this very point C, another behavior called obstacle avoidance behaviour must be elicited in other to avoid collision with the wall. The robot was able to avoid the collision by eliciting the obstacle avoidance behaviour. Thus, behaviour switching is necessary to meet the challenges posed by the dynamics of the unknown environment. The same patterns of behavioural switching were observed each time the robot encounters obstacles (e.g. at points E, G, A and C) and when walls must be followed (e.g. paths C->D->E, E->F->G, G->I->A, and A->B->C).The robot followed the walls and avoided obstacles very smoothly. This was not the case with the Triangular MF based controller. The triangular membership based controller robot exhibited wall following (e.g. paths N->O->P, P->Q->R, R->S->L and L->M->N) and obstacle avoidance (e.g. at points L,N,P, and R) behaviours but not as smooth as in Gaussian based controller. The navigation followed the same real-time behavioural switching as in Gaussian based controller but the robot experienced difficulties especially at point P and L when obstacle avoidance behaviour was elicited. This is evident in the jittering of the robot at these points as shown by non-smooth curve traced by the robot at the points.

In fig.7b, the robot moved from the initial position X to the target position T using wall following behaviour. The Gaussian MF based controller executed wall following behavior from point X up to the target point T very smoothly in simulated time of 33s whereas, the Triangular MF based controller reached the same target point much later in simulated time of 60 s. The path traced by the triangular membership based controller robot is not as smooth as that of the Gaussian based controller, hence the time delay in reaching the target point.

Fig.8 illustrates the simulated result in another environment. The Gaussian MF based controller robot moved from the starting point X to the target point T following walls and avoiding obstacles along its path. The robot begins to execute the behaviours according to the rule base of the controller depending on the current context. The robot follows wall from point X with maximum velocity until it senses an obstacle immediately after point A, then it changes its behaviour to obstacle avoidance. This affects its trajectory. The robot changed to wall following behaviour after avoiding the obstacle with medium velocity until when it senses another obstacle at point B where it executes obstacle avoidance behaviour again. After avoiding the obstacle at point B, the robot follows the wall of the obstacle up to the point C where it switches to obstacle avoidance again in order to avoid collision with the wall of the arena at point D. The robot executes series of behavioural changes after point D, switching between wall following and obstacle avoidance behaviours depending on the current context posed by the

arena structure; and having target reaching behaviour as the primary goal until it arrives finally at the target position T. The robot arrive the target position T, in 28 s compared to the simulated time in triangular MF controller which is 60 s.

In fig. 9, we experiment with exploration behaviour where the robot traverses the arena until a set time elapses without having any point in the arena as target. We positioned the robot at initial point X_0 for each of the robots controller. The robots move up till point F and returned back to its initial point X_0. The robots stopped at the same set time, approximately at point B for both controllers. We observed different navigational difficulties at different points for both robots in traversing the arena. It was noticed that the Triangular membership based controller robot has more difficulties compared to Gaussian membership based controller robot at points F and I. At point I, the Triangular MF controller robot experienced severe jittering because of the difficulty in resolving dead-end problem when the robot arrived point I. The Gaussian membership based controller robot experience no much difficulty at resolving dead-end condition as shown in figure 9. The same difficulty was experienced at point F for both controllers but the Gaussian based controller easily resolves the dead-end situation by executing the appropriate rule(s) from the rule based.

6.3. Comparison between the fuzzy controllers

Experimental results from figures 7, 8, and 9 shows the superiority of Gaussian membership function based controller over the Triangular membership function based controller. In fig.7, the wall following behaviour took 30s for both controllers but, the path traced by the Gaussian MF based controller is smoother and the path length in pixel is lesser than in Triangular MF based controller.

In fig. 8, apart from the path length and smoothness in Gaussian MF based controller, the completion time shows the suitability of Gaussian MF based controller over Triangular MF controller for mobile robot control.

