Searching for Goal by Mobile Robot with Collision-Free Motion in Unknown Environment

Prof Dr. Turki Y. Abdalla
Department of Computer Engineering
University of Basrah,
Basrah, Iraq

Seaar J. Al_Duboni
Department of Computer Engineering
University of Basrah,
Basrah, Iraq

Abstract

Obstacle avoidance and path planning are from the most important problems in mobile robots, especially in unknown environment . In this paper, we proposed an approach for mobile robot navigation combining path planning and obstacle avoidance. Methods such as obstacle avoidance are inspired from the nature, and have been developed by fuzzy logic to train an intelligent robot in unknown environment. The model of the robot has two driving wheels and the linear velocity and azimuth of the two wheels are independently controlled using PID controller. Inputs are obtained from ultrasonic sensors mounted on it.

Key words: Mobile Robot, Trajectory-Tracking , Fuzzy logic, Obstacle Avoidance, Navigation**.**

I. Introduction

An intelligent mobile robot is supposed to be a type of autonomous mobile vehicle that is able to independently plan its own route and navigate through obstacles to reach its specified destination. There is no human input or intervention. Applications range among civilians (automatic cars), industrial (co-operative mobile robots in factories), military (unmanned vehicles to destroy enemy target) and fun (soccer player robots) usages. A simplified schematic model of a mobile robot controller is depicted in Fig.1. The decision unit module acts as the brain of the robot deciding what to do, according to the robot goal and also the robot perception from the environment. It then sends a planning request to the path planner. The path planner should determine the nearest route in terms of distance or time to the target which is free from collisions. The motion controller is responsible for following the suggested plan by the path planner, according to the mobile robot low level (physical) motion limitations as closely as possible. In this paper we concentrate on the path planner and motion controller modules, and call them "robot navigator". Many classical methods have been reported in the literature such as genetic algorithm based avoidance of a mobile robot in a given search space [1], Neuro - Fuzzy controller for mobile robot navigation in [2],

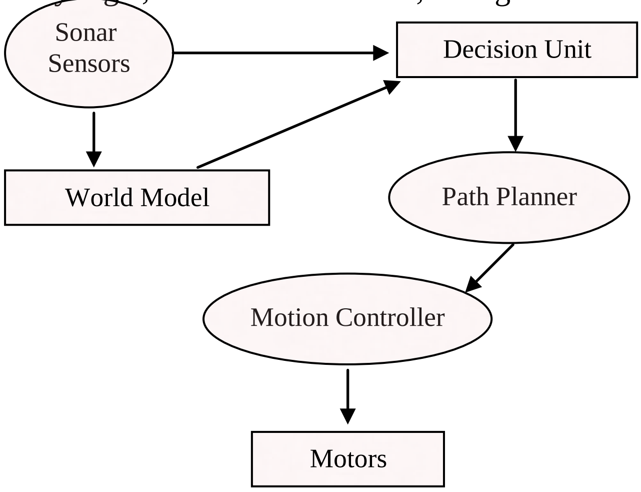

Fig.1: The simplified schematic model of a mobile robot controller

navigation the robot by using a sensor network embedded in the environment [3] or by using neural networks-based technique [4]. The paper is organized as follows: In Section I: Introduction. In Section II: The mathematical model of the robot. In Section III: Searching for Goal Point with Collision-Free Motion in an Unknown Environment. Simulation Results is presented in Section VI. Section V for Conclusions.

II. Modeling of the Mobile Robots

A. Dynamic Model

Mobile robots have received a great deal of research in recent years. A significant amount of research has been published in many aspects related to mobile robots. Most of the research is devoted to design and develop some control techniques for robot

motion and path planning[5]. A large number of researchers such as [6-8] have used kinematic models to develop motion control strategy for mobile robots . However, dynamic modeling of mobile robots is very important, as they are designed to travel at higher speed and perform heavy duty work. This paper uses dynamic model and propose a control strategy for wheeled mobile robot .
The mobile robot considered here is shown in Fig.2. It consists of a vehicle with two driving wheels mounted on the same axis, and a front passive wheel for balance. The two driving wheels are controlled independently by motors. Let the mobile robot be rigid moving on the plane.

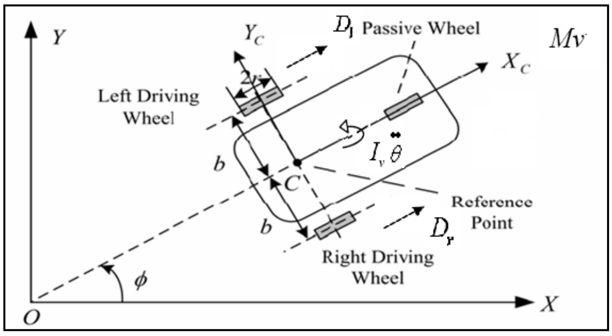

Fig.2: Model of mobile robot

We assume that the absolute coordinate system OXY is fixed on the plane. Then, the dynamic property of the robot is given by the following equation of motion [9]:

$$I_v \ddot{\theta} = D_r l - D_l l \qquad \dots (1)$$

$$M \dot{v} = D_r + D_l \qquad \dots (2)$$

For the right- and left-wheels, the dynamic property of the driving system becomes

$$I_w \ddot{\phi}_i + c \dot{\phi}_i = k u_i - r D_i \quad \dots (3)$$

$$(i = r, l)$$

Where
I_v: moment of inertia around the C.G. of robot , M : mass of the robot
D_ℓ, D_r: left and right driving forces
l : distance between left and right wheel and the c.g. of robot , θ: azimuth of the robot

v : velocity of the robot, I_w: moment of inertia of wheel, c : viscous friction factor
k : driving gain factor, r : radius of wheel
ϕ_i: rotational angle of wheel, u_i : driving input
On the other hand, the geometrical relationships among variables θ, v and ϕ_i are given by

