Fuzzy Position Control Approach for Autonomous Robot Controller

Ali ÇETNKAYA*, Novruz ALLAHVERDİ+, A. Y. AFAGHANI−, Tuğba SARAY*
*Graduate School of Engineering and Natural Sciences, KTO Karatay University
Konya,TURKEY
aalicetinkayaa@gmail.com, tugba.saray@karatay.edu.tr
+Department of Computer Engineering, Faculty of Engineering, KTO Karatay University
Konya,TURKEY
novruz.allahverdi@karatay.edu.tr
−Department of Mechatronics Engineering, Faculty of Engineering, KTO Karatay University
Konya,TURKEY
yasser.afaghani@karatay.edu.tr

***Abstract*— In this study, the application of position control on a fixed track by an autonomous robot fuzzy controller approach is realized. For this purpose, a system is designed in which the real time data exchange can be done both for hardware and software. The embedded system (Atmel 2560 Integrity) is used in the hardware section to achieve the target attainment and position control with the fuzzy controller. DC motor, servo motor, ultrasonic sensor, Bluetooth module and optical sensor are controlled using this system. In addition, the desktop software is designed using the microcontroller software as well as Visual Studio 2015 platform using c# language. The coordinates (x, y) and path information sent by Altu robot (The name of the robot designed for this study) on the designed track in real time are processed with the fuzzy controller. The obtained data is displayed with the desktop software, position information, and robot motion map of Altu robot. Experiments using the fuzzy approach are also tested with the conventional control method and the results are compared. Thus, the fuzzy logic approach has proved that it can provide more precise control than the conventional approach.**

***Keywords*- Fuzzy controller, Position control, Autonomous robot**

I. INTRODUCTION

Autonomous robots have been seen extensively in many applications last decades, e.g. in the army, agriculture, transportation, and many other fields. These robots have moving mechanisms and equipped with sensors which allow the robots to perform different tasks. One of the important tasks is to move towards a specified position with a minimum error. Moreover, the robot should avoid the obstacles based on the information received from the environment. Those information can be obtained by ultrasound sensor, GPS, radar, laser, camera, or any other similar means.. The autonomous robot should be able to analyze the data coming from these sensors and process them using control software, and thus, creating best path and start time. While the robot is moving along its normal path, the coordinates are updated taking into account the barriers along the road.

Normally, it is desirable to change the system parameters over time in order to provide an appropriate and dynamic control. Therefore, the fuzzy logic controller, which is a control method that can exemplify human thinking ability and knowledge in the control system, is an effective solution.

In this scope, many researchers have been applied fuzzy logic approach for controlling autonomous robots [2]–[14]. To control the robot and find suitable path, at least three different information should be available: the current position, target position, and the angle which oriented the robot.

In this work, the fuzzy logic control design process and the phases of execution that process are investigated on an autonomous robot that designed by author's team for the work purpose and named as Altu [1]. In addition, the conventional control method is examined on the same robot to show the advantage of the proposed control method. A fixed track environment is created for testing Altu robot with different methods. To track the movement of the robot on the path and to measure the error margin, c#-based software is implemented. The obtained results are checked using Matlab application for the accuracy verification.

The organization of the paper is as follows: In Section II, the structure and track of the designed robot are described. Section III describes fuzzy control, fuzzification, defuzzification, and fuzzy rules. Section IV presents some experimental results of controlling Altu robot. Finally, Section V is the conclusion part.

