A Fuzzy Error Correction Control System

Kim M. Moulton, Aurel Cornell, and Emil Petriu, *Fellow, IEEE*

***Abstract*—This paper describes a fuzzy error correction control system used to navigate a robot along an easily modifiable path in a well-structured environment. An array of Hall sensors mounted on the bottom of a robot gathers sensory information from a path of ferromagnetic disks placed on the ground. This sensory input is processed by an analog-to-digital converter and the output signals are then inputted into a fuzzy logic engine. The fuzzy engine outputs commands for the robot wheels. These commands determine the necessary angle of rotation to correct the direction of travel in order for the robot to remain on the path. The fuzzy logic controller stores prior disk information to predict a path trajectory when no path is detected. If the controller then senses a path, it anchors on it and starts following it.**

***Index Terms*—Degree of membership, error correction, ferromagnetic disks, fuzzy logic, Hall sensors, path prediction, robot, truth table.**

I. Introduction

BRAUNSTINGL [1] developed a wall-following robot that used a fuzzy logic controller and local navigation strategy to determine its movement. It obtained information by combining the readings of many ultrasonic sensors to obtain a general direction of travel. Three fuzzy input variables (perception angle, perception, and perception change) use this fuzzy input. The fuzzy logic controller uses the variables to control the firing of 33 rules. The fuzzy outputs (turning, steering, and acceleration) serve to control the robot's reaction to the environment. The main limitation of this implementation comes from the fact that the robot can only follow a wall and not a path (it is less flexible).

A fuzzy system developed by Surmann [2] controls the navigation of an autonomous mobile robot. The fuzzy navigator combines global strategies with local information it receives from sonar sensors, the global driving direction, and fuzzy state variables to determine what course of action to take. This system uses nine sonar sensors mounted on the sides of the robot to activate 27 different fuzzy rules. The entire system has about 180 fuzzy rules that associate 30 fuzzy inputs with 11 outputs. The fuzzy inputs consist of perceptions, sonar readings, and high-level commands. The fuzzy outputs consist of estimates of current state variables, steering angles, and driving speeds. The main limitation of this implementation is its 180 rules, which make it too complex.

A fuzzy system developed in Weingarten [3], Germany uses a set of 50 rules to control the docking of an automated guided vehicle (AGV). The fuzzy rule set is composed of rules that determine the vehicle's velocity and steering angle as a function of the position of the AGV. Infrared sensors on the AGV detect beacons placed on the docking target and are used to activate the appropriate fuzzy rules. This is a very limited application that only considers a docking station.

A helicopter fuzzy controller developed by Cavalcante in Florianopolis, Brazil [4], decomposes the movements of a helicopter into four separate blocks. These blocks define fuzzy inputs, outputs, sets, and rules that describe the fuzzy controller. The fuzzy inputs represent errors and error deviations. The fuzzy output represents corrections to the helicopter's movement to compensate for error deviations. The helicopter fuzzy controller relies on sensor accuracy and would not work well with fuzzy sensors; also, it is a very complex system. Other systems of interest are referenced in [5],and [6].

The contribution of the fuzzy logic controller presented in this paper is its ability to follow a discrete path. An additional benefit is that when the controller leaves the path, it can backup and predict where the path ought to be. If it finds the path, it can anchor onto it and continue following the path.

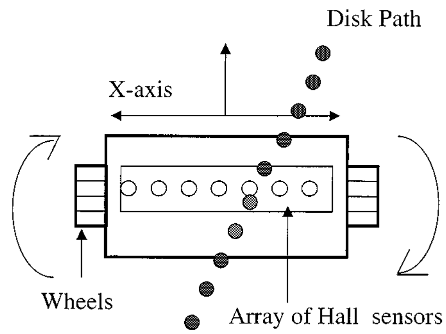

Fig. 1. Description of robot.

II. Description of the Robot

The task of the robot is to follow an imaginary path defined by a sequence of ferromagnetic disks placed on the floor. The robot for which the fuzzy control system has been designed has two driving wheels. At the bottom of the robot, positioned between the wheels, is an array of Hall sensors [7], [8]. The Hall sensors are biased with a magnet so that the sensors detect metal [5]. This array defines the X-axis of the robot. The direction the robot moves defines the Azimuth of the robot. See Fig. 1.

Each Hall sensor produces an electric signal as it passes over the top of the ferromagnetic disks (slide-by mode). This is represented by the triangles in Fig. 2. Depending on the position of the disk, one or two sensors are going to produce electric signals that define the position of the ferromagnetic disk on the X-axis

Manuscript received May 26, 1999; revised July 17, 2001.

K. M. Moulton is with PowerQuest Corporation, Orem, UT 84059 USA.

A. Cornell is with the Computer Science Department, Brigham Young University, Provo, UT 84602 USA.

E. Petriu is with the Electrical Engineering Department, University of Ottawa, Ottawa, ON, Canada.

Publisher Item Identifier S 0018-9456(01)09721-2.

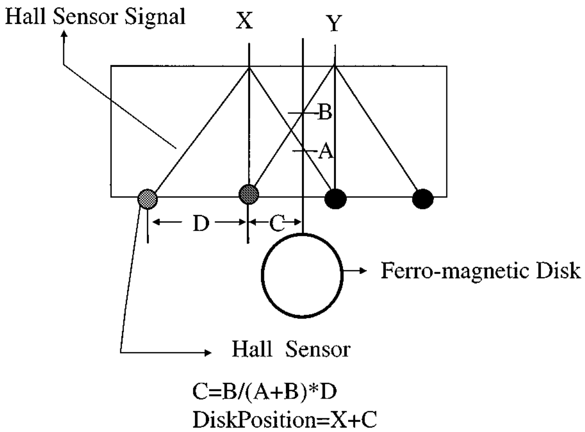

Fig. 2. Disk position calculation.

