Bio-inspired Transputer Based-Fuzzy Mobile Robot

Ebrahim Mattar[1], Khalid Al-Mutib[2], Hani Hamdan[3], and Mohamad Hamdan[4]

[1] University of Bahrain, P.O. Box 32038, Kingdom of Bahrain
[2] King Saud University, College of Computer and Information Sciences, Department of Software Engineering. P.O. Box 51178, Kingdom of Saudi Arabia
[3] SUPELEC, Department of Signal Processing and Electronic Systems, Rue Joliot-Curie, 91192 Gif-sur-Yvette cedex, France
[4] Université Libanaise, Faculté de Génie, Department of Electrical and Electronics, Engineering. Beirut, Lebanon

ebrgallaf@eng.uob.bh, mutib@ksu.edu.sa, Hani.Hamdan@supelec.fr, mhamdan@ul.edu.lb

Abstract. Mobile robots have been used widely in large number of intelligent based applications, where a random manoeuvring is an essential part of such systems, [1,2]. In this regards, experience has shown that mobile robots do require advanced parallel intelligent based control mechanisms. In this article, it is shown how a mobile robot can be controlled in movement via some rule based system. In this sense, the design, construction and testing of an autonomous mobile robot have already been defined in the literature. A Transputer Embedded Real-Time Controller has been widely used for industrial applications due to their advances. Here, a Fuzzy based Transputer System is used (on board the robot) to meet some intelligence requirements for the navigation and obstacle avoidance. A parallel fuzzy controller is implemented for a mobile robot guidance and obstacle avoidance. This is based on parallel processing of sonar data that do come from the mobile sensory system.

Keywords: Bio-Inspired Robots**,** Parallel Processing, Occam Processes, Mobile Robots Kinematics, Fuzzy Systems, Autonomous Robot.

1 Introduction

1.1 A Mobile Robotics System

The mobile robot indicated here in this article is known as the " Occam Mobile Robot". The robot has been modeled and controlled through parallel processing units. Occam parallel processes has been used to control the mobile robot is shown in Fig. (1). For the design of the robot, the following assumptions were made accordingly:

{a} The mobile robot must be capable to follow an (x,y) path in any direction (θ) over the plane.
{*b*} It must be robust in its mechanical design.

J. Suzuki and T. Nakano (Eds.): BIONETICS 2010, LNICST 87, pp. 607–620, 2012.

{c} It must include sufficient sensors and actuators.
{d} It has to be easily controlled and of ease to maintain.
{e} It should be intelligent to make navigation decisions on its own.
{f} It should have simple mechanical design.
{h} All steering axes are perpendicular to the surface.

In addition, the following operational assumptions were assumed:

{i} The mobile robot is to move on a planar surface.
{ii} The translation friction at the point of contact between a wheel and the surface is large enough so that no translational slip occur.
{iii} The rotational friction at the point of contact between a wheel and the surface is small enough so that rotational slip may occur.

Transputer Embedded Real-Time Controller has been widely used for industrial applications due to their advances. Fuzzy based Transputer system is used on board the robot to meet the intelligence requirement for the navigation and obstacle avoidance. Parallel fuzzy controller is implemented for the robot guidance and obstacle avoidance. This is based on processing of sonar data that do come from the mobile sensory system.

1.2 Proposed Control System

The system proposed here, is to allow a fuzzy controller to control the movement and the maneuver actions of a mobile robot. This is done via the use of a parallel processing of the sensory data. Sensory data are processed in parallel fashion, hence a fuzzy controller output is computed accordingly quickly. The proposed fuzzy controller is then used to control the movement of the mobile robot. The fuzzy controller has been trained and defined using the operators experience.

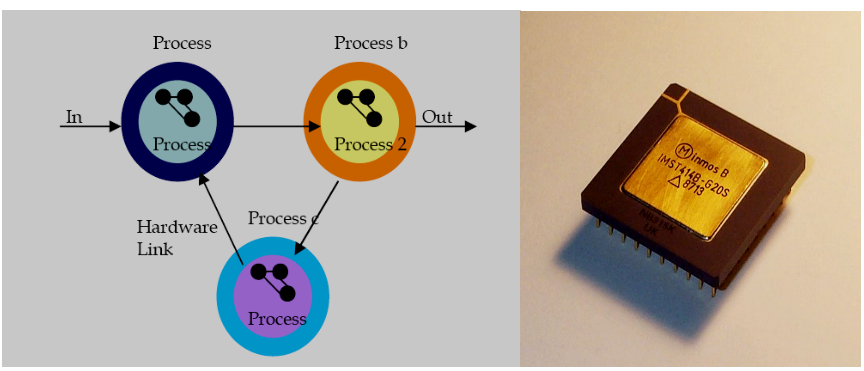

Fig. 1. The Occam process, and a T414 Transputer chip. The transputer architecture was designed as an Occam engine.

