IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Comparative Study on Machine Learning Algorithms for the Control of a Wall Following Robot

Dhaya Sindhu Battina

Sr. Data Engineer & Department of Information Technology

USA

Abstract— *The main aim of this paper was to compare machine learning algorithms used in controlling a wall-following robot. A robot seldom has access to a complete and precise representation of its surroundings; thus, it must often operate under uncertainty. Even the physical properties of a robot might be questionable when used in real-world applications since such attributes can change significantly in different ways [1]. The motors and the limbs may be destroyed or damaged, the sensors may be obstructed, or the components may be reassembled improperly after repair. These are all possible outcomes [1]. Following the execution of a given motor order, a robot may be unable to make accurate predictions about its location. For want of a better term, it does not understand the plant equation, which is the equation that explains the consequence of acts that have been performed on it. When it comes to robot recovery after its physical design has been modified, we are particularly interested in establishing an effective technique.*

Keywords: *Machine learning, wall-following robot, software development, Mobile Robot, Robot Control*

I. INTRODUCTION

Mobile robots, particularly wall-following robots, have been successfully used in a variety of industries, including manufacturing and the office. It is often used to mitigate possible dangers [1] and [2]. This robot operates in such a way that it can follow the outlines of objects in the environment, such as walls and barriers, and it may also be paired with more sophisticated behavior to execute high-level tasks. There are a variety of elements connected to the navigation job that has drawn the attention of various academics to the wall following robot. It has been shown by the many occurrences of various sorts of strategies that have been used to increase the capabilities of the wall-following robot. For example, receiving and analyzing sensor data and selecting when to activate the robot [1], [2] demonstrate that there has been an improvement. The major goal of upgrading the wall following the robot is to keep it on the path that has been set for the robot. As soon as the door is shut, the robot begins to back away from the wall. On the other hand, when the robot is far away from the wall, it creeps closer to it. It is, however, difficult to determine the appropriate value of the speed for the turn motion. As a result, the wall-following robot travels wavy as a result of the incorrect positioning. As a result, a specific controller is often used to resolve this problem. Different industries, including nuclear power plants, petroleum refineries, chemical production, and military surveillance, employ fully autonomous mobile robots nowadays. Autonomous mobile robots have a significant impact on process control [2]. These robots must be able to navigate by tracking a wall, which is one of its most important duties. Several tasks, including fault identification, search and rescue, and the identification of fractures in oil pipelines, may benefit from the application of wall following [3]. The ability to manage these robots with great accuracy is critical for the activities that they are designed to do. Any minor inaccuracy in the precision of the measurement might be expensive, and it could result in the examination of a specific section of the wall or pipe being missed. Numerous publications have offered various control approaches for wall following systems. Using a hexapod robot to demonstrate the concept, the design in [3] employs a data-driven fuzzy controller learned by dynamic optimization to operate the robot.

The wall following algorithms has been the subject of many works, including [1-3], which have presented various control approaches. Using a hexapod robot to demonstrate the concept, the architecture in [3] employs a data-driven fuzzy controller learned by dynamic optimization to operate the robot. This technology uses ultrasonic sensors to assess the sound pressure levels and then document the path the sound was traveling in. Consequently, a problem that is not linearly separable developed as a consequence of this. So [3] suggested a number of different neural network architectures, such as the Multi-Layer Perceptron (MLP) as well as the Elman Recurrent Network (ERN). Researchers from [4] have shared the datasets for the sensor data, as well as the robot's path based on those values, in the repository [3,4]. Several additional research articles published after

[4,5] have offered other architectures for the controller based on the dataset, including the concepts in [4-5] and [6]. The state-of-the-art in many sectors has witnessed a tremendous improvement as a result of recent developments in machine learning technology. Machine learning is now being applied in a variety of applications, including image identification and characterization, text classification, self-driving vehicles, diagnostics, and fraudulent financial prevention. This study will demonstrate how machine learning may be used to develop an incredibly precise controller-based robot. Decision Trees (DT) are a simple and popular machine learning tool. To enhance DT for specific tasks, ensemble learning, and boost approaches might be used. In this case, Random Forest Classifier (RFC) and Gradient Boosting Classifier (GBC) are types of classification algorithms [7]. This paper will go into great depth on the different machine learning methods that are used to operate Wall Following Robots.

II. PROBLEM STATEMENT

The main problem that this paper will address is to explore and compare the various ML algorithms used in controlling Wall Following Robots. The use of fully autonomous and intelligent robots has the potential to reduce human liability while also expanding automation into domains such as public services and the military. Disaster response, medical services, emergency preparedness, and lifesaving activities are among the services that artificial intelligence scientists are aiming to automate via the use of autonomous robotic vehicles [8]. One of the difficulties that these robots must confront is the ability to properly identify and avoid impediments such as debris, fire, traps, and other hazards. Unmanned aerial vehicles (UAVs) are electromechanical devices that may be programmed to do a variety of tasks. Moving and lifting items, receiving information about temperature and humidity, and following walls are some of the responsibilities that they are tasked with [8]. A well-designed autonomous robot must be adaptive enough to regulate its own activities from the standpoint of system engineering. Furthermore, it must be able to do the duties required precisely and correctly.