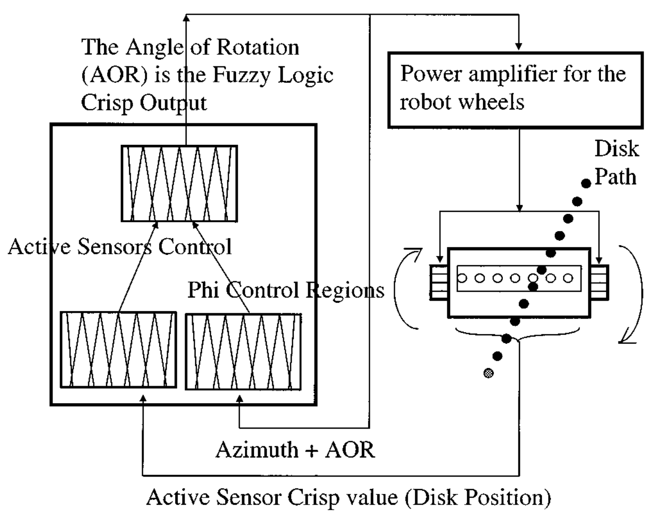

Fig. 3. Fuzzy logic controller.

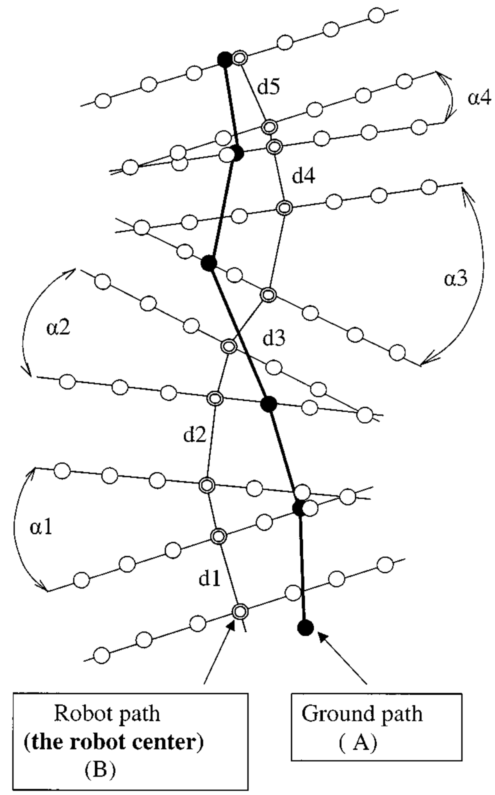

Fig. 4. Robot following path.

of the robot [9]. While the path to be followed can have any shape, the robot can follow smooth paths better.

III. Description of the Fuzzy Logic Controller

The fuzzy error correction control system uses two crisp inputs to produce an output, which is converted to electrical signals that control the two driving wheels.

The two input signals are as follows.

a) The disk position signal, produced by the array of Hall sensors. See Fig. 3.

b) The Phi input signal which represents the current direction (or Azimuth) of the robot. The Azimuth is calculated continuously as the robot moves.

The output of the fuzzy logic engine is the angle of rotation (AOR), which represents the error correction of the robot. The AOR is added to the old azimuth to obtain the new azimuth:

$$\text{Current Azimuth} = \text{Old Azimuth} + \text{AOR}.$$

The current azimuth then represents the new Phi and is input back into the fuzzy logic engine. Fig. 3 shows the structure of the fuzzy logic controller.

In this approach, the disks extend the sensing range of the Hall sensors, making the range continuous for the width of the sensor array. This provides the overlap necessary for smoothness and stability. The array of Hall sensors functions as a mobile X-axis, with the center sensor functioning as the origin. The goal of the fuzzy logic controller is to position this array so that the origin is centered over the path. This simple method allows a robot to follow an existing path, predict a path when none is present, and detect and follow a new path if one is encountered. One advantage of the method described in this paper is that there are no restrictions imposed on the path. It is also easy to modify the path—just reposition the ferromagnetic disks. The control system presented in this paper is simple, flexible, and low cost. Fig. 4 shows one ferromagnetic disk path, and a robot with a Hall sensor array following the path.

A. *Description of Fuzzy Control Regions*

The array of sensors is the basis for the first control region [10]–[12]. The active sensor control variable represents the current sensor information obtained through the Hall sensors. Each sensor represents a control region, which is a portion of the x-axis.

B. *Fuzzification of the Active Sensor Control Region*

There are a number of important guidelines to follow when creating fuzzy sets. It is usually advisable to divide the fuzzy variable into an odd number of fuzzy regions. This helps maintain a uniform distribution of values (positive and negative) on each side of the median [13]. It is also advisable to have between five and nine fuzzy regions for each fuzzy variable. Each fuzzy region should have a 10 to 50 % overlap with its nearest neighbors. The vertical sum of points that make up the overlap is constrained to be less than one. The reason the overlap is constrained to be less than one is because one equals complete membership. This overlap is important because it gives a smooth and stable surface to the fuzzy controller. Finally, the fuzzy sets should have their highest density around the optimal system control point and should spread out as they move away from that point [10].