2 Mobile Robot Physical Construction

2.1 Robot Physical Body

We define the following: The Occam Mobile Robot can follow a path in any direction but first it must rotate around its axis to face that direction. It has two diametrically opposed drive wheels. Because of the simplicity of its mechanical design it has simple kinematics. The mobile robot weighs about two kilograms in a rectangular shape of width (30cm) and length of (40cm). It has two wheels of diameter (10 cm) located at the centre of the robot used for motion and steering. Each wheel actuates through a gear (10:1 ratio) by an independent DC motor. A shaft encoder is attached to each motor for wheel position sensing. Two free-wheeling castors are located one in the front and one at the back of the robot to balance the robot. The corners of the robot circumference are flat and with (45^o) angle from the adjacent flat wall. These flat corners are used to mount the corner ultrasonic sensors in order to enable the robot to see in that direction. The sensors are mounted around the circumference of the robot body, and they are at 30cm high from floor. The sensors are slightly tilted upward to prevent ultrasonic echo reflection from the ground floor. The mobile robot is powered by four rechargeable (12V batteries), 2×12Ah and 2×6Ah. These batteries are configured to supply (+12V, −12V, +24V and −24V). A voltage converter is also used to provide (+5V).

2.2 Robot Kinematics Development

To provide a framework within which to develop the robot kinematic models, Muir and Neuman, 1986 [3], defined a wheeled mobile robot as: " A robot capable of locomotion on a surface solely through the action of wheel assemblies mounted on the robot and in contact with the surface. A wheel assembly is a device which provides or allows relative motion between its mount and a surface on which it is intended to have a single point of rolling contact", [2]. Kinematics is the study of the geometry of motion. All kinematics are derived by straightforward application of the axioms and corollaries of the transformation algebra, [3]. In the context of the wheeled mobile robots, the motion of the robot can be determined from the geometry of the constraint imposed by the motion of the wheels.

3 The Mobile Robot Fuzzy Controller

3.1 The System Controller

In the mobile robot fuzzy controller, the eight ultrasonic sensor range measurement data are the input to the controller and the output is the left and the right wheel speeds as shown in Fig. 2.

This fuzzy controller is integrated with the sensor fusion and the low level PID controller to form the complete controller for the mobile robot as shown in Fig. (3).

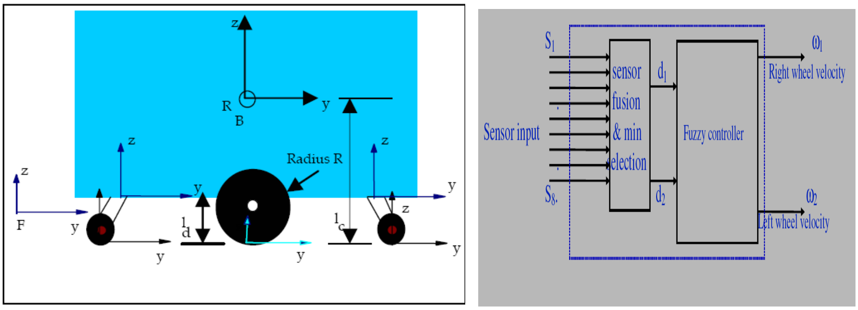

Fig. 2. Body coordinates assignments. Input-Output Mobil Robot Fuzzy Controller.

The overall controller starts with a user interface where the user can enter the desired control parameters and in this interface the statues of the mobile robot will be displayed before starting the controller. The robot checking routine will perform self testing on the various parts of the hardware of the robot and report any problem to the user before starting. The sensor fusion is the integration of all of the eight ultrasonic sensors to form the input to the fuzzy controller. The robot sensors are grouped in to four regions as shown in Fig. 4.

