A Mathematical Fuzzy Logic Control Systems Model Using Rough Set Theory for Robot Applications

Aaron Don M. Africa
Department of Electronics and Communications Engineering, De La Salle University Manila
2401 Taft Avenue Manila Philippines 1004
aaron.africa@dlsu.edu.ph

***Abstract*—Robotics is an important technology that can aid humans in a variety of ways. These devices can help humans in fields like car production, military service, commercialized agriculture and space exploration. A robot is essentially a mechanical device that can be programmed to follow a specific set of instructions. One problem that is usually encountered in robotics is there are instances when its output is in degrees of truth of either 0 or 1. This is a challenge because robotic representations must not be limited to specific degrees of truth; it must have the ability to be represented in a wide variety of ranges. This research solves that problem by applying Fuzzy Logic Control Systems to robotic representations specifically in the delta-speed motor control area. This provides the Mathematical Model the ability to represent robotic representations of delta-speed motor control to a wide variety of ranges. The output of the Mathematical Model is in the decision rules that show the delta-speed motor control in a wide variety of ranges. Rough Set Theory is then applied to the model in order to turn the rules into nominal form to simplify the rules so that only the necessary parameters are needed to obtain the output. A prototype was created to demonstrate the Mathematical Model of this research with the aid of Lego Mindstorms EV3. The Lejos EV3 Java framework and MATLAB was used to program and demonstrate the Mathematical Model of the robot.**

***Index Terms*—Control Systems; Sensory Systems; Computer Engineering; Electronic Engineering; Fuzzy Logic; Rough Set Theory.**

I. INTRODUCTION

Robotics is a technology that is useful to humans. This technology was originally created to help humans. Robots can navigate in several ways [1]. One way for robots to move or navigate is by using wheels [2]. The movement of the wheels can be classified to several parameters like speed, the rate of change (delta) and motor speed [3]. The problem in these parameters is their output is not always exact. Unlike binary systems where the output is either 1 or 0 the output of robotic systems can be indistinct or fuzzy [4]. This research solves that problem by applying Fuzzy Logic theory. This theory is applied in the delta-speed motor control of a robot. A prototype was created in this research by using the Lego EV3. The Lego EV3 robot is attached with an ultrasonic sensory system. This sensory system is used to detect the distance of the robot to an object thereby aiding in the creation of the delta-speed motor control model. The Mamdani Fuzzy Logic interference was applied in the model [5]. The computer simulations of the delta-speed model were done by using the MATLAB software [6]. The Mamdani model produced the decision rules which can still be converted into nominal form. Nominal meaning that only the necessary variables are needed to produce the output. Rough Set Theory was used to make the rules in nominal form. The Rough Set Data Explorer (ROSE) [7] is the software used to apply Rough Set Theory in the fuzzy decision rules.

II. FUZZY LOGIC

Fuzzy Logic is one of the most successful technologies used for developing advanced systems like control systems [8]. The main reason is it addresses sophisticated applications by resembling human decision process. This theory has the ability to generate accurate solutions from approximate information [9]. The Fuzzy Logic theory fills the gap in engineering design from a purely mathematical process and logic approaches in system design. Design using Fuzzy Logic can accommodate ambiguities of language and logic in real world applications [10]. This theory provides a method in describing systems in human terms and has the ability to automate the system applications to effective models. Fuzzy Logic was first applied in industrial models like cement making. One of its major applications was when this theory was adapted in a Fuzzy Logic controlled subway in Sendai in the year 1987. Since then the Fuzzy Logic was applied on a large scale affecting some common household items [11]. This research applies Fuzzy Logic in robotics specifically in the delta-speed control of a robot. Fuzzy rules are outputted by the system and these rules were further nominalized using Rough Set Theory.