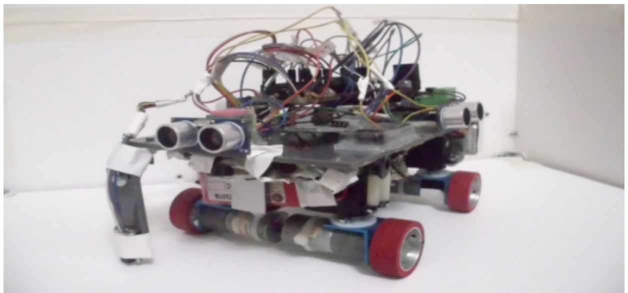
II. STRUCTURE AND TRACK OF DESIGNED ROBOT

A. Structure of the Robot and its Track

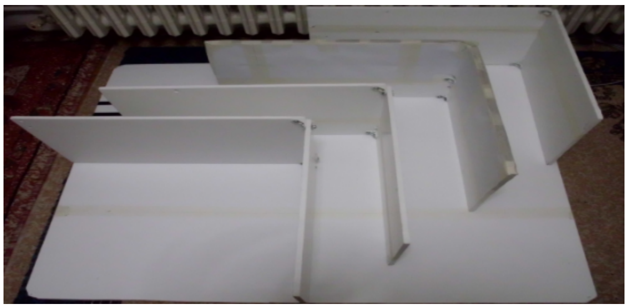
The robot body is made of a Plexiglass with a length of $140mm$ x $225mm$ and a thickness of $5mm$ as illustrated in Fig. 1(a). On the upper part of the body the following components can be found:

- Four ultrasonic sensors (H04).
- One wireless point (HC05).
- One Schmitt trigger circuit (74HS14).
- One microcontroller card (ATMEGA 2560).
- One regulation circuit (LM2576-5V).

On the lower part of the body:



(a) Designed robot



(b) Track of Altu robot

Fig. 1. Altu Robot hardware and its track [1].

- One Li-Po battery (11.1V 800mA).
- One Li-Po battery (7.4V 1050mA).
- One DC motor driver card.
- Four ON-OFF buttons and the charger section.
- Four servo motors (12 V 500RPM).
- Four silicon wheels;
- One regulated circuit (7805 R 5V).
- Two infrared sensors (QRD1114).

The interaction of Altu robot with the environment is provided by ultrasonic sensors. The working area of the sensor is up to $4m$. In the case of Altu robot, the track is limited to $1m^2$, and thus the sensor is suitable for this work. The track is made of strap and the edges of the track are covered by four L-shaped pieces as shown in Fig. 1(b).

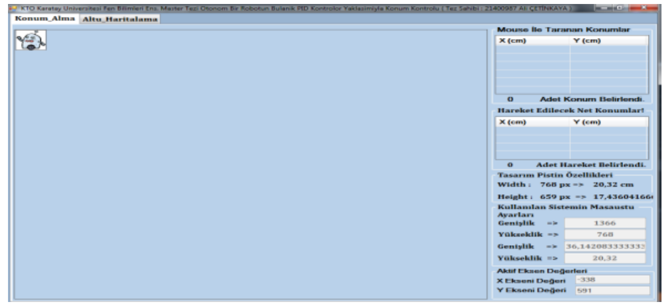
B. C#-Based Software

There are two parts of the desktop software created using C# language on Visual Studio 2015 platform. The first part is position acquisition part, where the position is obtained by the mouse as shown in Fig. 2(a). The second part is the mapping part where the operations are performed with the obtained position as shown in Fig. 2(b).

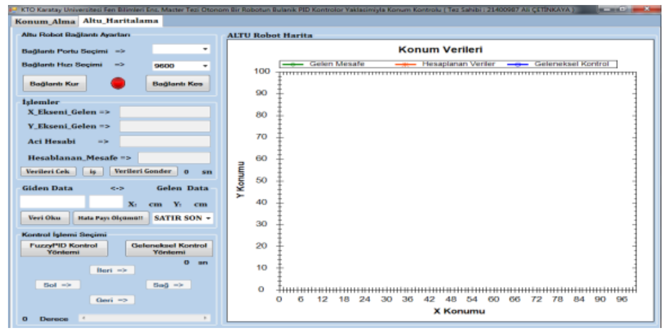
III. ROBOTS CONTROL AND SOFTWARE STRUCTURE

The block diagram of the fuzzy control system designed for this study is shown in Fig. 3. Three fuzzy inputs (X position, Y position and angle) and one fuzzy output (distance data) are used in the system. Unlike fuzzy control, the traditional control system usually works with 5 inputs data (forward, backward, left, right, and angle).