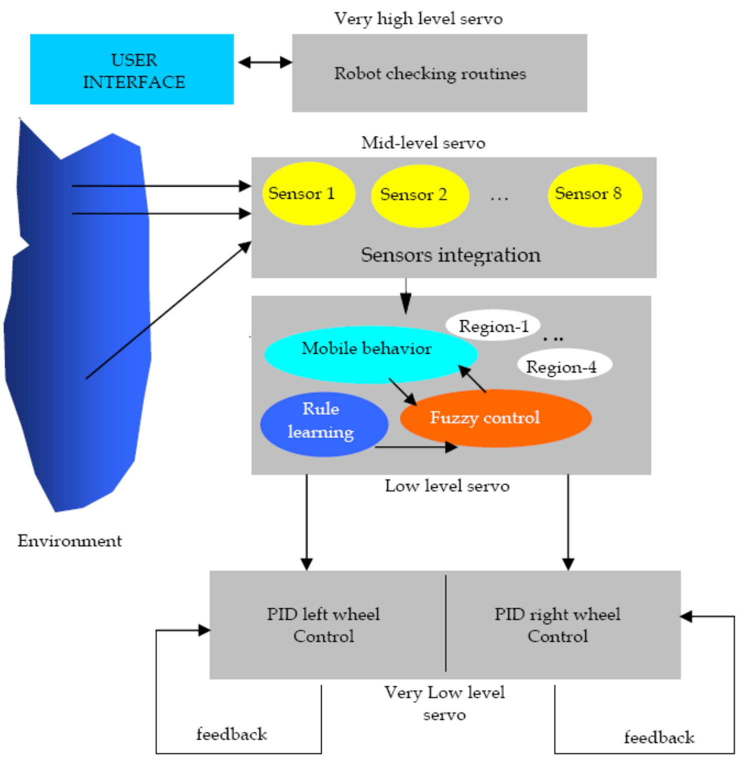

Fig. 3. Block diagram of the overall controller for the mobile robot

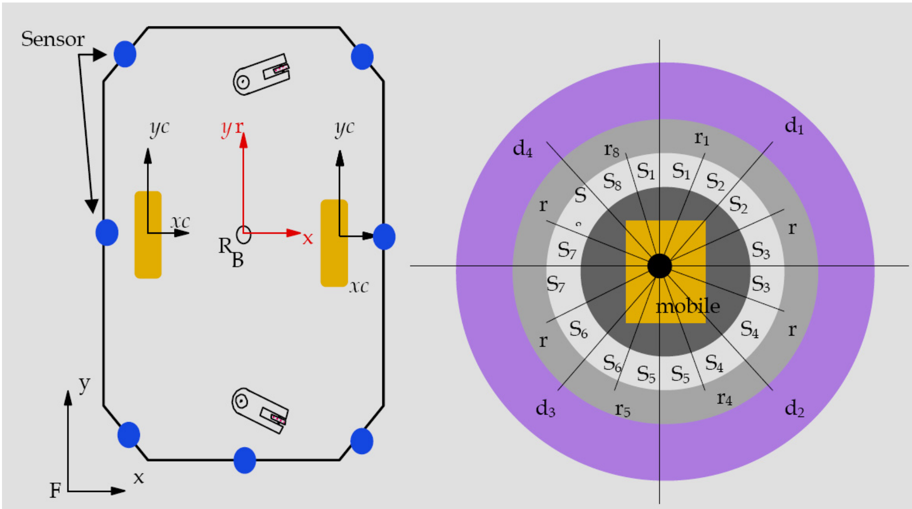

Fig. 4. Mobile sensing units: As grouped in four regions

Region #1 consist of sensor (1, 2 and 3). Region #2 consist of sensor 3, 4 and 5. Region #3 consist of sensor 5, 6 and 7. Region #4 consist of sensor #7, 8 and 1. In this arrangement some sensors are part of more than one region and this grouping is consistent with the fuzzy group classification where the borders between the sets (regions) are not crisp. In the process of selecting a region out of the four, the following steps are implemented:

1) Read all the sensors (S_1 to S_8), and pass to controller.
2) Arrange the sensor output value in ascending order.
3) Identifying the three sensors with the lowest value.
4) Check if two of the three are in one region.
5) If yes, that region will be selected.
6) If no, go to step one again.

The selection of the region will define the fuzzy rule that will be used in the fuzzy controller.

3.2 Fuzzy Control Information Processing

The processing of the information in the fuzzy process will first include the fuzzification of the input variables and then the defuzzification of the output variables as discussed in (4.4).

3.3 Fuzzification of Sensors Data (Variables d_{min1} and d_{min2})

Once the region is selected the information supplied by the sensor in that region will indicate the nearest obstacle to the robot and the next nearest obstacle (d_{min1} and d_{min2})

facing that region. The fuzzy control can be realized by the control of the robot wheels rotation to keep the robot within a pre-set distance from the obstacle facing that region of the robot. A fuzzy operator converts the crisp input data, say{d}, into the linguistic values $(\hat{d})$ determined as labels of fuzzy sets given by Equ (1):

$$\text{fuzzifier}(d_1, d_2) \Rightarrow \in (\hat{d_1}, \hat{d_2}) \tag{1}$$

where *fuzzifier* denotes a Fuzzification operation. From now on, the sign (^) representing the fuzzy set will be omitted for simplicity. The input linguistic variables d_1 and d_2 are expressed by linguistic terms as illustrated by the membership function in Equ. (2) and Equ. (3) and in Fig. 5. respectively. The meanings of the linguistic terms are given in Table (1).