III. ROUGH SET THEORY

Rough Set theory is a new Mathematical tool that can be used for imperfect data analysis. This theory was developed by Z. Palwak in 1982 [12]. This theory is designed to handle imperfect knowledge or data [13]. The Rough Set Theory has been applied to several fields like banking, engineering, decision support etc [14]. This theory is based on the assumption that with every object in the universe of discourse some data is associated with it. Objects which are characterized by the same information are called indiscernible or similar [15]. This is in view of the presented information about them. The mathematical basis of the Rough Set Theory is the indiscernibility relation generated by the objects. Any sets of similar objects are called an elementary set. These elementary sets form a granule or atom of knowledge from the universe of discourse [16]. The elementary union of sets is called a crisp or precise set. These can also be referred to as vague or rough sets. Every rough sets have a line boundary where the objects can be classified. The rough sets have lower and upper approximations. The lower approximations have the objects that belong to a set while the upper approximations contain the objects that may belong to a set [17]. The main distinction of the lower and upper approximations depends on the boundary regions of the set. The main concept of Rough Set Theory are approximations [18].

IV. MATLAB

MATLAB or Matrix laboratory is an environment used for numerical computing. It is a programming language developed by Mathworks. This software allows algorithm implementations, matrix manipulations, data plotting functions etc. This software can be integrated to other programming languages like Java, Fortran, C++ etc. MATLAB has an additional add-on called Simulink. This add on can be used for model based design simulations for dynamic or embedded systems. The application MATLAB is created around the MATLAB scripting language [19]. This program can be used to as an interactive mathematical shell for scripting and executing text files which are created using the MATLAB code. MATLAB has several structure data types. This data types can be used as structure arrays for easier implementation. The software MATLAB supports object oriented programming this includes classes, packages, virtual dispatch etc. MATLAB has a built-in function called the Fuzzy Logic Tool box. This toolbox uses the Mamdani Fuzzy Interference. This interference produces fuzzy membership functions as its output and after the aggregation process, defuzzification is required for each fuzzy set [20]. This research used the Mamdani interference method to produce the Fuzzy Logic Control System model for representing robotic systems.

V. LEJOS

Lejos is a Java based firmware designed for Lego Mindstorms. This Java firmware replaces Lego Mindstorm's default software which is flowchart graphic based. Lejos is based on text command programming. Lejos can be used in both Lego NXT and EV3 variants [21]. The Lejos program allows the Lego Mindstorm robot to be programmed in a Java environment. Lejos provides access the robot's IC ports. This, in turn, lets Lejos command the different sensors of the robot like the touch, ultrasonic and sound sensors. The implementation details of the robot's actuator and sensors can be hidden behind multiple interfaces because of Lejos object oriented programming. This also allows the robot's programmer to work with advanced abstractions without interfering with hexadecimal addresses of the robot's hardware controller. The Lejos program can implement a Kalman filter and a PID controller. This program also provides support for mapping, navigation and behavioral robotics [22].

VI. REPRESENTATIONS OF DATA

The distance of the object, the rate of change of the distance and motor speed are converted into a membership function. The parameters will then be inputted to Lejos and MATLAB in order to create the membership functions. Tables 1, 2 and 3 shows the table representation of the membership functions of the distance of the object, rate of change and the motor speed.

Table 1
Membership Function of Distance of The Object

ID#	Fuzzy Range	Distance of object (cm)	Fuzzy Input
1	Very Close (VC)	10	-2, .1, 13, 18
2	Close (C)	15	13, 18, 23
3	Neutral (NE)	20	18, 23, 28
4	Distant (D)	25	23, 28, 33
5	Very Distant (VD)	30	28, 33, 43, 48

Table 2
Membership Function of The Rate of Change

ID#	Fuzzy Range	Rate of change in distance	Fuzzy Input
6	Negative Major (NMJ)	-100	-150, -120, -100, -50
7	Negative Minor (NM)	-50	-100, 50, 0
8	Zero (Z)	0	-50, 0, 50
9	Positive Minor (PM)	50	0, 50, 100
10	Positive Major (PMJ)	100	50, 100, 120 150

Table 3
Membership Function of The Motor Speed

ID#	Fuzzy Range	Motor Speed (% rotation)	Fuzzy Input
11	Very Slow (VS)	0	-25, -10, 0 ,25
12	Slow (S)	25	0, 25 ,50
13	Normal (N)	50	25 ,50 ,75
14	Fast (F)	75	25 ,50, 75
15	Very Fast (VF)	100	75 ,100 ,110 ,125

VII. DATA AND RESULTS

Figure 1: Picture of the Lejos robot

The data of the membership functions were inputted in the Lejos Robot and MATLAB. Figure 1 shows the picture of the robot on where the model was implemented.