The distance calculation diagram of the designed system is shown in Fig. 4. Where, the symbol A shown on the diagram represents the starting position of Altu robot, the symbol V represents the target position of the robot, and the symbol T represents the intersection of the A and V points on the X-axis.



(a) Position acquisition part



(b) Mapping part

Fig. 2. Altu Robot Software.

The symbol "t" represents the distance between the starting position and target position of Altu robot, and the symbol "a" represents the distance between the target position and the intersection point T on the X-axis. The Θ symbol represents the angle change used to reach the target position from the position of the robot, and the α symbol is represented by $90 - \Theta$.

A. Fuzzy Controllers

The fuzzy unit of Altu robot consists of input and output variables, fuzzification, fuzzy inference and defuzzification units. The input variables of the designed controller are:

- X Position (cm) = the position of the robot on the x-axis within the range 0 and 100 cm.
- Y Position (cm) = the position of the robot on the y-axis within the range 0 and 100 cm.
- Angle (°) = the angle information relative to the axes of Altu robot.

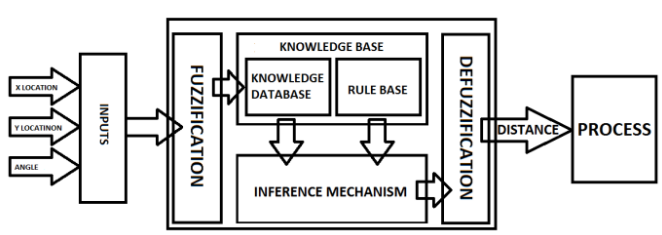


Fig. 3. Blok. diagram of the fuzzy system.

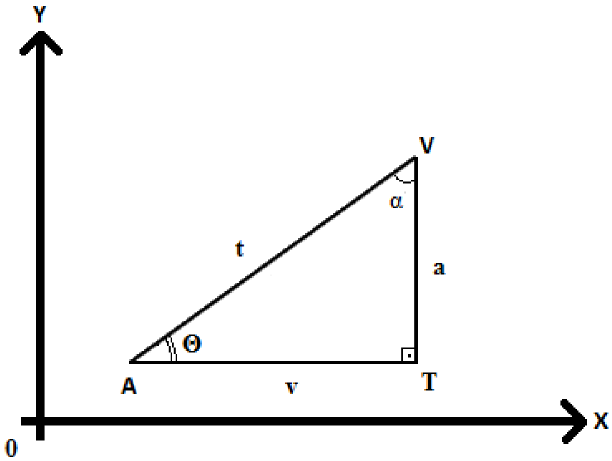


Fig. 4. The distance calculation description.

On the other hand, the output variable is:

- Distance (cm) = the total travelling distance which affected by the angle data and position data. The change range for this parameter is $0 < Distance < 140cm$.

1) Fuzzification: In this process, exact numerical input and output variables are expressed as fuzzy ratios. Variables X and Y are expressed separately using seven fuzzy subsets, the angle variable is expressed with five fuzzy subsets, and the output variable is expressed with five fuzzy subsets. The graphical representation of those fuzzy sets are shown in Fig. 5(a)-Fig. 5(d).

2) Fuzzy Inference: Fuzzy rules are the most important part of fuzzy control. In this part, the Altu robot fuzzy knowledge base and decision making mechanism are created. There are many techniques available for creating fuzzy inference. In this study max-min (Mamdani) method is adopted.

3) Determination of Fuzzy Rules: In fact, the number of fuzzy rules in which input variables and output variables are related is 245 rules. However, there are situations where some rules cannot be applied in the reality. Those situations can be removed from the rules table.