$$r \dot{\phi}_r = v + l \dot{\theta} \qquad \dots (4)$$

$$r \dot{\phi}_l = v - l \dot{\theta} \qquad \dots (5)$$

From these equations, defining the state variable for the robot as

$$x = \begin{bmatrix} v & \theta & \dot{\theta} \end{bmatrix}^T$$

The manipulated variable as $u = [u_r, u_l]^T$, and the output variable as $y = \begin{bmatrix} v & \theta \end{bmatrix}^T$ yields the following state equation:

$$\dot{x} = Ax + Bu \qquad \dots (6)$$

$$y = Cx \qquad \dots (7)$$

$$A = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & a_2 \end{bmatrix}, \quad B = \begin{bmatrix} b_1 & b_1 \\ 0 & 0 \\ b_2 & -b_2 \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$ Where

$$a_1 = -\frac{2c}{Mr^2 + 2I_w}, \qquad a_2 = -\frac{2cl^2}{l_v r^2 + 2I_w l^2}$$

$$b_1 = \frac{kr}{Mr^2 + 2I_w} \qquad b_2 = -\frac{krl}{l_v r^2 + 2I_w l^2}$$

To simulate the mobile robot model, we use the State-Space block in Matlab environment with 4th order Runge-Kutta-Gill method with an integration step 1[ms]. The velocity and azimuth of the robot are controlled by manipulating the torques for the left wheels (U_ℓ) and the right wheels (U_r) .The motion of the mobile robot is controlled by PID controller designed and tested in[10] . The mobile robot parameters used in the simulation are given in Table1[9].

Table.1: values of parameters

Parameter	Value	Unit
I_v	*10*	$Kg.m^2$
M	*20*	*Kg*
L	*0.3*	*M*
I_w	*0.005*	$Kg.m^2$
C	*0.05*	*Kg/s*
R	*0.1*	*M*
K	*5*	-

B. Simulation of Mobile Robot

Fig3 illustrates the mobile robot used in this work, it is equipped with two driving wheels independently (left wheel and right wheel) and one freewheeling castor provides stability (front wheel) used to support the robot. Two encoders fixed on axels of left wheel and right wheel used to determine the current position of mobile robot. This independent drive system not only gives mobile robot capability to move in a straight line, and perform turns, but this system also allows the robot to have a zero turning radius feature. This feature allows mobile robot to turn directly around its center without backward motion. Matlab was taken as the mobile robot platform because it is a high featured and user friendly environment for technical computing such as graphical user interface building, simulation, algorithm development, and graphical presentation of collected data [11]. There are 16 Ultrasonic

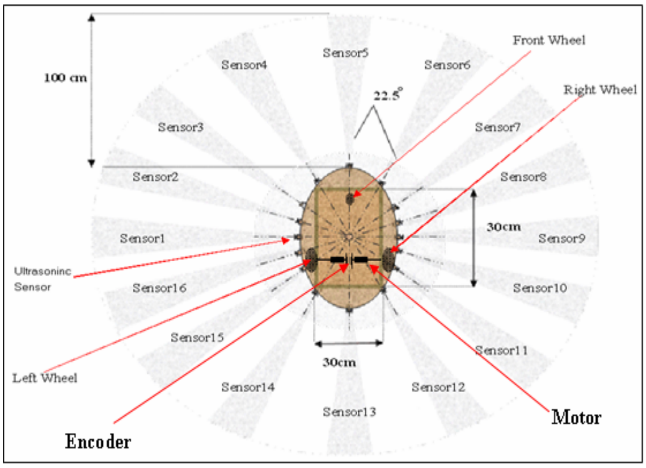

Fig.3: The mobile robot platform

Sensors (USs) distributed around the robot body. The angle between sensors is 22.5˚. The number of sensors and their arrangement are selected after several simulation experiments. In order to give the mobile robot an ability to navigate completely through forward or back move. The 16 USs are divided into two groups:

- Front group has left side (US1 to US5) and right side (US5 to US9).
- Back group has left side (US9 to US13) and right side (US13 to US1).

Every sensor (drive wheels Encoder and Ultrasonic sensors) measurement can be corrupted by white noise.

In this paper, the mobile robot is simulated by using S-Function block in Matlab V-7.6.0 (R2008A), especial S-Function block was written to create a whole mobile robot (platform and its ultrasonic sensors with environment) for online simulink. The mobile robot has three degrees of freedom that are represented by a posture $P_c = (X_c, Y_c, \theta)$, where (X_c, Y_c) indicate the spatial position (current position) of the mobile robot in the global coordinate system and θ (Azimuth) is the heading angle of the robot. The inputs into simulation block of mobile robot are θ, X_c, Y_c, X_g, Y_g and time, where (X_g, Y_g) represent the coordinate of target position, while the outputs of the simulation S-Function block are:

- **Left_Distance**: The shortest distance between mobile robot and left obstacles. Left side sensors in front group (US1 to US5) are responsible for measuring this distance for forward movement of mobile robot, while right side in back group sensors (US13 to US1) are responsible for backward movent.
- **Left_Angle (θ_{ol}):** Angle between heading angle of the robot and left obstacle, measured in degree.
- **Right_Distance**: The shortest distance between mobile robot and right obstacle, right side sensors in front group (US5 to US9) are responsible for measuring this

distance for forward movement of mobile robot, while left side in back group sensors (US9 to US13) are responsible when for backward moving.

- **Right_Angle (θ_{or}):** Angle between heading angle of the robot and right obstacle, measured in degree.
- **Stop Simulation, when robot is hitting the obstacle**, the output becomes logic one if collision occurs, or logic zero if no collision for with obstacle occurs.
- **Sense**: When mobile robot senses any obstacle, this output is logic one, in order to select which controller must be active. When the mobile robot not detects any obstacle, this output is logic zero.
- **Angle of Goal**: When the user wants to move the mobile robot to target point (X_g, Y_g), straight line is created between the current position of mobile robot (X_c, Y_c) and target point, the angle between this straight line and horizontal axis is the angle of goal, in degree.
- **Stop Simulation:** When robot is arriving the goal, this output is logic one. If the center of mobile robot arrives near the point of goal (X_g, Y_g), it stops if the distance between them is less than 0.03[m] (this accuracy can be changed), otherwise it is logic zero.