$$\{es\ \ vvs\ \ vs\ \ s\ \ m\ \ b\ \ vb\ \ vvb\ \ eb\} \subseteq m(d_1) \tag{2}$$

$$\{es\ \ vvs\ \ vs\ \ s\ \ m\ \ b\ \ vb\ \ vvb\ \ eb\ \ eeb\} \subseteq m(d_2) \tag{3}$$

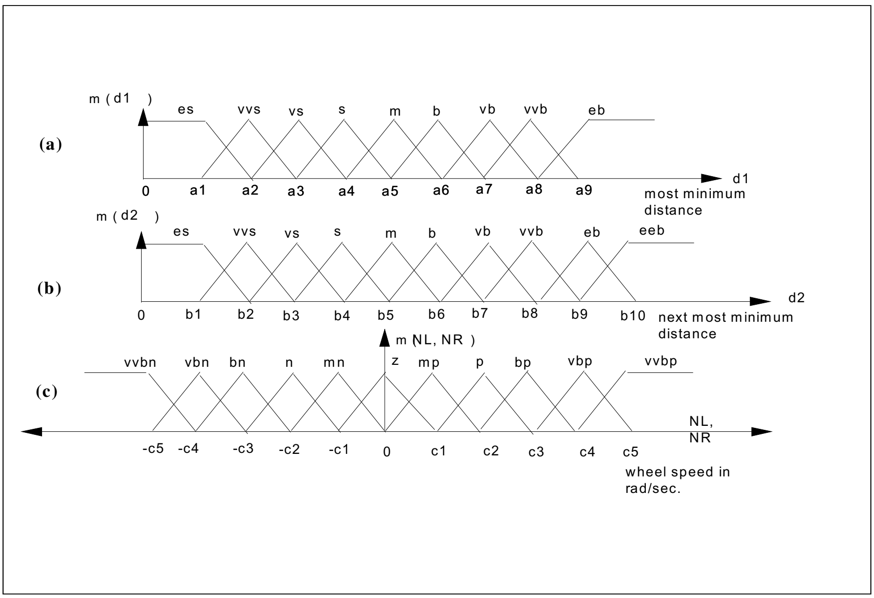

Fig. 5. Fuzzification of the input and output variables

The fuzzy output is the speed of the right and the left wheels in (rad/sec). They are expressed by the linguistic values with membership functions having a triangular shape functions as shown in Fig. 5. The halfway of this membership function is determined by an initial value of zero. The linguistic provisions for the output variables are given in Equ (4) and in Table (2) :

$$\{vvbn,\ vbn,\ bn,\ n,\ mn,\ z,\ mp,\ p,\ bp,\ vbp,\ vvbp\} \subseteq m(N_L) \tag{4}$$

Table 1. Membership for the input variables d_1 and d_2

Symbols	Linguistic variable
es	extremely small
vvs	very very small
vs	very small
s	small
m	medium
b	big
vb	very big
vvb	very very big
eeb	extremely big

Table 2. Membership for output variables (N_L and N_R)

Symbols	Linguistic variable
vvbn	very very big negative
vbn	very big negative
bn	big negative
n	negative
mn	medium negative
z	zero
mp	medium big
p	positive big
bp	big positive
vbp	very big positive
vvbp	very very big positive

In Equ. (4), the eleven states membership function is expressed in terms of degree of membership function (MF). Fuzzy subsets embody elements with degree of membership. On the other hand, fuzzy FM $\mu_z(\ )$ of the fuzzy set{ }, assigns a real number between (0 to 1) to every element in the universe of discourse. With reference to the membership functions of the fuzzy sets defined in Fig. 5., these membership values indicate the degree to which the element (a number) belongs to the fuzzy set. A fuzzy membership function can have various shapes depending on the designer's preference and the need for it in a particular fuzzy controller. The triangular shape is the most common shape of the membership function in the literature and the easiest to implement, [7]. After isolating the control features of the fuzzy controller and decomposing each variable into fuzzy sets, the conceptual model is completed. This done by writing the production rules describing the action taken on each combination of the control variables. The mobile fuzzy controller is a two input (d_1, d_2) and one output which is the desired speed of the motor (refer to Fig. 5.). It is convenient to consider such control as an individual matrix of actions in an ($d_1 \times d_2$) array.

The fuzzy states of one control variable (d_1) form the horizontal axis and the fuzzy states of the other control variable (d_2) form the vertical axis. At the intersection of a row and column is the fuzzy state of the solution variable. For this system the representation assumes the shape of ($d_1 \times d_2 \times N_R$) cube. Systems of higher order are most difficult to conceptualize but, in principle, the same kind of correspondence between control and solution variables exists. This form of representation, very common in the control engineering field, is called a Fuzzy Associated Memory or the (FAM). Table. 3. shows the FAM for the right wheel speed and in Table. 4. the FAM for the left wheel speed.