In our experiment, seven control regions were selected. Each control region has a one-to-one correspondence with a sensor in the sensor array. The magnitude of sensor activation represents the degree of membership within the control region. We selected this number based on the recommendations above, the size of the robot, and the size of the ferromagnetic disks.

The membership function is selected based on the following consideration: when the Hall sensor crosses over the center of a disk, the electrical signal is maximized and has an activation of one in that particular control region. The further the Hall sensor is away from the center of the disk, the weaker the electrical signal gets, until at a certain point it decreases to zero. This represents no membership in the control region. Between these two extremes, we used linear interpolation to determine set membership. The width of the control regions is determined by the distance between the Hall sensors.

These seven control regions of the Hall sensor array are: *Lukewarm Left, Warm Left, Hot Left, Hot Center, Hot Right, Warm Right* and, *Lukewarm Right,* as shown in Fig. 5. The names of the control regions are intended to represent the idea that Hot Center is the desired position of the sensor array (e.g., the position to correct to). The fuzzy controller's objective is to position the sensor array so that Hot Center is always over the path being followed. For instance, if the crisp input value is 0.5, then two regions are considered in the process of calculating the output: Hot Center with a degree of membership 0.7 and Hot Right with a degree of membership of 0.2 (point A and point B in the graph of Fig. 5).

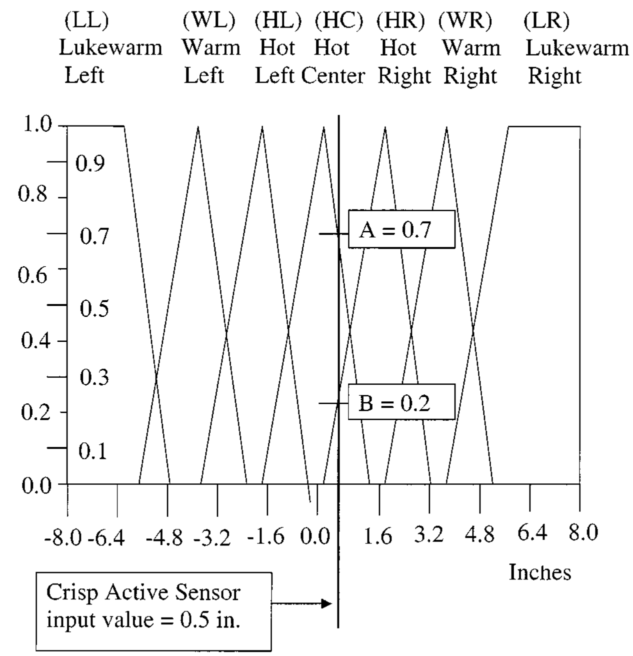

Fig. 5. Active sensor control regions.

C. Fuzzification of the Phi Control Region

The second input to the fuzzy logic system is the direction in which the robot is traveling, its azimuth or Phi. When the robot encounters a ferromagnetic disk, one or more of the Hall sensors will fire. When this occurs, the fuzzy logic controller will calculate an AOR correction based on the old azimuth value and active control variable (which was calculated based on which Hall sensors fired). This correction is then used to change the direction of travel of the robot. This is shown using the angles $\alpha 1$, $\alpha 2$, $\alpha 3$, and $\alpha 4$ in Fig. 4. After the robot changes direction, it obtains a new directional azimuth that is equal to the old azimuth plus the AOR. The robot travels linearly in the direction of the new directional azimuth until it encounters a new ferromagnetic disk, at which point the process is repeated.

Fig. 4 shows how the straight-line movement of the robot alters as the directional azimuth is updated. Line A represents the ground path to be followed and line B shows the movement of the center of the robot. The robot initially travels a distance d1 in a straight line until it encounters a ferromagnetic disk, then a correction $\alpha 1$ takes place. The robot then travels straight the distance d2, encounters another disk, and correction $\alpha 2$ takes place and so on. The AOR angle is calculated by the fuzzy engine and is used as an input to the Phi control regions as given in Fig. 3. We defined the angle of rotation to be between -180 to $+180$ degrees. If the angle exceeds these limits, it is corrected back within this 360-degree range. Based on previously considered recommendations, the following seven control regions were selected for the Phi variable: *Large Negative (LN), Medium Negative (MN), Small Negative (SN), Zero (ZE), Small Positive (SP), Medium Positive (MP), and Large Positive (LP).* These control regions (shown in Fig. 6) are used to fuzzify the crisp Phi input value.

For example, using a crisp input value of 8.0, two regions are considered in the process of calculating the output: Zero with a degree of membership of 0.8 and Small Positive with a degree of membership of 0.25 (see point C and D in the graph).

D. Angle of Rotation Fuzzy Regions

The output of this algorithm is the Angle of Rotation (AOR), which represents the actual path correction angle needed to remain on the path.

There are seven control regions in the AOR variable: *Negative Large, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium, and Positive Large.* They represent a range of -30 to $30°$. Fig. 7 shows how each control region covers a portion of that 60-degree range.

E. Description of the Fuzzy Rules

In this application, there are two fuzzy inputs and one fuzzy output. The two fuzzy inputs are the Active Sensor, AS, and the Azimuth, Phi. Each fuzzy input has seven control regions. The combination of these fuzzy input regions requires forty-nine fuzzy rules.