III. Searching for Goal Point by Mobile Robot with Collision-Free Motion in Unknown Environment

In this section, fuzzy control is applied to navigate the mobile robot in unknown environment with obstacles to reach the goal point. When the mobile robot is moving and sensors detect an obstacle, an avoiding strategy is necessary. With obstacles present in the unknown environment, the mobile robot reacts based on the sensed information .The mobile robot changes its orientation and velocity when detect the obstacle. When the obstacle in unknown environment is very close, the mobile robot slows down and rapidly changes its orientation. The intelligent mobile robot reactive behavior is formulated in the fuzzy control scheme. The sensed information is processed to generate the values of the distance of the mobile robot from the obstacle and the relative angle between the heading angle of mobile robot and the obstacle.

A. Description of the Fuzzy Controller for Obstacle Avoidance (FCOA)

Fuzzy logic based control is applied to realize a mobile robot motion in unknown environment with obstacles. Inputs to the controller (FCOA) are:

1) **Left Distance**: The shortest distance between mobile robot and left obstacle.
2) **Left Angle**: The Angle between heading angle of the robot ($\theta_{o\ell}$) and left obstacle.
3) **Right Distance**: The shortest distance between mobile robot and right obstacle.
4) **Right_Angle**: Angle between heading angle of the robot (θ_{or}) and right obstacle.

Outputs of the FCOA controller are left and right wheels angular speed. The following membership functions are:

1) **Distance:** The distance membership functions are a three triangular membership function representing robot conditions that are good, near and far from the obstacle as shown in Fig.4.

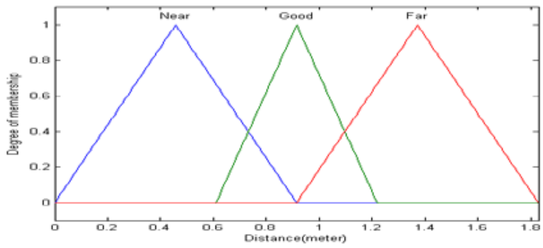

Fig.4: Distance membership functions

2) **Angle:** The angle membership functions are a five triangular membership function, representing angle conditions that are very negative, negative, zero, positive and very positive as shown in Fig.5.

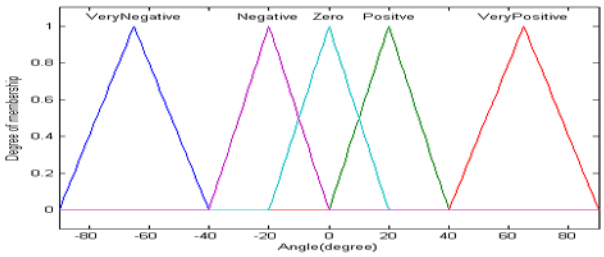

Fig.5: Angle membership functions

3) membership functions are a five triangular membership function, representing motor speed conditions that are backward, slow backward, stop, slow forward and forward as shown in Fig.6.

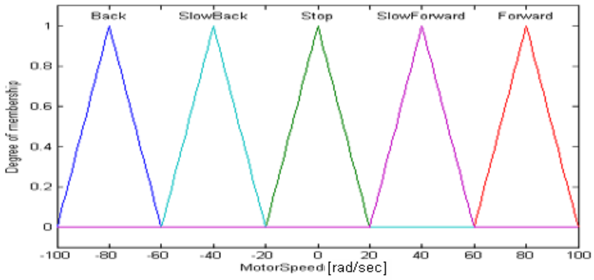

Fig.6: Motor speed membership functions

The next step illustrates the appropriate fuzzy logic rules. There are two schemes of rules as follows:

- Right Side Control (RSC) Rules.
- Left Side Control (LSC) Rules.

Table 2 illustrates matrix of rules for RSC, while LSC rules illustrated in the Table 3. The distance input is located on the vertical column, and the angle input is located on the horizontal row. Based on the fuzzified values of the inputs, the controller selects the appropriate motor conditions listed in the matrix (RSF=right motor slow forward, LSB=left motor slow backward, RF=right motor forward, LSF=left motor slow forward, LF=left motor forward, RSB=right motor slow back, RS=right motor slow)

Table 2: Right side controller (RSC) rules matrix

	Angle [Degree]				
Distance [m]	Very NE	NE	ZE	PO	Very PO
Near	RF LSF	RF LSF	RSF LF	RSB LSF	RSB LSF
Good	RSF LSB	RSF LSB	RF LF	RSB LSF	RSB LSF
Far	RSF LSB	RSF LSB	RF LSF	RSF LF	RSF LF

Table 3: Left side controller (LSC) rules matrix

	Angle [Degree]				
Distance [m]	Very NE	NE	ZE	PO	Very PO
Near	RF LSF	RF LSF	RSF LF	RSB LSF	RSB LSF
Good	RSF LSB	RSF LSB	RF LF	RSB LSF	RSB LSF
Far	RSF LSB	RSF LSB	RF LSF	RSF LF	RSF LF

B. The Fuzzy Controller for Reaching the Goal (FCRG)

In this section a fuzzy controller is used to move the mobile robot to a given goal point. The method is called Fuzzy Controller for Goal Reaching (FCRG). The Goal Reaching task is expected to align the robot's heading with the direction of the goal; θ_{diff} is the input of the fuzzy controller, which is angle difference between azimuth (the robot heading angle (θ)) (can be determined by encoder or by electronic compass) and the goal angle (θ_{goal}), as shown in Fig.7. Outputs of the fuzzy controller are left and right wheels angular speed (W_ℓ and W_r). The input variable "θ_{diff}" has 9 fuzzy sets: (back, left back, left, front left, front, front right, right, right back and also back). The output variable " W_ℓ" has 6 fuzzy sets: (quick turn right (QTR), go back (GOB), go more on right (GOMR), quick turn left (QTL), go ahead (GOA) and go fast ahead (GOFA)) and the other output variable "W_r" has 6 fuzzy sets: (quick turn left (QTL), go back (GOB), go more on left (GOML), quick turn right (QTR), go ahead (GOA) and go fast ahead (GOFA)).