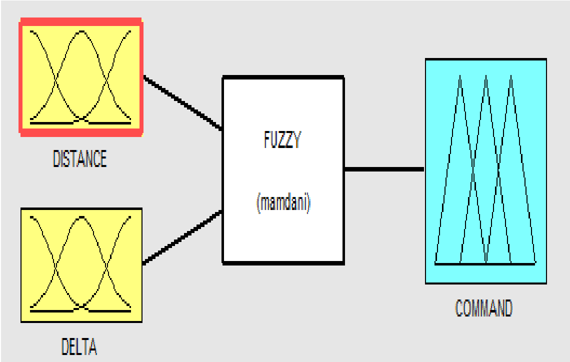

Figure 2: Input variables of the system

The Mamdani Fuzzy Interference was used to create the model. The inputs are distance and delta while the output is the motor speed command as shown in Figure 2.

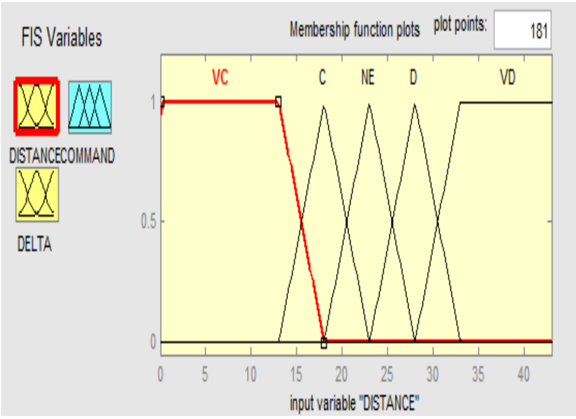

Figure 3: Plot of the membership function for the distance

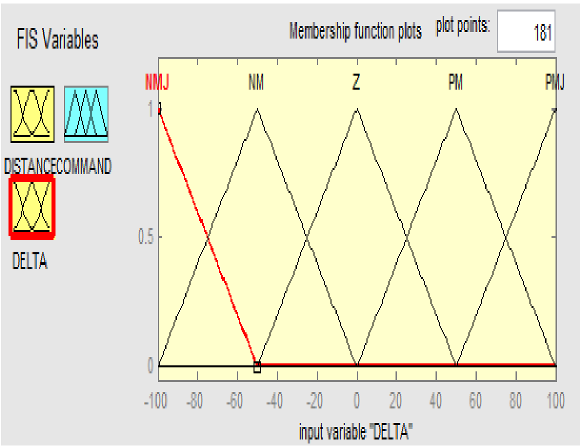

Figure 4: Plot of the membership function for the delta

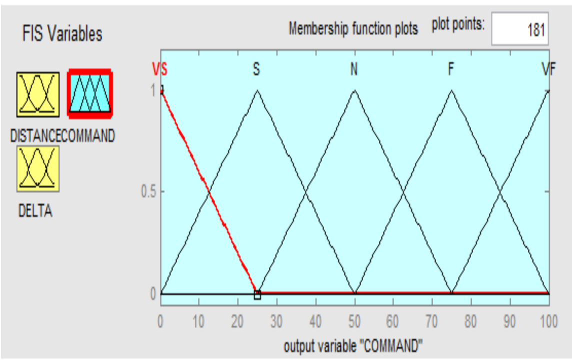

Figure 5: Plot of the membership function for the motor speed

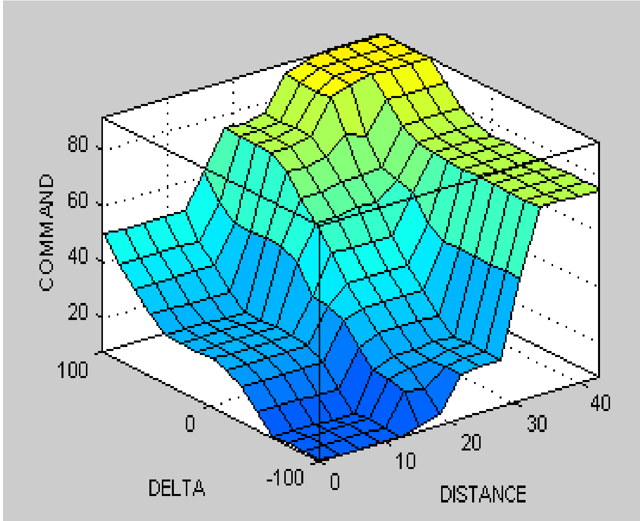

Figure 6: 3D model of the surface of the fuzzy system

Figures 3, 4 and 5 shows the plot of the membership functions for the distance, delta and motor speed. These parameters were plotted in 3D with the use of MATLAB's Fuzzy logic toolbox as shown is Figure 6.

Table 4
Decision Rules by the Fuzzy Model

#	Rule
1	(DIS = 1) & (DEL = 6) => (COM = 11)