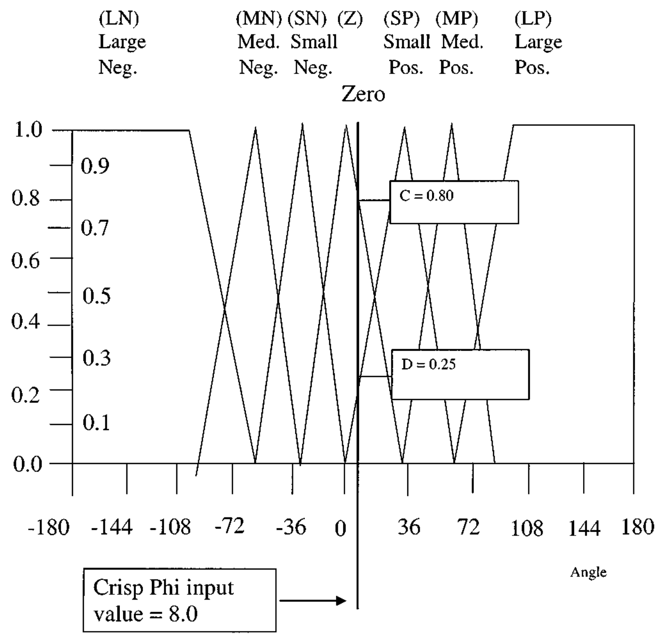

Fig. 6. Phi control regions.

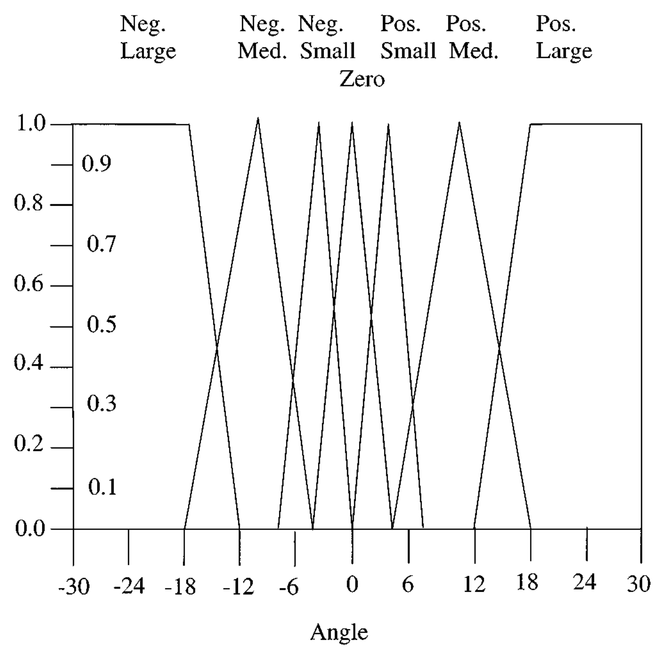

Fig. 7. Angle of rotation (AOR) regions.

The fuzzy output has seven control regions. For each combination of two inputs, there is an associated output region. Each region represents how much error has occurred. This association between the inputs, and output region is based on the amount of correction necessary to keep the robot on the path. There is a certain amount of symmetry in this correction. If the active sensor is zero and phi is zero, no correction is necessary. The farther the active sensor is from the center of the sensor array, then, depending on the value of phi, the larger the necessary correction will be. The fuzzy output is a consequence of the symmetry and the amount of correction necessary to return the robot to Hot Center. The truth table in Fig. 8 shows the relation between the

		ACTIVE SENSOR						
		LL	WL	HL	HC	HR	WR	LR
P	LN	PL	PM	PM	PS	ZE	ZE	NS
H	MN	PL	PM	PM	PS	ZE	NS	NS
I	SN	PM	PM	PS	ZE	NS	NS	NM
	ZE	PM	PS	PS	ZE	NS	NS	NM
	SP	PM	PS	PS	ZE	NS	NM	NM
	MP	PS	PS	ZE	NS	NM	NM	NL
	LP	PS	ZE	ZE	NS	NM	NM	NL

Fig. 8. Direction AOR truth table.

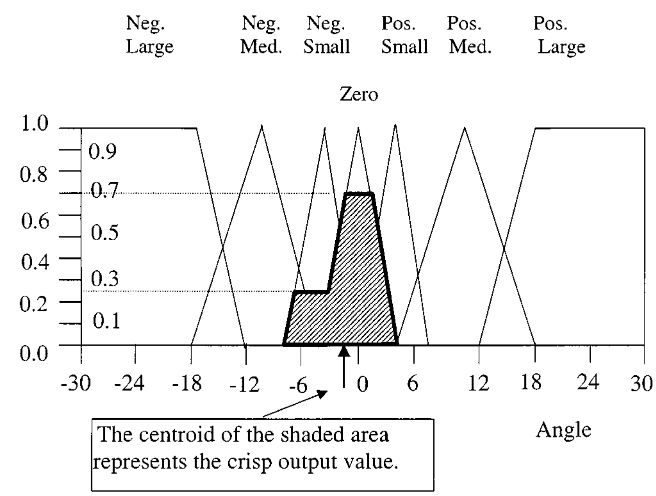

Fig. 9. Defuzzification of the output.

fuzzy inputs, the fuzzy output, and the symmetry of the correction.