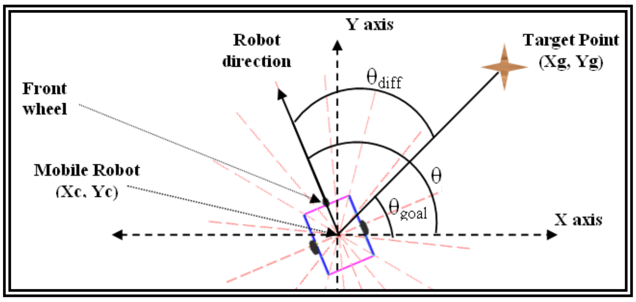

Fig.7: Goal orientation resolving

The following Figures (8, 9 and 10) for one input and two outputs membership functions respectively:

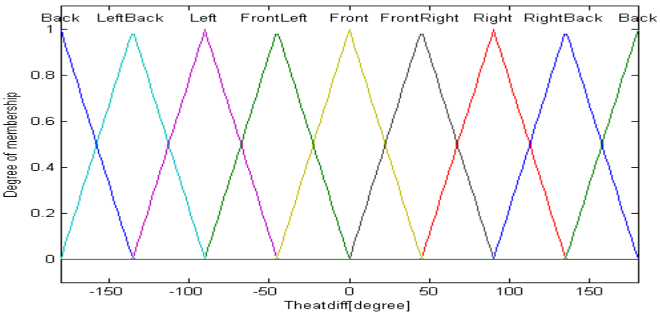

Fig.8: θ_{diff} membership functions (input)

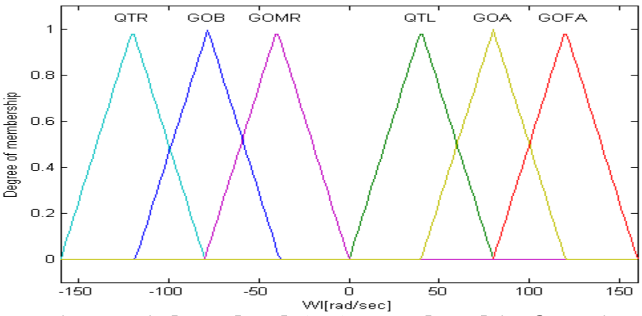

Fig.9: Light Wheel (W_l) membership functions (output1)

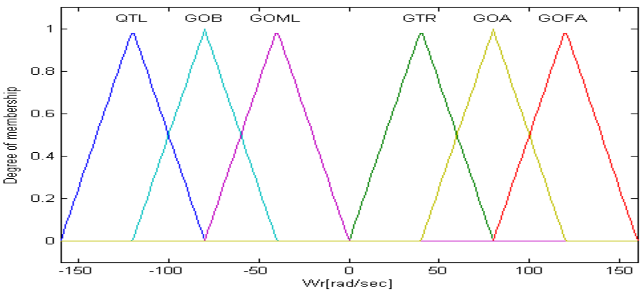

Fig.10: Right Wheel (W_r) membership functions (output2)

The inference system defined by the following rules:

1) If θ_{diff} **is** *front* **then** *go ahead.*
2) If θ_{diff} **is** *left* **then** *go more on left.*
3) If θ_{diff} **is** *right then go more on right.*
4) If θ_{diff} **is** *back then go back.*
5) If θ_{diff} **is** *front left* **then** *quick turn left.*
***6)* If** θ_{diff} **is** *left back* **then** *go back* **and** *go more on left.*
7) If $\theta diff$ **is** *right back* **then** *go back* **and** *Go more on right.*
8) If θ_{diff} **is** *front right* **then** *quick turns right.*

Fig.11 illustrates the θ_{diff} values and the fuzzy linguistic sets for the input variable. The mobile robot is stopping when arrive to the goal point with 0.03[m] accuracy of arrival (the distance between the center of mobile robot (X_c, Y_c) and target point (X_g, Y_g)). The mobile robot has two controllers, one controller for reaching the goal task and

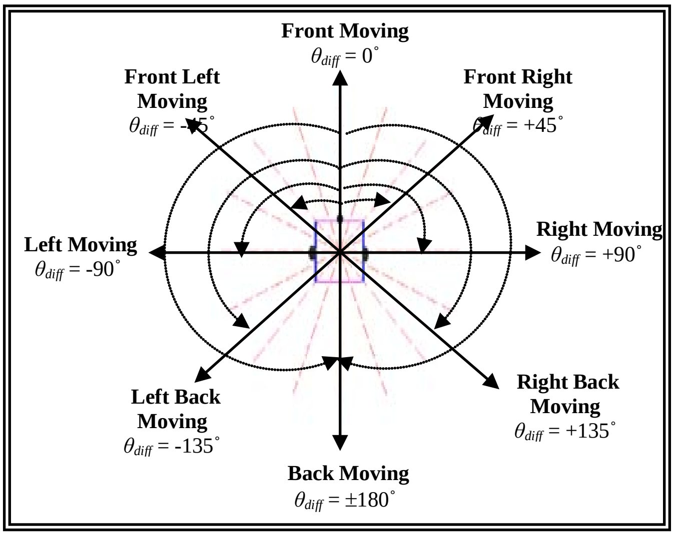

Fig.11: Movement of mobile robot according to the value of θ_{diff} to reach to goal point

The second controller for avoiding obstacles task. Mobile robot is moving toward the goal point in unknown environment by using fuzzy controller (FCRG), during movement of the robot toward goal; encounter obstacle, at the moment, the robot is sensing this obstacle by it sensors, autonomously, the mobile robot is transferring the control from (FCRG) to the second fuzzy controller (FCOA) to avoid collision with this obstacle, when the robot is not sensing any obstacle, the fuzzy controller (FCRG) perform the task of moving to goal point. This proposed approach is illustrated in Fig.12. The implementation of this system by using Matlab blocks is illustrated in Fig.13.