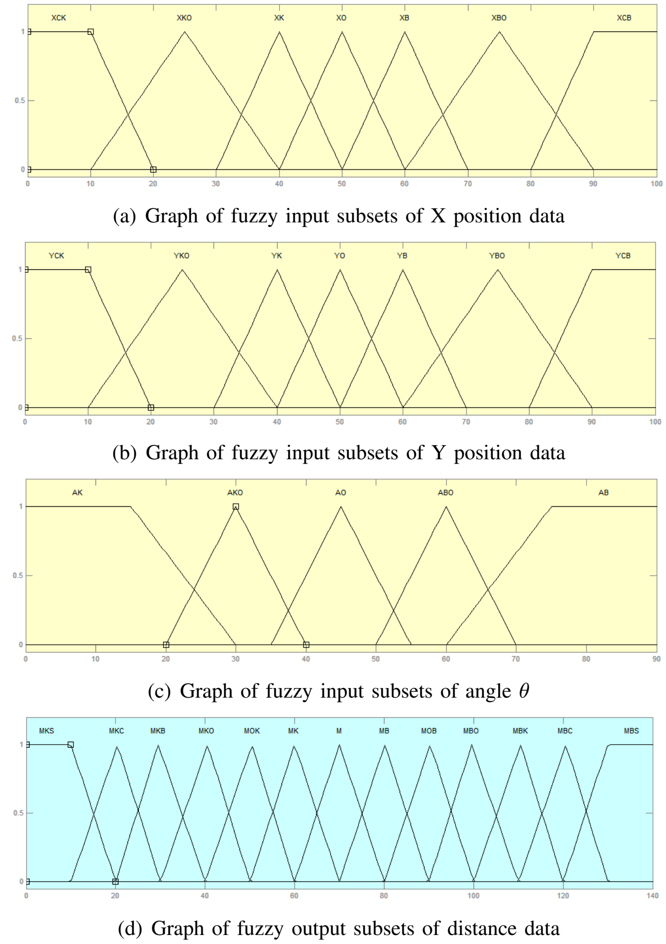
In this work, 141 fuzzy rules are defined that relate the verbal expression and the fuzzy expressions of each coordinate (x, y), angle, and the distance variables. Some of those rules are illustrated in TABLE I.

4) Defuzzification: The information obtained from the output of the subroutine fuzzy logic controller is the fuzzy knowledge. This knowledge has to be transformed into an exact value. There are several methods to do that transformation. In this work, the weight average method is used to identify the single and exact values.

TABLE I

SOME FUZZY RULES

Rule No.	Rule Description
Rule 1	If XCK, YCK, and AK, then the distance value is MKS
⋮	⋮
Rule 71	If XO, YCK, and AB, then the distance value is MOK
⋮	⋮
Rule 141	If XCB, YCB, and EU, then the distance value is MBS



(a) Graph of fuzzy input subsets of X position data

(b) Graph of fuzzy input subsets of Y position data

(c) Graph of fuzzy input subsets of angle θ

(d) Graph of fuzzy output subsets of distance data

Fig. 5. Input and output sets of fuzzy rules.

IV. EXPERIMENTS

Many experiments have been done to evaluate the methodology feasibility. The experiments have proved that the control of the robot using the fuzzy position control approach has moved the robot easily in the tested environment and reached to the desired targets. Different scenarios are created in order to be able to examine the results in details and to monitor them on the desktop software.

The fuzzy control algorithm used by Altu robot aims to minimize the power consumption of the robot. Thus, in each experiment, the robot is assigned with specific start and target positions, then, when the robot completes the tasks and reached at the destination, the error ratios of the arrival position data are calculated. In addition the the total time of the motion is calculated to evaluate the method functionality.