3.4 Construction of Rule Base Fuzzy System

The rule base for realizing each behaviour can be constructed based on the operator experience. For the fuzzy controller, the partial mapping is to be translated into linguistic rules. The fuzzy rule has an (IF-THEN) format as follows :

Table 3. The FAM for the right wheel speed (Region 1)

d_1 / d_2	es	vvs	vs	s	m	b	vb	vvb	eb
es	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vvs	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vs	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
s	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
m	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
b	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vvb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
eb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
eeb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp

The FAM for the right wheel speed (Region 1)

d_1 / d_2	es	vvs	vs	s	m	b	vb	vvb	eb
es	z	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp
vvs	vvbp	p	mp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vs	vvbp	bp	mp	bp	vbp	vvbp	vvbp	vvbp	vvbp
s	vvbp	vbp	bp	bp	p	vbp	vvbp	vvbp	vvbp
m	vvbp	vvbp	vbp	vvbp	vbp	bp	vbp	vvbp	vvbp
b	vvbp	vvbp	vvbp	vvbp	bp	vbp	vbp	vvbp	vvbp
vb	vvbp	vvbp	vvbp	vvbp	vbp	vbp	vbp	vbp	vvbp
vvb	vvbp	vvbp	vvbp	vvbp	vbp	vvbp	vbp	vvbp	vvbp
eb	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp

Table 4. The FAM for the left wheel speed. The linguistic rules of the left wheel speed in region two.

d_1 / d_2	es	vvs	vs	s	m	b	vb	vvb	eb
es	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vvs	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vs	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
s	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
m	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
b	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vvb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
eb	vbp	vbp	vbp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp

The linguistic rules of the left wheel speed The FAM for the right wheel speed (Region 1)

Table 4. *(continued)*

d_1 \ d_2	es	vvs	vs	s	m	b	vb	vvb	eb
es	z	bp	bp	bp	bp	bp	bp	bp	bp
vvs	bp	z	mp	vbp	vvbp	vvbp	vvbp	vvbp	vvbp
vs	bp	bp	mp	bp	vbp	vvbp	vvbp	vvbp	vvbp
s	bp	vbp	bp	mp	p	vbp	vvbp	vvbp	vvbp
m	bp	vvbp	vbp	vvbp	p	bp	vbp	vvbp	vvbp
b	bp	vvbp	vvbp	vvbp	bp	p	vbp	vvbp	vvbp
vb	bp	vvbp	vvbp	vvbp	vbp	vbp	bp	vbp	vvbp
vvb	bp	vvbp	vvbp	vvbp	vvbp	vvbp	vbp	vvbp	vvbp
eb	bp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp	vvbp

IF (d_1 is *mb* AND d_2 is *vb*) then (N_L is *vvb* AND N_R is *vb*)

where d_1, d_2 , N_L, and N_R are the fuzzy variables and mb, vb, vvb and vb are the fuzzy subsets in the universe of discourses (X, Y and Z). The fuzzy rule base consists of several rules of the form given above. The experience of the operator play a big role in the shape and form of the rules and in the number of the rules, in the rule base fuzzy controller. In this fuzzy controller, the membership functions (μ_1, μ_2, μ_3, μ_4) with association with the (d_1) and (d_2) are computed as follows, [7]:

$$x = Int\,((d_1+1)/10) \qquad y = Int\,((d_2+1)/10)$$

where (Int) is the integer part of the expression :

$$xx = (10x + 5 - d_1) \qquad yy = (10y + 5 - d_2)$$

and complement of these fuzzy sets is given by:

$$(C_{xx} = 1 - xx) \qquad (C_{yy} = 1 - yy)$$

$$\begin{aligned} \mu_I &= \mu_i(d_1) \oplus \mu_i(d_2) \\ \mu_1 &= min\,(xx , yy) \\ \mu_2 &= min\,(xx , C_{xx}) \\ \mu_3 &= min\,(C_{xx} , yy) \\ \mu_4 &= min\,(C_{xx} , C_{yy}) \end{aligned} \tag{5}$$

In Equ. (5) is the fuzzy OR operator for the fuzzy set which select the minimum of the two variables given. The membership function in Equ. (5) is used in the defuzzification process, [12].