VI.Simulation Results

The mobile robot is generating a navigational plan in a given environment between predefined starting and goal points. The block diagram that is used in this simulation is shown in Fig.13. In the first case the mobile robot is assumed to move in static environment to reach the goal point (16, 16) as shown in Fig.14. As it can be seen, the navigation of mobile robot with velocity 8[m/sec], it takes a time of 3.58[sec] to reach the goal point without collision; also it illustrates the velocity, azimuth and errors for X and Y . In the second case a dynamic environment is considered. In the second case, during navigation, the robot faces a difficult situation when navigates in a complex structure of environment as shows in Fig.15. The obstacles are organized as shape (U). Robot succeeds to avoid being trapped in such obstacle, it takes a time of 4.49[sec] to reach the goal point (18, 8) at velocity 8[m/sec], it is also illustrates the velocity, azimuth and errors for X and Y.

V. CONCLUSION

This work deals with the modeling and control strategies of the motion of wheeled mobile robots. A new control scheme is developed for the mobile robot motion in an unknown environment with obstacles . The developed control scheme consists of two controller. The first one is for Obstacle Avoidance (FCOA) and the second for Reaching the Goal (FCRG).

This work revealed the success of the proposed fuzzy logic based control scheme for robot navigation in unknown environment with obstacles as shown in the simulation results in Figures[14-15]. Also results show that using such control scheme , the mobile robot can effectively achieve many tasks such as:

- The mobile robot can generate a navigational plan to avoid collisions with obstacles on its way in unknown environment.
- The mobile robot can generate a navigational plan to arrive into goal point.
- The mobile robot can generate a navigational plan between predefined starting and goal points and avoiding collisions with obstacles on its way in unknown environment.
- The mobile robot can navigate in a complex environment that includes obstacles with (U) shaped trap.

References

[1] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright and H. M. Tai. "Autonomous Local Path Planning for a Mobile Robot Using a Genetic Algorithm", Proceedings of the IEEE Conference on Robotics and Automation, Sacramento, California, 7-12, pp. 1398-1404, April, 2003.

[2] K. C. Ng, M. M. Trivedi, "A Neuro – Fuzzy Controller for Mobile Robot Navigation and Multirobot Convoying", IEEE Trans. on Systems, Man and Cybernetics-Part B: Cybernetics, Vol. 28, No. 6, pp. 829-840, ISSN 1083-4419 December 1998.

[3] M. A. Batalin, G. S. Sukhatme and M. Hattig, " Mobile Robot Navigation using a Sensor Network", In IEEE International Conference on Robotics and Automation pp. 636-642, New Orleans, LA, April 26 - May 1, 2004.

[4] D. Janglova, " Neural Networks in Mobile Robot Motion", International Journal of Advanced Robotic Systems, Volume 1 Number 1, pp. 15-22, ISSN 1729-8806, 2006.

[5] A. Albagul1 and Wahyudi, "Dynamic Modelling and Adaptive Traction Control for Mobile Robots", International Journal of Advanced Robotic Systems, Volume 1, Number 3 , ISSN 1729-8806, pp. 149 - 154,. Albagul, 2004,

[6] X. Jiang and Y. M. Xingquan, " Predictive Fuzzy Logic Controller for Trajectory Tracking of a Mobile Robot" , SMCia/05 2005 IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications Helsinki University of Technology, Espoo, Finland, June 28-30, 2005.

[7] S. E. Mahmoudi , A. A. Bitaghsir , B. Forouzandeh and A. R. Marandi, "A New

Genetic Method for Mobile Robot Navigation", Electrical and Computer Engineering Dep. University of Tehran, P.O Box : 14395/515, IRAN 2004.

[8] V. Mohan Peri and D. Simon" Fuzzy Logic Control for an Autonomous Robot" MSc Thesis, Cleveland State University Cleveland ,Ohio ,USA, 2002.

[9] J. Peng, Y. Wang and W. Sun," Trajectory-Tracking Control for Mobile Robot Using Recurrent Fuzzy Cerebellar Model Articulation Controller", Neural Information Processing – Letters and Reviews, Vol. 11, No. 1, January, 2007.

[10] S. J. Aldobuni. Fuzzy logic control scheme for motion tracking and navigation of autonomous mobile robot, University of Basrah, college of engineering, 2009.