The rules of the fuzzy system, given below, show how the two fuzzy inputs are related to the fuzzy output. The output is AOR (angle of rotation).

Lukewarm Left (LL)

IF AS is LL AND Phi is LN THEN AOR is PL;
IF AS is LL AND Phi is MN THEN AOR is PL;
IF AS is LL AND Phi is SN THEN AOR is PM;
IF AS is LL AND Phi is ZE THEN AOR is PM;
IF AS is LL AND Phi is SP THEN AOR is PM;
IF AS is LL AND Phi is MP THEN AOR is PS;
IF AS is LL AND Phi is LP THEN AOR is PS.

Lukewarm Right (LR)

IF AS is LR AND Phi is LN THEN AOR is NS;
IF AS is LR AND Phi is MN THEN AOR is NS;
IF AS is LR AND Phi is SN THEN AOR is NM;
IF AS is LR AND Phi is ZE THEN AOR is NM;
IF AS is LR AND Phi is SP THEN AOR is NM;
IF AS is LR AND Phi is MP THEN AOR is NL;
IF AS is LR AND Phi is LP THEN AOR is NL.

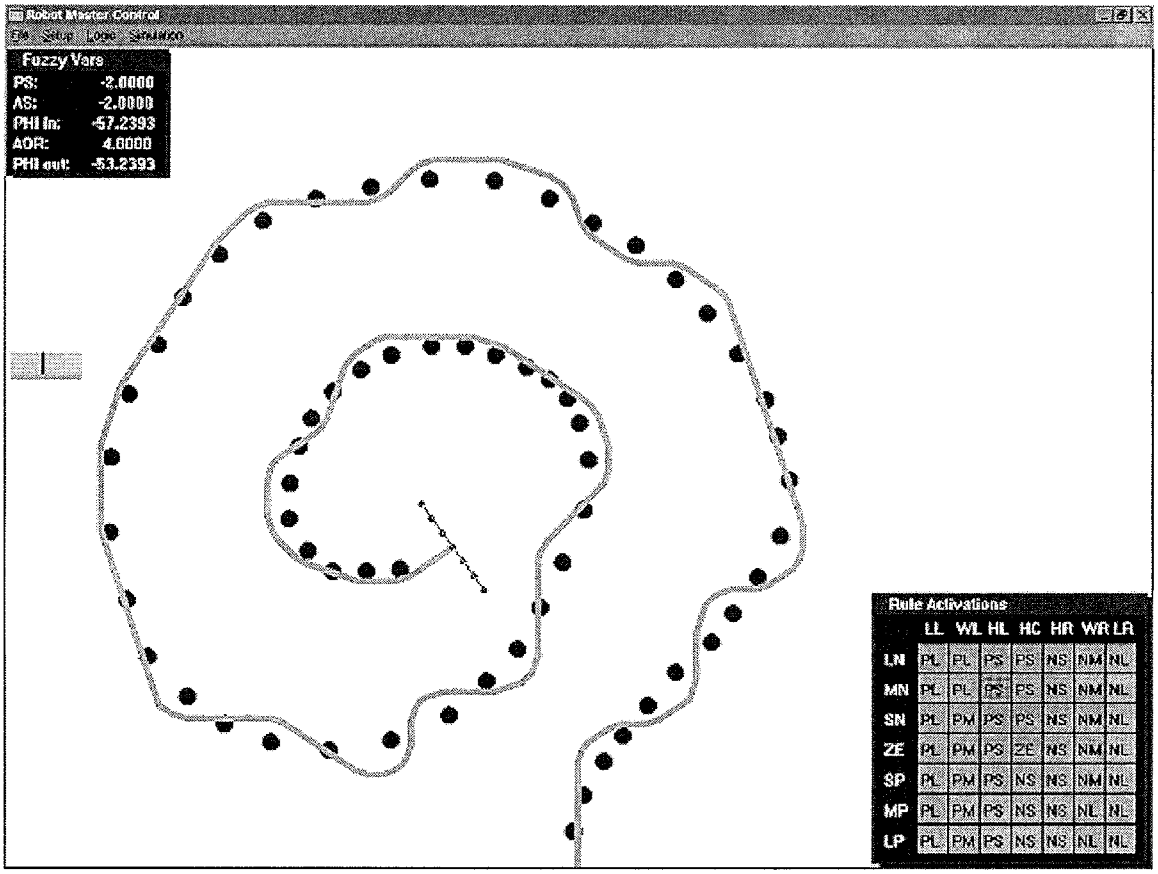

Fig. 10. Fuzzy logic controller simulation.

Warm Left (WL)

IF AS is WL AND Phi is LN THEN AOR is PM;
IF AS is WL AND Phi is MN THEN AOR is PM;
IF AS is WL AND Phi is SN THEN AOR is PM;
IF AS is WL AND Phi is ZE THEN AOR is PS;
IF AS is WL AND Phi is SP THEN AOR is PS;
IF AS is WL AND Phi is MP THEN AOR is PS;
IF AS is WL AND Phi is LP THEN AOR is ZE.

Warm Right (WR)

IF AS is WR AND Phi is LN THEN AOR is ZE;
IF AS is WR AND Phi is MN THEN AOR is NS;
IF AS is WR AND Phi is SN THEN AOR is NS;
IF AS is WR AND Phi is ZE THEN AOR is NS;
IF AS is WR AND Phi is SP THEN AOR is NM;
IF AS is WR AND Phi is MP THEN AOR is NM;
IF AS is WR AND Phi is LP THEN AOR is NM.