2	(DIS = 1) & (DEL = 7) => (COM = 11)
3	(DIS = 1) & (DEL = 8) => (COM = 12)
4	(DIS = 1) & (DEL = 9) => (COM = 12)
5	(DIS = 1) & (DEL = 10) => (COM = 13)
6	(DIS = 2) & (DEL = 6) => (COM = 11)
7	(DIS = 2) & (DEL = 7) => (COM = 12)
8	(DIS = 2) & (DEL = 8) => (COM = 13)
9	(DIS = 2) & (DEL = 9) => (COM = 14)
10	(DIS = 2) & (DEL = 10) => (COM = 14)
11	(DIS = 3) & (DEL = 6) => (COM = 12)
12	(DIS = 3) & (DEL = 7) => (COM = 12)
13	(DIS = 3) & (DEL = 8) => (COM = 13)
14	(DIS = 3) & (DEL = 9) => (COM = 14)
15	(DIS = 3) & (DEL = 10) => (COM = 14)
16	(DIS = 4) & (DEL = 6) => (COM = 12)
17	(DIS = 4) & (DEL = 7) => (COM = 12)
18	(DIS = 4) & (DEL = 8) => (COM = 13)
19	(DIS = 4) & (DEL = 9) => (COM = 15)
20	(DIS = 4) & (DEL = 10) => (COM = 15)
21	(DIS =5) & (DEL = 6) => (COM = 14)
22	(DIS =5) & (DEL = 7) => (COM = 14)
23	(DIS =5) & (DEL = 8) => (COM = 14)
24	(DIS =5) & (DEL = 9) => (COM = 15)
25	(DIS =5) & (DEL = 10) => (COM = 15)