III. LITERATURE REVIEW

A. Wall following robot

A wall following robot is a device that is programmed to navigate along a wall without colliding with it. It is equipped with obstacle detection sensors located on the body, which detect the presence of a wall and drive DC motors linked to the wheels, allowing the robot to continue traveling along with it [8]. Designing the robot to be right or left orientated, or even to follow either side, is entirely up to the individual designer. An easily constructed right- or left-orientated wall follower robot can be done using just two sensors. Although more sensors may be utilized in the construction of such a robot, the routing accuracy of the robot will eventually increase as a result of this [9]. In order to create a wall follower that can move in any direction, it is necessary to employ at least three sensors, and the program logic is quite complicated and advanced. If the robot is built to follow a right-oriented wall, the obstacle detection sensors must be positioned on the robot's front and right sides, respectively [10]. If a robot with a left-oriented wall follower is constructed, the obstacle detector sensors must be installed on the robot's front and left sides, respectively. If the robot is intended to go in any direction, obstacle detector sensors must be installed on the robot's front, left, and right sides. Obstacle detection and avoidance system must be included in the wall following the robot's design [11]. It is necessary to include many sensors in the construction of such a robot, including bump sensors, infrared sensors, ultrasonic sensors, and other similar devices. By attaching these sensors to the robot, it will be able to gather information about the surrounding environment. Because it is modest in cost and has a reasonably long range, an ultrasonic sensor is an excellent choice for collision avoidance in a slow-moving autonomous robot [11].

B. ML Control systems for a Wall Following Robot

i. Fuzzy Logic Controller (FLC)

FLC may be configured and assigned to operate in a closed-loop fashion. That is, the product of decision-making is immediately utilized as feedback for the subsequent stage. The FLC operates on the same principles as a Proportional Integral Derivative (PID) controller in its operation [11]. Used as a wall-following robot controller, both of these devices frequently rely on the error, which represents the values between the setpoint (zero-error) and the actual error provided by the distance sensor, to perform their functions properly. Fuzzy Logic Control (FLC) is a control design in which a decision is reached by the application of a fuzzy interference system based on rules or information that comprises the string of if-then fuzzy rules [12]. As a key influencer, the existence of these controllers is vitally beneficial to the organization. It shifts the emphasis away from conventional controls that are concerned with robot sensor accuracy and towards current controls that are concerned with decision making in conditions of high uncertainty, complexity, and nonlinearity. Fuzzy Logic Control (FLC) is a control design in which a decision is reached by the application of a fuzzy interference system based on rules or information that comprises the string of if-then fuzzy rules [13]. As a key influencer, the existence of these controllers is vitally beneficial to the organization. It shifts the emphasis away from conventional controls that are concerned with robot sensor accuracy and towards current controls that are concerned with decision making in conditions of high uncertainty, complexity, and nonlinearity. The popularity of FLC may be attributed to numerous kinds of mobile robot behaviors that have been effectively enhanced, such as path-following, obstacle-avoiding, and goal-seeking robots, all of which are gaining in popularity. The configuration of the input membership function has a significant impact on the success of FLC when used in conjunction with other techniques [13]. Traditionally, it has been modified manually by simply linking the input and output membership functions with the width of all the subsets in the input and output membership functions. Using the premise that each representative function's range is known, the analysis is carried out. Nevertheless, manual configuration is no longer suggested due to the difficulty in determining the value of the input that corresponds to a certain value of output [13]. Consequently, numerous approaches have been developed that are based on the heuristic-based approach that is often used. The Genetic Algorithm is a well-known name for one of these algorithms (GA) [14].

For the reasons that have been addressed in this work, the role of GA is employed to increase the ability of FLC [14]. As a result of the random input, FLC is addressed in order to create the exact value of the output. An FLC is used as the controller in order to maintain a safe distance between the robot and the surrounding wall. After that, the FLC analyses the robot's starting distance from the wall and determines its speed using the initial length and feedback value [14]. In this case, the value is referred to as an error, and it is obtained by referring to the disparity between the actual distance and the planned distance or setpoint. Because the mistake is represented by the variables of the input membership function, the layout of the error is handled automatically by the GA algorithm. Initially, a random number generator is used to create some numbers that reflect the values of the mistake. With regard to the greatest and lowest values of the input membership function, it is assumed in certain configurations that they are equal [14]. All of the resulting data are then transformed to binary values in order to make the process of mutation and crossover as simple and straightforward as possible. They are assessed on a continuous basis by using the fitness function. A function that represents the sum of errors is called the error summation function (also known as the summation function). This is in addition to referring to the last phase of GA, which is when a new population is formed via mutation and crossover [14]. These two distinct capabilities, referred to as the regular and optimized performance levels, are compared in the experiment against one another. They are simulated and compared to one another. According to the results of the comparison, the suggested strategy has the potential to considerably increase the performance of the wall following the robot, as shown. If the suggested approach is compared to a traditional method, it can be said that it is more stable and accurate [15].