[11] T. Petrinic, E. Ivanjko and I. Petrovic, "Amorsim − A Mobile Robot Simulator for Matlab", Department of Control and Computer Engineering, Faculty of Electrical Engineering and Computing, University of Zagreb Unska 3, HR-10000 Zagreb, Croatia, 2000.
http://www.uamt.feec.vutbr.cz/robotics/simulations/amrt/simrobot.zip

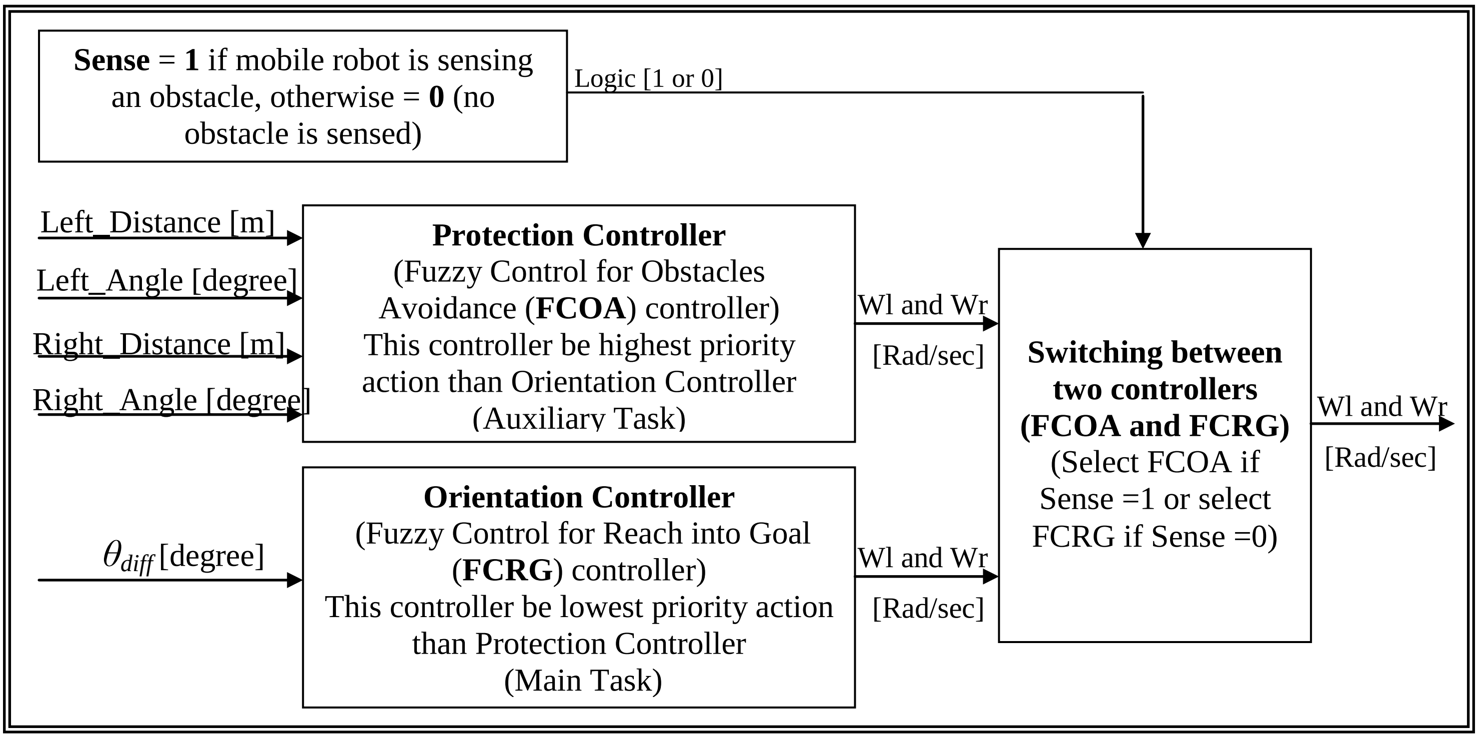

Fig.12: The proposed control structure to search for goal point with collision-free motion in an unknown environment with obstacles