A. First Experiment

In this experiment, a "Fuzzy Control" method is applied. The starting position of Altu robot track is defined as $(X =$

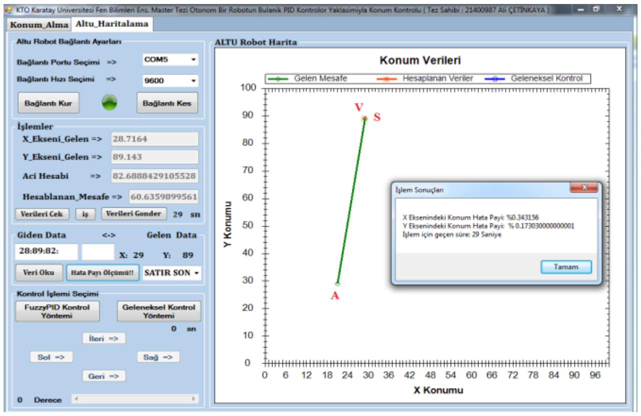


Fig. 6. First Exp. : Resulted map and error ratios on both axes.

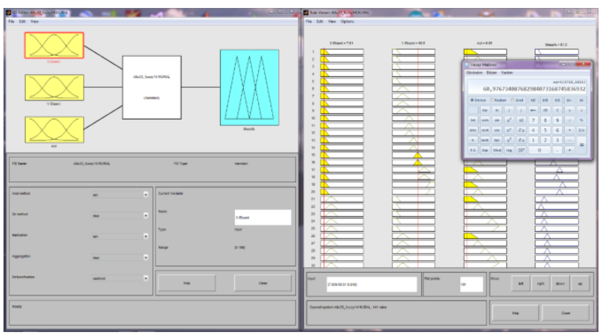


Fig. 7. First Exp. : Results using Matlab fuzzy tool box.

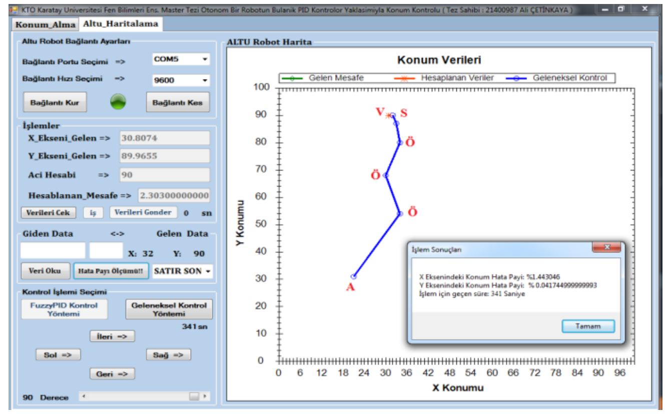


Fig. 8. Second Exp.: Resulted map and error ratios on both axes.

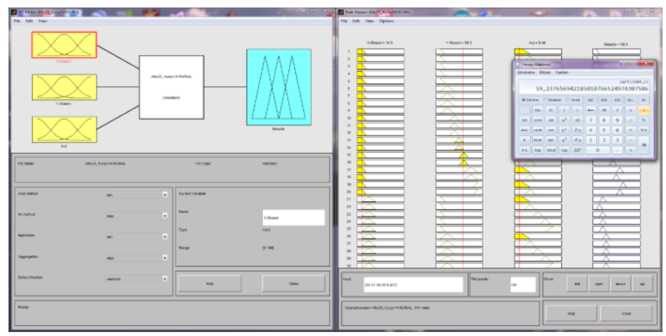


Fig. 9. Second Exp. : Results using Matlab fuzzy tool box.

$21cm, Y = 29cm$), and the target position is determined as ($X = 28cm, Y = 89cm$). By selecting the "Read Data" button in the interface, the robot starting position A is illustrated as a green diamond-shape. The target position V is determined by pressing the "Pull Data" button and the V point will appear in the shape of an orange star as illustrated in Fig. 6.

It can be noticed that the robot has reached its target position (V). Then, the "Fault Share Measurement" button is pressed for checking the accuracy of reached position. Thus, "Operation Results" window is opened which shows the error margin on X-axis, the error margin on Y-axis, and the total motion time from point "A" to point "V". The final position obtained on the basis of the measurements made on the robot is indicated by the "S" letter in the map as shown on the right side of Fig. 6.