Fig. 9 further confirms the efficiency of Gaussian MF based fuzzy controller over the Triangular MF based fuzzy controller in terms of path length and smoothness

7. Conclusion

We have designed Fuzzy logic based controllers for Khepera II mobile robot. The controllers have been successfully used to simulate the mobile robot navigation in different environments. Two controllers were designed. These include Gaussian membership function based fuzzy controller and Triangular membership function based fuzzy controller. It has been demonstrated that by using the two controllers, the robot is able to follow wall, avoid obstacles, explore

its environment and reach target position. Comparative analysis of performance of the two controllers shows better performance of Gaussian membership function based fuzzy logic controller over the Triangular membership function based fuzzy logic controller.

R E F E R E N C E S

[1]. *S.K. Harisha, et. al.*, Fuzzy Logic Reasoning to Control Mobile Robot on Pre-defined Strip Path, Proceedings of World Academy of Science, Engineering and Technology, **Vol. 32**, August 2008,ISSN 2070-3740

[2]. *S. Pradhan, D. Parhi, A. Panda*, Fuzzy logic techniques for navigation of several mobile robots , *Applied Soft Computing*, **Vol. 9**, No. 1. (January 2009), pp. 290-304

[3]. *M. Jamshidi et. al*. Ed. Fuzzy Logic and Control: Hardware and Software Applications, Prentice Hall, Englewood Cliffs, New Jersey, 1993

[4]. *F.G. Pin and Y. Watanabe*, Driving a car using reflexive fuzzy behaviour. IEEE International Conference on Fuzzy Systems, pages 1425-1430, 1993

[5]. *S. Ishikawa*.,A method of autonomous mobile navigation by using fuzzy control, Advanced Robotics, 9(1):29-52, 1995

[6]. *E.H. Ruspini*, Fuzzy logic-based planning and reactive control of autonomous mobile robots. In IEEE International Conference on Fuzzy Systems, pages 1071-1076, 1995

[7]**.** *Y.H. Joo, H.S. Hwang, K.B. Kim, K.B. Woo*, Fuzzy system modeling by fuzzy partition and GA hybrid schemes, Fuzzy Sets and Systems 86 (3) (1997) 279–288

[8]. *H. Surmann et. al.*, A fuzzy System for Indoor Mobile robot Navigation, Proc. Of the fourth IEEE Int. Conf. on Fuzzy Systems, pp. 83-86, 20-24.03.1995, Yokohama, Japan.

[9]. *Y. Iwakoshi, T. Furuhashi, Y. Uchikawa*, A Fuzzy Classifier System for evolutionary learning of robot behaviors, Applied Mathematics and Computation 91 (1) (1998) 73–81

[10]. *T.-L. Lee, C.-J. Wu*, Fuzzy motion planning of mobile robots in unknown environments, Journal of Intelligent and Robotic Systems 37 (2) (2003) 177–191

[11]. *P. Vadakkepat, O.C. Miin, X. Peng, T.H. Lee*, Fuzzy behavior-based control of mobile robots, IEEE Transactions on Fuzzy Systems 12 (4) (2004) 559–564

[12]. *M. Dragoicea, I. Dumitrache, D. Cuculescu*, Multi-behavioural model based autonomous navigation of the mobile robots, International Journal, Automation Austria, Jg. 11,(2003) 1-20, ISSN 1562 – 2703

[13]. *K.P. Saroj, R.P. Dayal, K.P. Anup*, Fuzzy Logic Techniques for navigation of several mobile robots, Elsvier Journal of Applied Soft Computing, 9 (2009) 290 – 304

[14]. Khepera II, K-team, 2002, http://www.k-team.com

[15]. *F. Mondada , E. Franzi, A. Guignard*, The Development of Khepera. In proceedings of First International Khepera Workshop, Paderborn, December 10-11, 1999

[16]. *A. C. Nearchou*, Adaptive navigation of autonomous vehicles using evolutionary algorithms, Artificial Intelligence in Engineering, **vol. 13**, 1999, pp. 159-173.