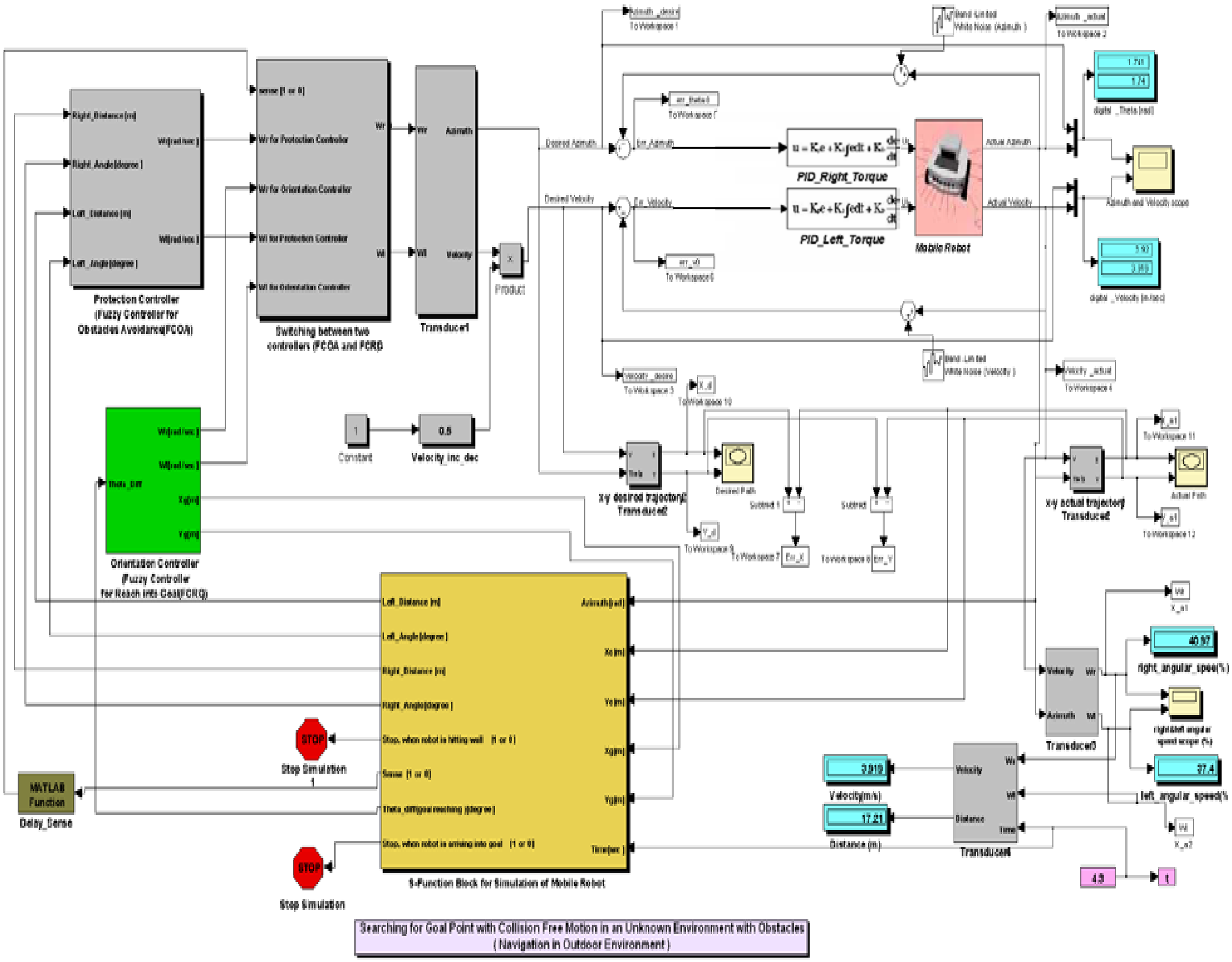

Fig.13: Block diagram for robot with (FCOA and FCRG) controllers

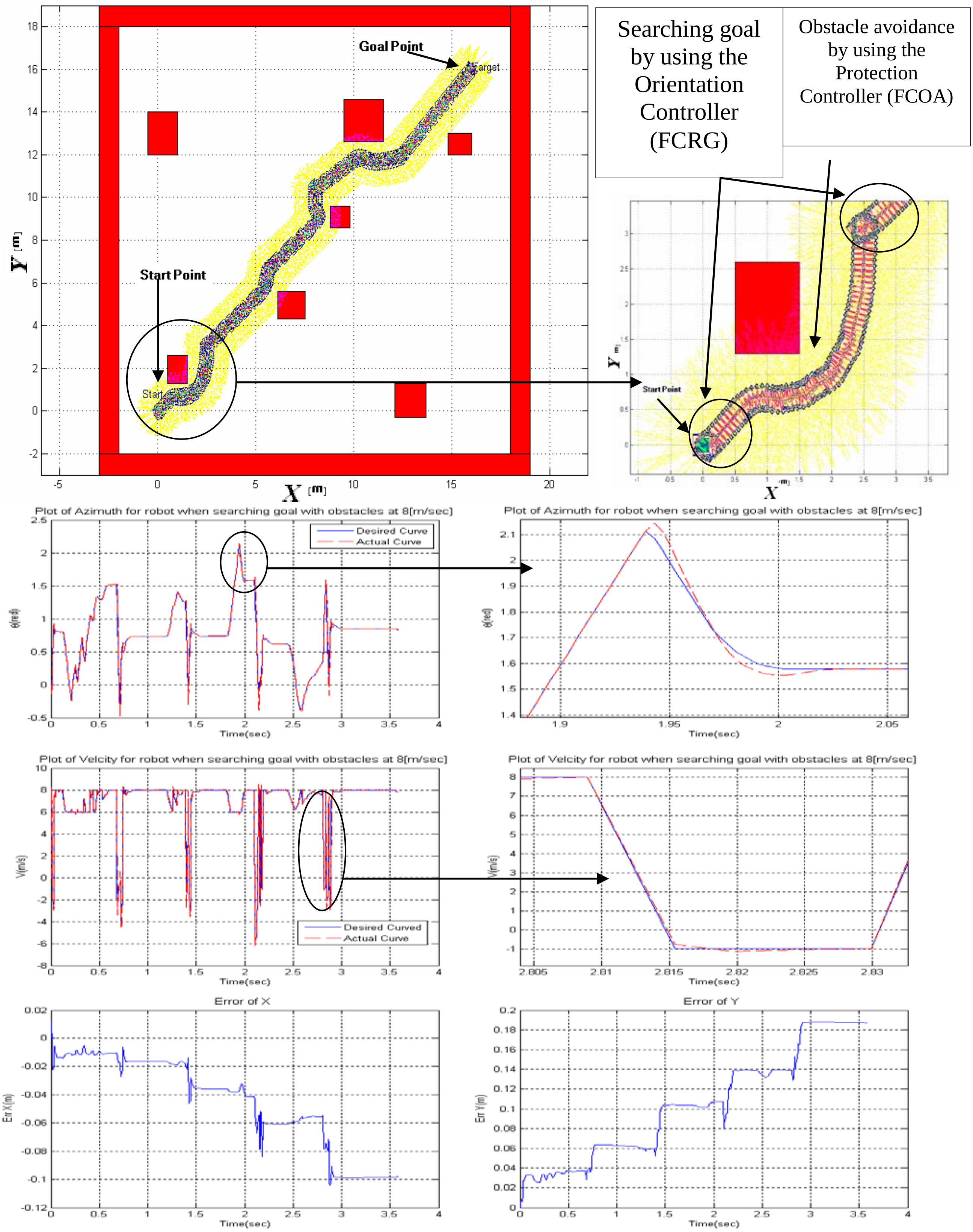

Fig.14: Navigation of mobile robot for searching proposed goal point (16, 16) and avoids collisions in static environment with obstacles in its way at max speed 8[m/sec] during 3.58 [sec] with curve of velocity, azimuth and errors for each X and Y