Hot Left (HL)

IF AS is HL AND Phi is LN THEN AOR is PM;
IF AS is HL AND Phi is MN THEN AOR is PM;
IF AS is HL AND Phi is SN THEN AOR is PS;
IF AS is HL AND Phi is ZE THEN AOR is PS;
IF AS is HL AND Phi is SP THEN AOR is PS;
IF AS is HL AND Phi is MP THEN AOR is ZE;
IF AS is HL AND Phi is LP THEN AOR is ZE.

Hot Right (HR)

IF AS is HR AND Phi is LN THEN AOR is ZE;
IF AS is HR AND Phi is MN THEN AOR is ZE;
IF AS is HR AND Phi is SN THEN AOR is NS;
IF AS is HR AND Phi is ZE THEN AOR is NS;
IF AS is HR AND Phi is SP THEN AOR is NS;
IF AS is HR AND Phi is MP THEN AOR is NM;
IF AS is HR AND Phi is LP THEN AOR is NM.

Hot Center (HC)

IF AS is HC AND Phi is LN THEN AOR is PS;
IF AS is HC AND Phi is MN THEN AOR is PS;
IF AS is HC AND Phi is SN THEN AOR is ZE;
IF AS is HC AND Phi is ZE THEN AOR is ZE;
IF AS is HC AND Phi is SP THEN AOR is ZE;
IF AS is HC AND Phi is MP THEN AOR is NS;
IF AS is HC AND Phi is LP THEN AOR is NS.

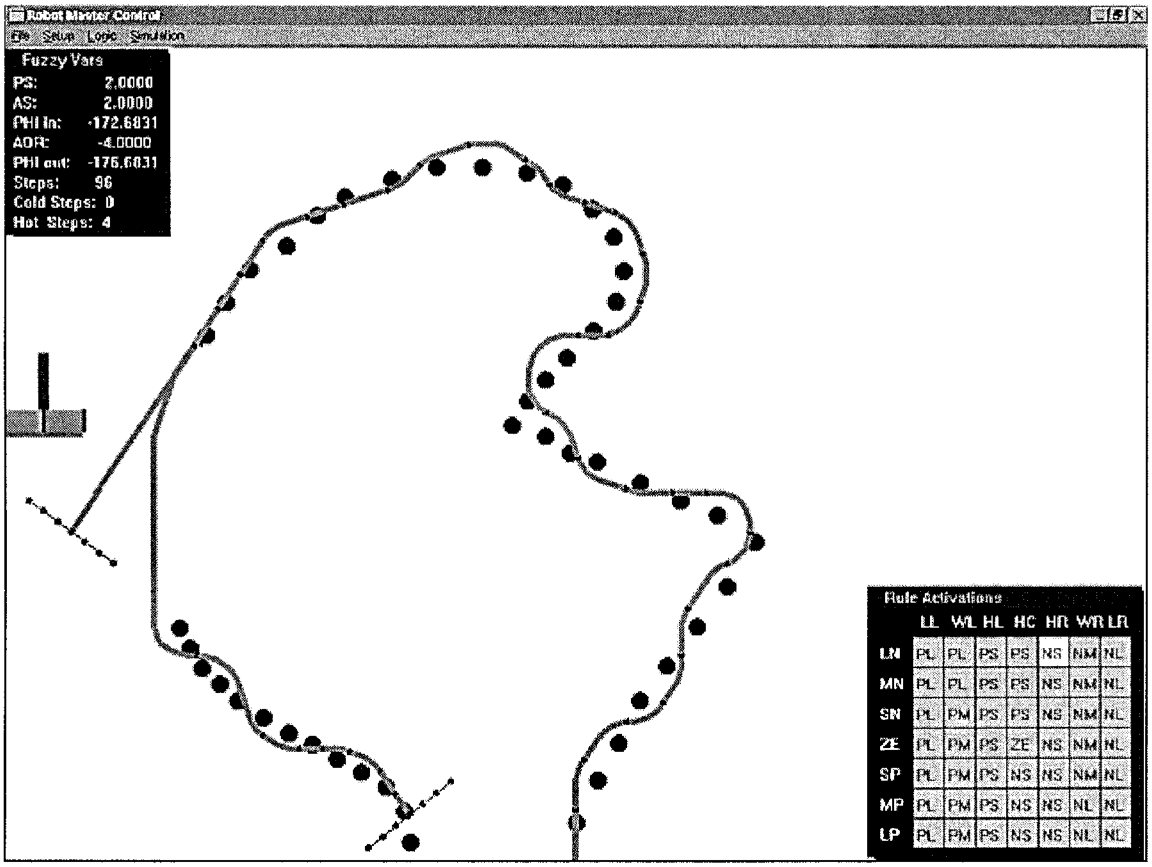

Fig. 11. Robot finding the path.

The same rules are represented below as a truth table with the linguistic variables grouped to show the symmetry of the rules. See Fig. 8.

F. Defuzzification of the Fuzzy Output

To show how the crisp output is produced, let us consider an example. For a crisp AS input of 0.5 the active sensor defines Hot Center with a degree of membership 0.7 and Hot Right with a degree of membership of 0.2. (See point A and point B in Fig. 5.)

For a crisp Phi input value of eight, the Phi control regions define Zero with a degree of membership of 0.8 and Small Positive with a degree of membership of 0.25 (Point C and D in Fig. 6).