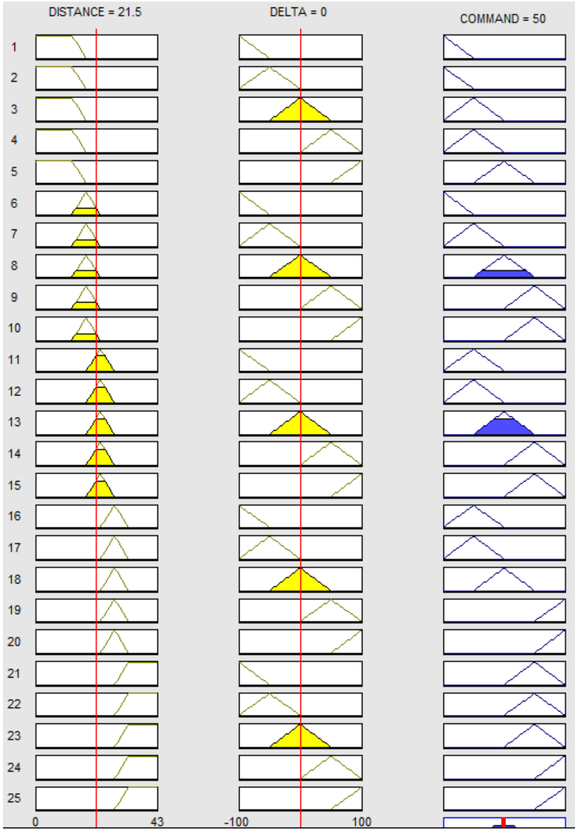

Figure 7: Fuzzy Interference rules as shown in MATLAB

MATLAB's Fuzzy logic toolbox produced 25 decision rules as shown in Table 4. Figure 7 shows the Fuzzy Interference rules as shown in MATLAB. These rules can be converted into nominal form meaning it will be simplified so only the necessary parameters are needed to produce the output. Table 5 shows the rules in nominal form with the use of Rough Set Theory. The software Rough Set Data Explorer (ROSE) was used to implement Rough Set Theory.

Table 5
Nominal Rules produced by Rough Set Theory

#	Rule
1	(DIS = 1) & (DEL in {6, 7}) => (COM = 11)
2	(DIS = 2) & (DEL = 6) => (COM = 11)
3	(DIS in {2, 3, 4}) & (DEL = 7) => (COM = 12)
4	(DIS = 1) & (DEL in {8, 9}) => (COM = 12)
5	(DIS in {3, 4}) & (DEL = 6) => (COM = 12)
6	(DIS in {2, 3, 4}) & (DEL = 8) => (COM = 13)
7	(DIS = 1) & (DEL = 10) => (COM = 13)]
8	(DIS = 5) & (DEL in {6, 7, 8}) => (COM = 14)
9	(DIS in {2, 3}) & (DEL in {9, 10}) => (COM = 14)
10	(DIS in {4, 5}) & (DEL in {9, 10}) => (COM = 15)

The rules produced using Rough Set Theory was simplified from 25 to 10. These rules are in nominal form meaning only the essential inputs are needed to determine the command. If for example one parameter cannot be obtained the correct command can still be determined.

VIII. Analysis of Data

Three membership functions are generated in this research these are the distance, delta and motor speed. These data are inputted in the Lejos EV3 robot and MATLAB's fuzzy logic toolbar. The membership function of the distance has 5 fuzzy ranges namely Very Close, Close, Neutral, Distance and Very Distant. The rate of change has 5 fuzzy ranges they are Negative Major, Negative Minor, Zero, Positive Minor and Positive Major. The output of the system also has fuzzy ranges namely Very Slow, Slow, Normal, Fast and Very Fast. The Mamdani Fuzzy interference method was used to defuzzify the membership functions and produced 25 rules. These rules are used to know the command of the motor speed in order to reach the object. The rules are converted into nominal form and are simplified by the use Rough Set Theory. From 25 the rules became 10 reducing it to 40% of its initial value.

IX. Conclusions and Recommendations

This research produced a novel Mathematical Fuzzy Logic Control Systems Model using Rough Set Theory. The output of this research can be applied in robots especially in the movement of its wheels specifically in the delta-speed motor control area. A robot prototype was built using Lego Mindstorms EV3. The model developed in this research was applied in the Lego Mindstorms EV3 robot and simulated in MATLAB. This research solved one of the most common problems in robotic systems in which the output are in degrees of truth of either 1 or 0. This is done by giving fuzzy ranges and membership functions to the output of the robot. The Mamdani interference method was applied to the fuzzy logic model. This interference method was applied with the use of MATLAB's fuzzy logic toolbox. A total of 25 rules was created. These rules were nominalized and simplified by using Rough Set Theory. From the original 25 rules, it becomes 10 rules, reducing it to 40% of its original value. This is useful because the rules needed to determine if the conditions were satisfied are reduced.