3.5 Defuzzification of Output Variables

In order to determine the crisp output (N_L ,N_R) from the fuzzy control action, a defuzzification process is required , [8]. The centre of gravity defuzzification is a commonly used method utilized in the control engineering applications. In the centre of gravity method we define the value (α) located under the centre of gravity of the area that is given by the function $\mu_{i\,=1,2,3,4}$. The crisp output is founded by, [8]:

$$N_L = \frac{\alpha_1\mu_1 + \alpha_2\mu_2 + \alpha_3\mu_3 + \alpha_4\mu_4}{\mu_1 + \mu_2 + \mu_3 + \mu_4} \tag{6}$$

where (α_1 , α_2 , α_3 ,α_4) are the corresponding values for the fuzzy rule for the N_L output. And for the second crisp output N_R the same Equ. (5) will apply using the corresponding (α_1 , α_2 , α_3 ,α_4) values. Each one of the four regions of the sensor grouping of Fig. 4. has two base rules, ones rule for the left wheel and another rule for the right wheel. The mobile robot fuzzy controller described above can be illustrated further by Fig. 6. This figure shows the interaction of the various parts of the fuzzy controller, the robot environment and the robot dynamics. In addition to the four regions rules, there is a fifth region which is the negative of the rule for region one. This fifth region is applied whenever the robot gets trapped in a situation where it cannot go left or right. This region well be selected by the fuzzy controller to reverse the robot from the entrapment. After the robot is out of the entrapment, one of the other four regions rules will be selected as applicable. The above fuzzy controller has been divided in two parts: the obstacle avoidance routine and then a

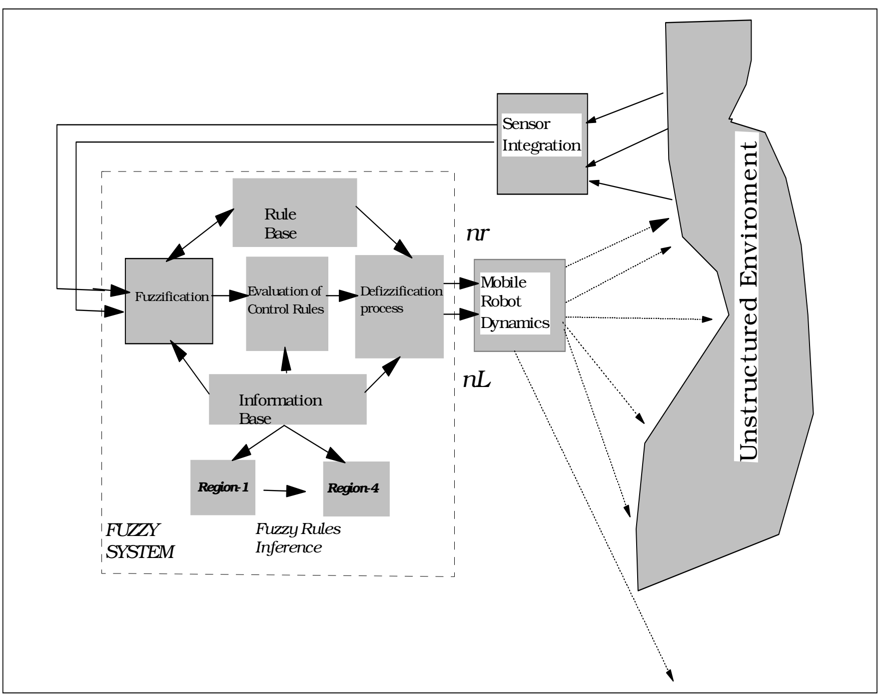

Fig. 6. Overall configuration of the fuzzy controller and its interaction with the robot dynamic and with the environments surrounding the robot speed in region one

goal seeking routine. The obstacle avoidance routine has been tested by simulation and by experiment which will be shown in the next section.

4 Simulation and Experimental Evaluation Fuzzy Controller

4.1 The Simulation Environment

The obstacle avoidance fuzzy controller has been simulated using the fuzzy rules and the kinematics of the mobile robot using MATLAB programming algorithms. The simulation x-y space has been made similar to the size of the operating area (6×6 meters) of the robot in the experimental setup. The simulation program starts by asking the initial (x,y) location in the (x,y) plane and the orientation angle with respect to the x-y world coordinate. The simulation shows that the robot moves around the plane geometry governed by the fuzzy-base rules without colliding with the surrounding boundaries. The simulation results are shown in Fig. 7. with the mobile robot as moving circles. The robot was tested, in real-time, using this fuzzy controller used in the simulation but was written in Occam language. Fig. 7. shows the experimental court (circles) with the erected walls. The robot in the experimental testing exhibited similar behavior to the one shown in this simulation. The experimental behavior of Occam Robot was recorded on a video tape.