Considering all the possible combinations of the input regions, the following four rules will create an output AOR of -2:

1. IF AS is HC AND Phi is ZE
THEN AOR is ZE; $\rightarrow$ output value of 0.7.
2. IF AS is HC AND Phi is SP
THEN AOR is ZE; $\rightarrow$ output value of 0.25.

The above rules (1 and 2) are combined by taking the larger value of each point on the horizontal axis. The output of the ZE region has a degree of membership of 0.7.

3. IF AS is HR AND Phi is ZE
THEN AOR is NS; $\rightarrow$ output value of 0.2.
4. IF AS is HR AND Phi is SP
THEN AOR is NS; $\rightarrow$ output value of 0.2

Using the same logic for two rules presented above (3 and 4), the output for the NS region has a degree of membership of 0.2.

The combination of these four rules produces the shaded area shown in Fig. 9. The crisp output, AOR, is calculated by taking the centroid of this area [14].

IV. The Fuzzy Controller Simulation

In order to test the design of the fuzzy logic controller, we implemented a simulation program with a graphical interface. The simulation demonstrates the dynamic behavior of the robot as it corrects its movements following a path. In the simulation, we used the Cubicalc [14] fuzzy logic engine.

An example of the graphical interface of the simulation is shown in Fig. 10. The large round dots represent ferrous-metal disk and the thick line represents the path the robot follows. The smaller round dots (at the end of the path) represent the array of sensors. The boxes are used to calculate and display the fuzzy variables and rule activations as the robot moves.

The simulation illustrates the ability of the robot to follow the path, even when the path is somewhat irregular.

V. Additional Features of the Controller

Two additional features were added to the design of the controller: first, the ability to predict the path based on the trajectory

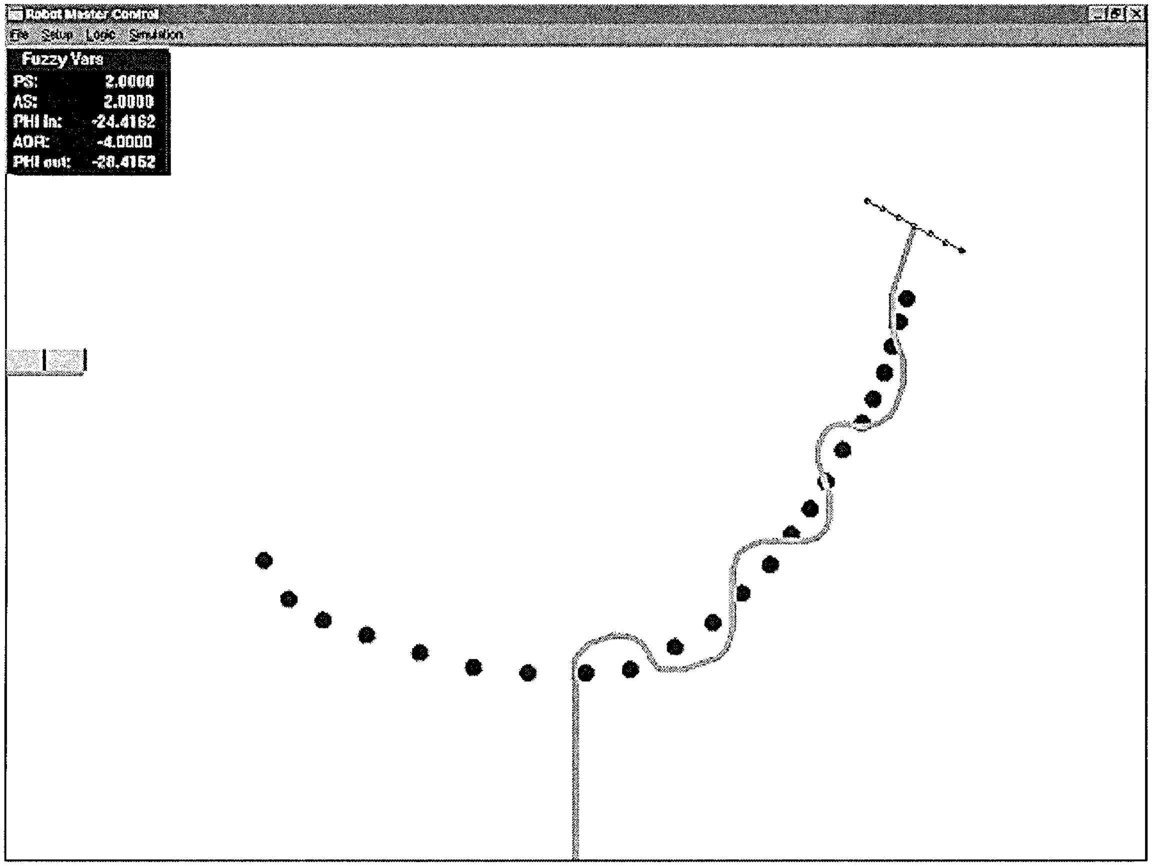

Fig. 12. Robot anchoring to the path.

of the most recent segment of the path, and, secondly, the ability of the robot to encounter a path, anchor on it, and follow it.

Fig. 11 illustrates how the controller loses the path at point A. At point B, it realizes it has lost the path, so the robot returns to point A and starts predicting the path based on the trajectory of the previous path segment. When it encounters the path at point C, it anchors on it and follows it. Fig. 12 illustrates how the controller anchors on a path when it encounters it.