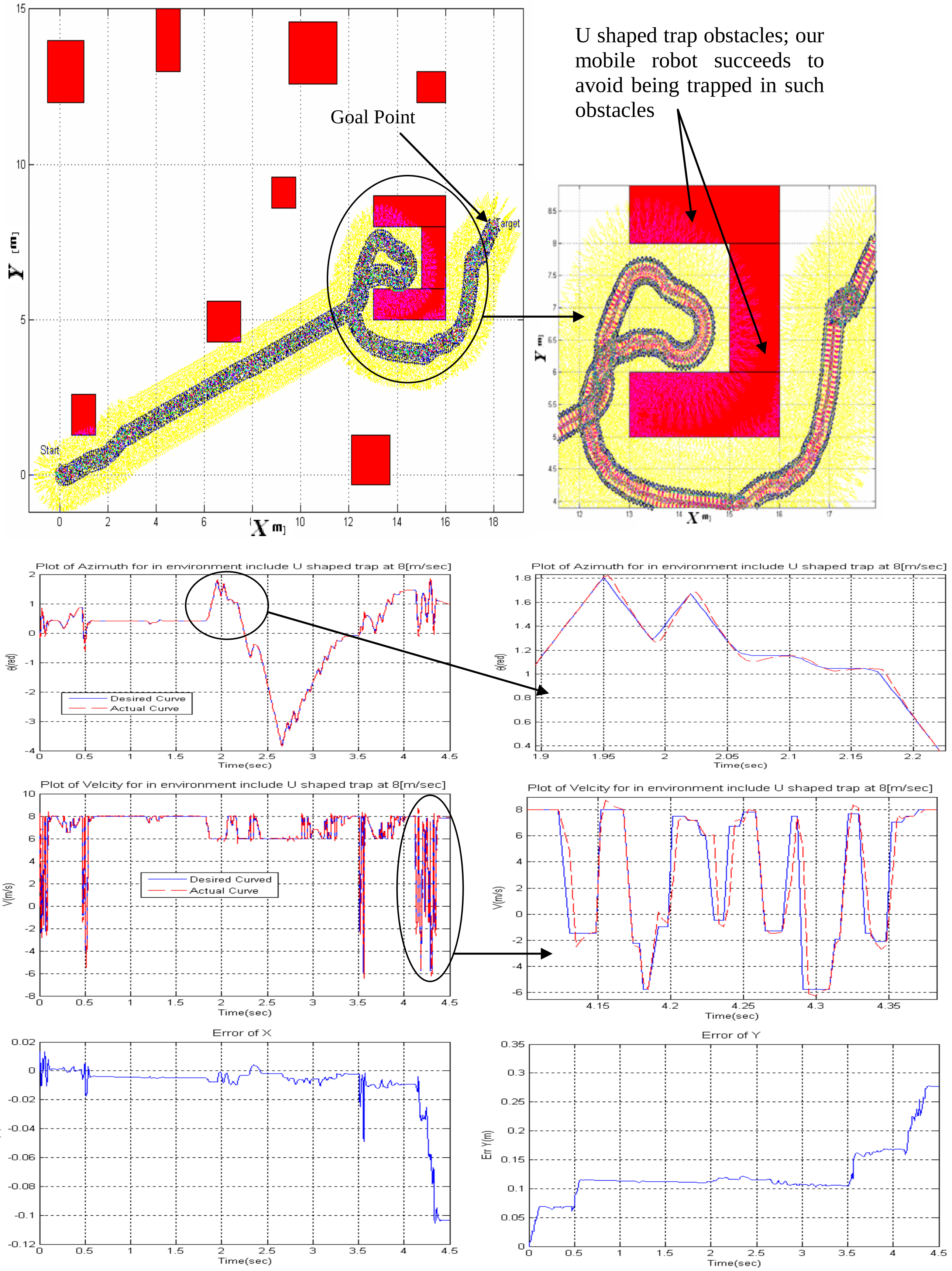

Fig.15: Navigation of mobile robot for searching proposed goal point (18, 8) and avoids collisions obstacles as shape (U) at max speed 8[m/sec] during 4.49 [sec] and curves of velocity, azimuth and errors for each X and Y trajectory tracking

البَحْث عن هدفٍ مِن قِبل روبوت نقَّالٍ بحركةٍ خالية مِنْ الاصطدام في بيئةٍ مجهولةٍ

أ.د.تركي يونس عبد الله

كلية الهندسة – قسم هندسة الحاسبات

جامعة البصرة – عراق

سُئار جواد الدبوني

كلية الهندسة – قسم هندسة الحاسبات

جامعة البصرة – عراق

الملخص

تجنّب العقبات إثناء حركةِ الروبوت النقال مِنْ المشاكلِ الأكثر أهميةً، خصوصاً في البيئاتِ غير المعروفة. في هذا البحث، تم اقتراح أسلوب من اجل الملاحة التي تَدْمجُ جزءان منفصلان هما تخطيط الطريقِ إلى الهدف وتجنّبِ عقبات إثناء الحركةِ وذلك من خلال تطوير جهاز سيطرةٍ ضبابيٍ. إنّ الروبوت النقَّالَ يحرك البيئة من خلال ستّة عشرَ متحسس من نوع سونارَ (موجات فوق الصّوتية)؛ هذه المتحسسات مثبتة حول الجسمِ. إنّ الإستراتيجياتَ التفاعليةَ المستعملة في هذه البحث أستندت على المعلوماتِ الحسّيةِ الفوق الصّوتيةِ والتفاعلاتِ الآنيةِ النسبيةِ بين الروبوت النقَّالِ والبيئةِ المجهولةِ . تم تصميم محاكي الروبوت النقَّالِ لإختِبار وتَطبيق أنظمةِ السيطرةَ بالإضافة إلى معرفة سلوكَهُ في البيئاتِ المختلفةِ. يحتوي نموذجَ الروبوت النقَّالِ على عجلتي قيادة حيث يتم السيطرة على العجلتين من خلال زاوية ميل الروبوت (Azimuth) و سرعته (Velocity) وذلك بواسطة جهازي سيطرة من نوع (PID). تم استعمال الخوارزمية الوراثية (Genetic Algorithm) من أجل حصول على أفضل تصميم لمسيطر(PID) لتتبع طريقِ الإنسان الآلي النقَّال.