For future research, it is recommended that this model be tested in other fields like aircraft flight control systems. This model can help aircraft in their flight by reducing the fuzzy logic rules of the autopilot with the aid of Rough Set Theory.

References

[1] K. Hou, H. Sun, Q. Jia, and Y. Zhang,"An Autonomous Positioning and Navigation System for Spherical Mobile Robot," *Procedia Engineering*, vol. 29, no. 1, pp. 2556-2561, 2012.

[2] A. Zaki, O. Arafa, and S. Amer, "Microcontroller-based mobile robot positioning and obstacle avoidance," *Journal of Electrical Systems and Information Technology*, vol. 1, no. 1, pp. 58-71, 2014.

[3] A. Africa, R. Jamisola, C. Llorente and A. Montesines-Nagayo, "WALLy: A Fuzzy Controlled Mobile Robot for Soccer Competition," *5th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control*, Environment, 2009.

[4] J. Zhang, "Applications of fuzzy logic control in autonomous robot systems," *International Journal of Systems Science*, vol. 24, no. 10, pp. 1885-1904, 2007.

[5] E. Mamdani, and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.

[6] Fuzzy Interference. https://www.mathworks.com/help/fuzzy/what-is-mamdani-type-fuzzy-inference.html. 2016.

[7] ROSE. http://idss.cs.put.poznan.pl/site/rose.html. 2016

[8] C. Lim, "Implementation and experimental study of a fuzzy logic controller for dc motors," *Computers in Industry*, vol. 26, no. 1, pp. 93-96, 1995.

[9] L. Zadeh, "Fuzzy logic and the calculi of fuzzy rules, fuzzy graphs, and fuzzy probabilities," *Computers & Mathematics with Applications*, vol. 37, no. 11-12, pp. 35, 1999.

[10] T. Vitez, R. Wada, and A. Macario, "Fuzzy logic: Theory and medical applications," Journal *of Cardiothoracic and Vascular Anesthesia*, vol. 10, no. 6, pp. 800-808, 1996.

[11] Watkins. http://www.sjsu.edu/faculty/watkins/fuzzyjapan.htm. 2016
[12] Z. Palwak, "Rough Sets," . *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341-356, 1982.
[13] H. Zhang, W. Zhang, and W. Wu, "On characterization of generalized interval-valued fuzzy rough sets on two universes of discourse," *International Journal of Approximate Reasoning*, vol. 51, no. 1, pp. 56-70, 2009.
[14] S. Mahapatra, S. Sreekumar, and SS. Mahapatra, " Attribute selection in marketing: A rough set approach," *IIMB Management Review*, vol. 22, no. 1-2, pp. 16-24, 2010.
[15] K. Kaneiwa, and Y. Kudo, "A sequential pattern mining algorithm using rough set theory," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 881-893, 2011.
[16] E. Lashin, A. Kozae, A. Khadra and T. Medhat, "Rough set theory for topological spaces," *International Journal of Approximate Reasoning*, vol. 40, no. 1-2, pp. 35-43, 2005.
[17] Y. Yao, "Probabilistic rough set approximations," *International Journal of Approximate Reasoning*, vol. 49, no. 2, pp. 255-271, 2008.
[18] Y. Yao, and B. Yao, "Covering based rough set approximations," *Information Sciences*, vol. 200, no. 1, pp. 91-107, 2012.
[19] MATLAB. https://www.mathworks.com/. 2016
[20] MATLAB Fuzzy Logic Toolbar. https://www.mathworks.com/products/fuzzy-logic/. 2016
[21] Lejos. http://www.lejos.org/. 2016
[22] Lejos Mapping. http://www.lejos.org/ev3/docs/lejos/robotics/mapping/package-summary.html. 2016.