VI. Conclusion

We conducted an experiment to prove that the Hall sensor array could be used to obtain a degree of membership in a fuzzy input control region. The Hall sensor array was designed and built. The Hall sensors were back biased with Samarium Cobalt magnets. The sensors were then able to detect the presence or absence of a ferromagnetic disk and produce an analog signal. Moving the array over the ferromagnetic disk produced this signal. This signal was fed through an amplifier to an analog-to-digital converter (ADC). The ADC had an eight-channel analog multiplexer (MUX) with an address input latch. The MUX provided the ability to feed in eight different analog inputs and the address input latch allowed one of the eight lines to be selected for input to the ADC. The ADC outputs eight lines of digital data. Four of the eight lines were brought in the CPU through the data register of the parallel port and the other four lines were brought in through the control register. This data was then displayed on the screen as vertical bars whose length was proportional to the strength of the signal. This way we were able to show the degree of membership obtained through the Hall sensor array when moved over ferromagnetic disks.

Further research of the controller should be conducted to further optimize its performance. This research should show the following.

a) The optimal relation between the size of the ferromagnetic disks and the Hall sensor array. For instance, in the current implementation only two Hall sensors will fire at the same time. By modifying the size of the ferromagnetic disks, more than two sensors can fire, thus modifying the number of rules that will be activated.
b) The optimal distribution of the Hall sensors along the x-axis. The density of the sensors should be higher around the origin.
c) The optimal path design (the optimal distance between the ferromagnetic disks).
d) The optimal design of the control and fuzzy regions (optimal number of fuzzy rules).

References

[1] R. Braunstingl, J. Mujika, and J. P. Uribe, “A wall following robot with a fuzzy logic controller optimized by a genetic algorithm,” in *Proc. IEE Int. Conf. Fuzzy Syst.*, 1995, pp. 77–82.
[2] H. Surmann, J. Huser, and L. Peters,, “A fuzzy system for indoor mobile robot navigation,” in *Proc. IEE Int. Conf. Fuzzy Syst.*, 1995, pp. 83–88.

[3] H. Roth and K. Schilling, "Control strategies for mobile robots based on fuzzy logic," in *Proc. IEE Int. Conf. Fuzzy Syst.*, 1995, pp. 89–96.
[4] C. Cavalcante, J. Cardoso, J. G. Ramos, and O. R. Neves, "Design and tuning of a helicopter fuzzy controller," in *Proc. IEE Int. Conf. Fuzzy Syst.*, 1995, pp. 1549–1554.
[5] M. Akutagawa, K. Yohsuke, and N. Hirofumi, "A neural measurement system for a moving object using magnetic sensors," in *Proc. IEE Int. Conf. Fuzzy Syst.*, 1995.
[6] I. J. Cox and G. T. Wilfong, *Autonomous Robot Vehicles*, New York: Springer-Verlag, 1990.
[7] Allegro MicroSystems, Inc., *Integrated and Discrete Semiconductors*: Data Book AMS-501, 1993.
[8] H. R. Everett, *Sensors for Mobile Robots*: A K Peters, Ltd., 1995.
[9] M. Abdelrahman, "Fuzzy sensors for fuzzy logic," *Control Eng.*, pp. 50–51, Dec. 1990.
[10] E. Cox, "Fuzzy fundamentals," *IEEE Spectrum*, pp. 58–61, Oct. 1992.
[11] B. Kosko and I. Satoru, "Fuzzy logic," *Sci. Amer.*, pp. 76–81, July 1993.
[12] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Inf. Sci.*, vol. 8, pp. 199–249, 1975.
[13] K. Self, "Designing with fuzzy logic," *IEEE Spectrum*, vol. 105, pp. 42–44, Nov. 1990.
[14] Hyper Logic Corporation, "CubicCalc: The third wave in intelligent software," 1990.

Kim M. Moulton photograph and biography not available at the time of publication.

Aurel Cornell photograph and biography not available at the time of publication.

Emil M. Petriu (M'86–SM'88–F'01) received the Dipl. Eng. and Dr. Eng. degrees from the Polytechnic Institute of Timisoara, Romania, in 1969 and 1978, respectively.

He is currently a Professor of electrical and computer engineering and Director of the School of Information Technology and Engineering (SITE), University of Ottawa, Ottawa, ON, Canada, where he has been since 1985. He was a Visiting Scholar at the Department of Applied Physics, Technical University of Delft, The Netherlands, in 1979 and 1982, Visiting Researcher at the Canadian Space Agency, 1992, and a Visiting Professor at the Research Institute of Electronics, Shizuoka University, Hamamatsu, Japan, 1994. His research interests include test and measurement systems, interactive virtual environments, intelligent sensors, robot sensing and perception, neural networks, and fuzzy control. During his career he has published more than 150 technical papers, authored two books, edited two other books, and received two patents.

Dr. Petriu is a Registered Professional Engineer in the Province of Ontario, Canada. He was elected a Fellow of the Engineering Institute of Canada, 2000, and Fellow of the Canadian Academy of Engineering, 2001. He is currently serving as Vice-President (Publications), a Member of the AdCom, and a Co-Chair of the TC-15 of the IEEE Instrumentation and Measurement Society. He is also an Associate Editor of the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT and a Member of the Editorial Board of the I&M MAGAZINE.