As a result, the robot has moved 60.53 cm, and the error margins on the X and Y axes are $1cm$ and $0cm$, respectively. Based on these results, the error ratios on the X and Y axes are calculated as $\%0.345156$ and $\%0.17303030$, respectively. The total motion time for this experiment has taken $29sec$.

It is observed that the same results are obtained when using the Matlab fuzzy tool box with the same input data as shown in Fig. 7.

B. Second Experiment

In this experiment, a "Traditional" method is implemented, where the starting position of the robot on the track is set as ($X = 21cm, Y = 31cm$) and target position is ($X = 30cm, Y = 89cm$). When the "Read Data" button is pressed, the coordinates of the robot's start position is presented as blue circles ("A" point) while the target position ("V" point) is presented on the map in the shape of an orange star when the "Pull Data" button is pressed. Fig. 8 showing the aforementioned points. After accomplishing the motion and reaching at the target position, the "Fault Share Measurement" button is pressed to get the error ratios on the X and Y axes which are $\%1.443046$ and $\%0.04174$, respectively. In this experiment, the total traveling distance is $63.83cm$ and the motion time is $341sec$. It is clear the big difference between the motion time of the first experiment using fuzzy control and this experiment which uses the traditional control method.

During the movement, the robot position data is obtained by the system and marked on the map using the symbol "Ö". The symbol "S" indicates the last real position of the robot. Again by using the Matlab fuzzy tool box with the same input data, the same results are obtained as shown in Fig. 9.

C. Third Experiment

Here, the "Fuzzy Control" method, as in the first experiment, is applied with different input values. Where, the starting position is set as ($X = 24cm, Y = 33cm$), and target position as ($X = 75cm, Y = 85cm$). By applying the same procedure as in the first experiments, the point "A",

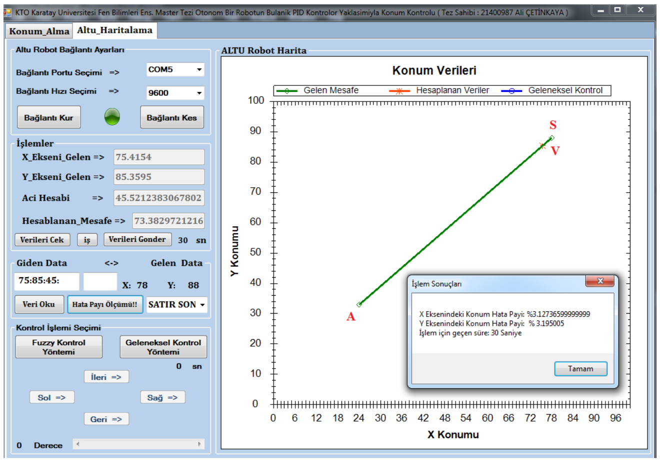


Fig. 10. Third Exp.: Resulted map and error ratios on both axes.

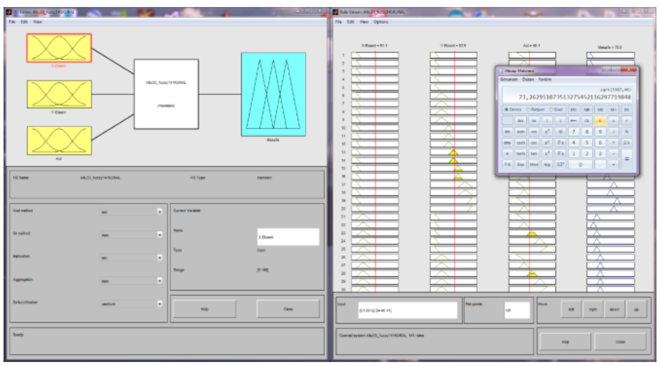


Fig. 11. Third Exp.: Results using Matlab fuzzy tool box.