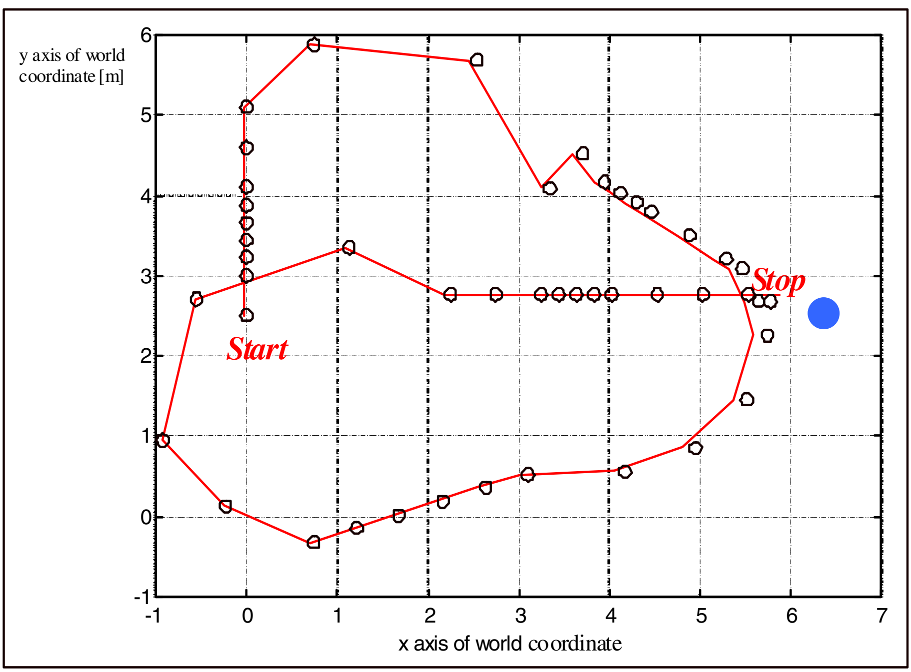

Fig. 7. The simulation of the robot movement using fuzzy controller with the robot moving in a circle. It started at 90° orientation at $x = 2.5$ and $y = 2.5$.

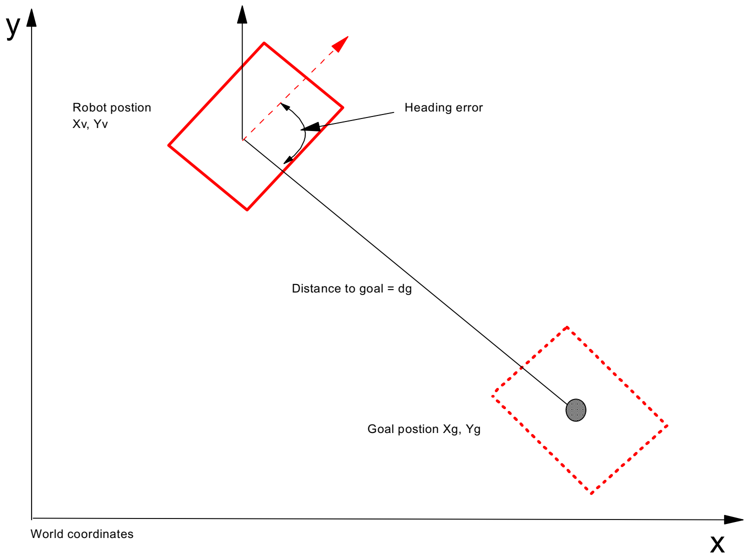

Fig. 8. The robot goal seeking and the global coordinate system

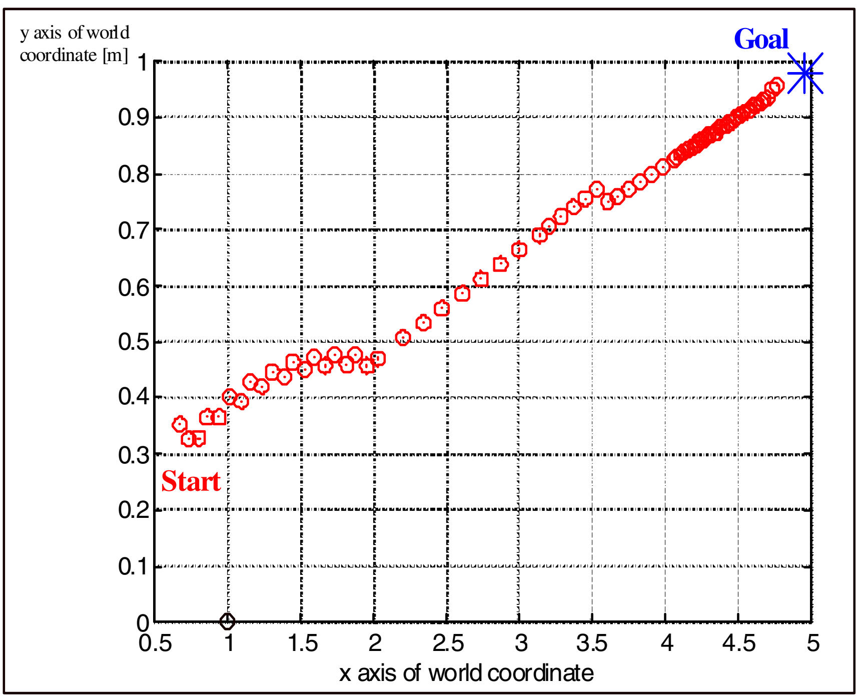

Fig. 9. The goal seeking simulation, the robot started at $x = 0.75$, $y = 0.35$ and 90^o orientation. First the robot rotated until it faced the goal and then proceeded to the goal.

4.2 Robot Goal Seeking

The goal seeking simulation is shown in Fig. 9. Here the robot starts from known position and orientation in the world coordinate and stopped at the location of the desired goal position. The control algorithm enables the robot to start from a known position and stop at the specified goal position. If the robot encounters any obstacle in its way, the obstacle avoidance controller will take control of the mobile robot until the path of the robot is clear from any obstacle within the seeing range of the robot. After the path is clear of any obstacle the goal seeking controller will be active again.

4.3 Robot Collision Avoidance

The goal seeking algorithm is based on first rotating the robot until it does face the goal if it is not already facing it, than the algorithm try to reduce the distance between the robot and the goal to zero as shown in Fig. 9., where the distance to goal d_g , [8]:

$$d_g = \sqrt{(x_g - x_v)^2 + (y_g - y_v)^2}, \quad \text{and heading error} \quad \theta_{HE} = (\theta_v + \theta_g)$$

5 Conclusions

In this paper a Transputer-based embedded controller has been successfully used to drive the mobile robot. Coordinate system frames are hence used for modeling purpose, as has been presented in details. The ability of the Transputer embedded controller (as a good example of parallel computation), to perform a demanding parallel fuzzy controller, has been demonstrated in a real-time experimental testing of the mobile robot in the obstacle avoidance behavior. Ultrasonic sensor fusion in the mobile robot navigation have been accomplished successfully. For such moving mobile robots, the parallel implementations of fuzzy controller and its advantages have been outlined. The obstacle avoidance fuzzy controller has been constructed and tested by simulation and experimentally. The goal seeking fuzzy controller has been also tested via some simulated robot movements and hence verified by some experimental tests.

References

1. Reister, D.B., Pin, F.G.: Time-Optimal Trajectories for Mobile Robots With Two Independently Driven Wheels. The International Journal of Robotics Research 13(1), 38–54 (1994)
2. Shgiley, J.E.: Kinematic Analysis of Mechanisms. McGraw-Hill, New York (1969)
3. Muir, P.F., Neuman, C.P.: Kinematics Modeling of Wheeled Mobile Robots. Journal of Robotics Systems 4(2) (1987)
4. Sarkar, N., Yun, X., Kumar, V.: Control of Mechanical Systems With Rolling Constraints: Application to Dynamic Control of Mobile Robots. The International Journal of Robotics Research 13(1), 55–69 (1994)

5. McKerrow, P.: Introduction to Robotics. Addison-Wesley Longman Publishing Co., Inc., Boston (1991) ISBN: 0201182408
6. Pedrycz, W.: Fuzzy Control and Fuzzy Systems, p. xiv + 225. Research Studies Press (1989) ISBN 0471 9231 17
7. Beom, H.R., Cho, H.S.: A Sensor-Based Navigation for a Mobile Robot Using Fuzzy Logic and Reinforcement Learning. IEEE Transaction on Systems, Man, and Cybernetics 25(3) (1995)
8. Baxter, J.W., Bumby, J.R.: Fuzzy control of a mobile robotic vehicle. In: Proceeding of Institute of Mechanical engineering, IMechE
9. Klir, G., Yuan, B.: Fuzzy Sets And Fuzzy Logic, Theory And Applications, 1st edn. Prentice Uranologist PTR (1995)
10. Meystel, A.: Autonomous Mobile Robots, Vehicles with Cognitive Control. World Scientific Publishing Co. Pte. Ltd. (1991)
11. Klafter, R.D., Chmielewski, T.A., Negin, M.: Robotic Engineering an Integrated Approach. Prentice-Hall (1989)
12. Mir, S.A., Zinger, D.S., Elbuluk, M.E.: Fuzzy Controller for Inverter Fed Induction Machines. IEEE Transaction on Industry Applications 30(1) (1994)
13. Benmounah, A.: Transputer Control of an AGV, Design, Construction, And Testing of a Mobile Platform. Ph.D. Thesis, University of Reading, UK (June 1991)
14. Maamri, M.: Control of Mobile Platform in a visually monitored environment. Ph.D. Thesis (May 1991)