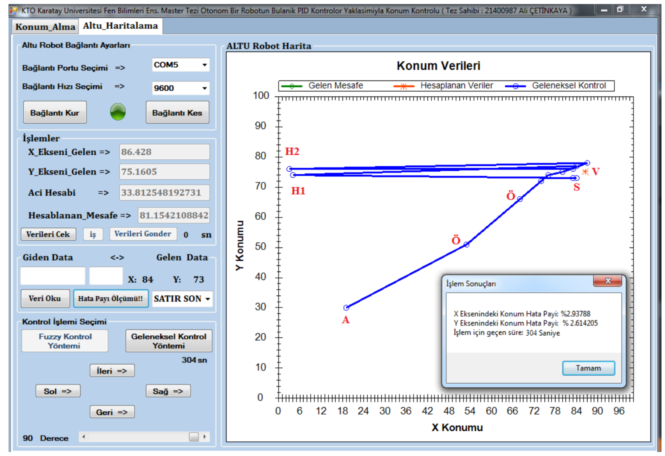


Fig. 12. Fourth Exp.: Resulted map and error ratios on both axes.

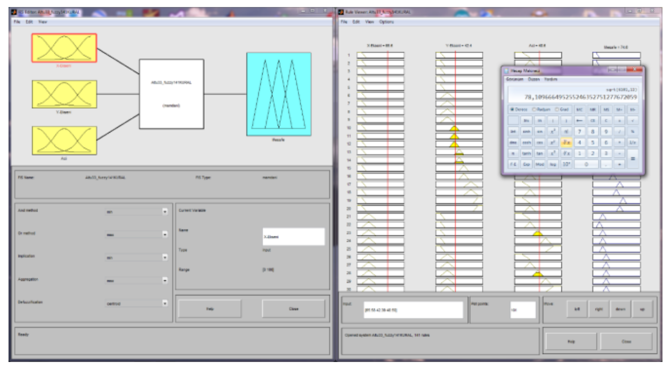


Fig. 13. Fourth Exp.: Results using Matlab fuzzy tool box.

point "V", and point "S" are obtained as shown in Fig. 10.

The robot is this experiment has moved $77.1cm$, the error margins on both X and Y axes are $3cm$. And the error ratios on the X and Y axes and time motion are calculated and they are $\%3.127365$, $\%3.195005$, 30 seconds, respectively. Here, it is noticeable the motion time is quite short as well. Again, by using Matlab fuzzy tool box, the both results are same as shown in Fig. 11.

D. *Fourth Experiment*

"Traditional" method is implemented again, as in the second experiment, with different values. In this experiment, the starting position is $(X = 21cm, Y = 33cm)$, and target position is $(X = 86cm, Y = 75cm)$. It can be seen the point "A" and point "V" in Fig. 12 where the path is a bit noisy. In this case, the "Fault Share Measurement" button should be pressed for acquiring the accuracy of the arrival position. From the "Operation Results" window, the error margins on the X and Y axes, and the measurement period of the motion of the robot from the point "A" to the point "V" point indicated on the map. For the sake of the correctness of the path information, the position data of the robots are taken during the motion and displayed as "Ö" points on the map. The "S" sign indicates that the measurements made on the map of the robot are the final position. Points indicated as "H1" and "H2" on the map are referred to the error of the measurements taken from the ultrasonic sensors while the robot is moving.

The final results of this experiment are the traveling distance which is $97.43cm$ with the position error of $2cm$ on the X-axis and $2cm$ on the Y-axis. Based on these results, the position error margin at X-axis is $\%2.93788$ and at Y-axis is $\%2.614205$. The motion time is measured as $304sec$. The experiment has been checked with Matlab as well and the results are same and shown in Fig. 13.

V. CONCLUSIONS

In this study, the "Fuzzy Approach" and the "Traditional" methods for controlling the Altu robot autonomously have been designed and applied practically. Both methods have been tested and the results have been compared in the experiment section.

The Altu robot has been designed by the author which is equipped with ultrasonic sensors used in the system detect the positions required for reaching the target on a fixed track.

The experimental results have proved that applying the fuzzy approach method has decreased the motion time significantly compared with the traditional method. With the fuzzy approach, the motion time has last to $29-30sec$, while with the traditional approach took $300 - 341sec$. Thus, the fuzzy control method seems to work ten times faster than traditional control method.

As future work, digital electronic compass can be used to determine the direction of the robot. To improve the communication speed between the robot and the station, a unique network can be established by using the Wi-Fi module instead of using the Bluetooth module which is used in this study. To get more accurate results, various artificial intelligence methods can be applied. Moreover, using robot vision system can increase the precision of the mapping process.

ACKNOWLEDGMENT

The authors thank the KTO Karatay University for supporting this work.

REFERENCES

[1] Cetinkaya A., *Otonom bir robotun bulanık kontrollör yaklaşımı ile konum kontrolü*, Y.L. Tezi, Fen Bilimleri Enstitüsü, KTO Karatay University, 2017.

[2] Erenoğlu I., *Akıllı melez bulanık PID kontrolör yapısı*, Y.L. Tezi , Fen Bilimleri Enstitüsü, İ.T.Ü., 2006.

[3] Civelek C., *Bulanık PID kontrollör ile türbinin hatve açısının Kontrolü*, Y.L. Tezi, Fen Bilimleri Üniversitesi Kırıkkale Üniversitesi, 2013.

[4] Güzay C., *Bulanık PID kontrölleri için birleştirme operatörüne dayalı yeni bir öz-ayarlama yöntemi tasarımı*, Y.L. Tezi, Fen Bilimleri Enstitüsü, İ.T.Ü., 2014.

[5] Hacimurtazaoğlu M., Allahverdi N., *Controlling the Helicopter Takeoff and Landing with Fuzzy Logic System*, International Conference on Advanced Technology Sciences, ICAT'16, Rome, Italy, 2016 .

[6] Kabasakal O., *Bulanık PID kontrolörleri için çevrim içi kural ağırlıklandırma yöntemi*, Y.L. Tezi, Fen Bilimleri Enstitüsü, İ.T.Ü., 2012.

[7] Sağlam G., *Hiyerarşik Bulanık mantık PID kontrolörleri*, Y.L. Tezi, Fen Bilimleri Enstitüsü, İ.T.Ü., 2007.

[8] Wolfer J., George A.C., *Fuzzy logic control for robot maze traversal: an undergraduate case study*, World Congress on Computer Science, Engineering and Technology Education, March 19 - 22, São Paulo, BRAZIL, 2006.

[9] Boufera F., et. al., *Mobile robot navigation using fuzzy limit-cycles in cluttered environment*, I. J. Intelligent System and Applications, 07, 12-21, 2014.

[10] Okatan A., Gökalp G., *Obstacle avoidance and fuzzy control of a mobile robot*, International Conference on Informatics (ICI), 2004.

[11] Arda M., Güllü A., Kuşçu H, *Bulanık Mantık yönteminin PID denetleyici performansına etkisi*, 16. Ulusal Makine Teorisi Sempozyumu, Erzurum, 2013.

[12] Ying H., *Theory and application of a novel fuzzy PID controller using a simplified Takagi-Sugeno rule scheme*, Information Sciences, 123, 281-293, 2000.

[13] Dogan H. Kaplan K., Kuncan M., Ertunç M.H., *Süspansiyon Sistemi kontrolünde PID ve Bulanık Mantık yaklaşımları*, Otomatik Kontrol Ulusal Toplantısı, TOK'2015, Denizli, 2015.

[14] Ryu C., Kwak S., Choi B., *Design of Fuzzy Logic Systems for Position and Balance Control of Segway Mobile Robot*, International Conference on Advanced Technology Sciences, ICAT'14, Antalya, Turkey, 2014.