a. Implementation of a Fuzzy Logic Controller

It has already been established that the goal of maintaining the robot on the required course at all times is achieved by changing the right linear velocity, and in this project, the Fuzzy Logic Controller is responsible for taking care of this task. Because it is the closed-loop controller, it is participating in the process. FLC is, theoretically, a problem mechanism system inspired by the notion of a human expert deciding [15] [15, 16]. FLC is distinguished by the widespread use of linguistic phrases to describe specific values that are processed before being given to the membership function. The layout of the input membership function is critical to the efficacy of its use in this context. And, in order to achieve the right adjustment, the fuzzy set theory must be taken into consideration [16]. In general, the following procedures may be used to construct the FLC's architectural framework. The first step is fuzzification, which turns the chirp data into a linguistic word and organizes the membership function in a logical manner. The second kind of inference is fuzzy inference, which combines the membership function and rules base and assumes relation IF-THEN, and it is referred to as fuzzy inference. Lastly, defuzzification, which is the process by which the linguistic data is converted back into the chirp data [16], is performed.

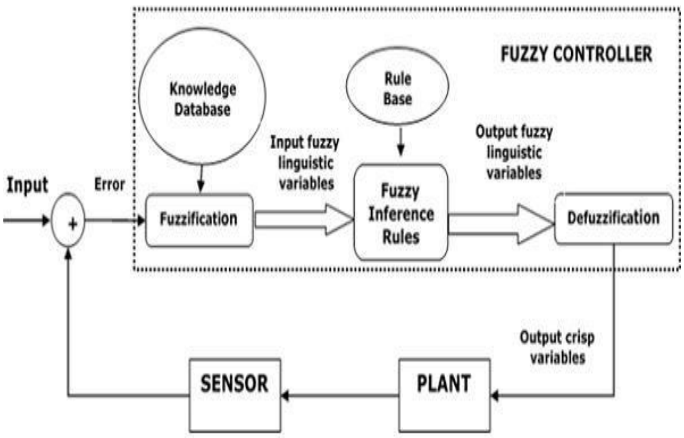

Figure i. Flowchart of Fuzzy Logic Control system

ii. Proportional Integral Derivative (PID) Controller

Control algorithms based on proportional-integral-derivative (PID) principles are the most widely employed in industry, and they have been generally recognized in the field of industrial control. While PID controllers are popular for their robustness and ease of use in a broad variety of operating circumstances, it's also because of the fact that they can be used in an intuitive way by engineers [16]. In order to produce the best response, the PID algorithm utilizes a variety of different combinations of the 3 major correlation coefficients: proportional, integral, and derivative. Several topics are covered in this work, including closed-loop control systems, classical PID theory, and the impacts of tweaking a closed-loop control system [16,17]. The PID toolset in LabVIEW, as well as the simplicity with which these VIs may be used, are also covered. Control outputs are adjusted accordingly by Proportional Integral Derivative (PID) controllers depending on the ratio between a set point (SP) and a measured process variable (PV).

There are some similarities between PID and proportional control, however, PID has algorithm components that are dependent on integral and derivative error data. Instead of responding just to the present error value, the algorithm now incorporates a component of historical context into its operation [17]. It is recommended that one uses a PI or PID controller in non-integrating operations, which are defined as any process that ultimately returns to its original outcome provided the same set of input parameters and perturbations. In order to integrate operations, a P-only controller is the most appropriate choice. Integral action is utilized to eliminate misalignment and is similar to adjustable ubias ubias [17].

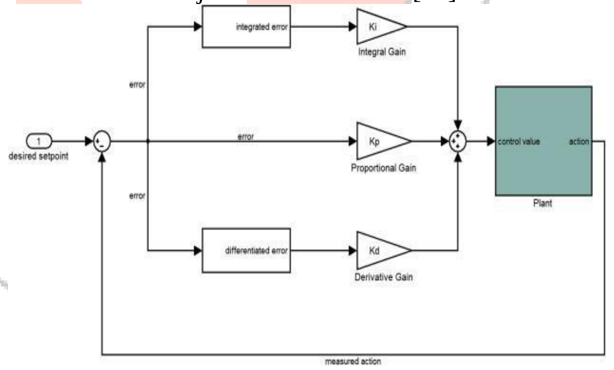

Fig ii: PID control system

C. Comparison of ML algorithms used in Controlling a Wall Following Robot

i. Decision Tree

It is possible to address regression and classification issues using a decision tree method. With an inverted tree, a homogeneously distributed root node branches out to extremely heterogenous leaf nodes, with the output being derived from this structure. For continuous-valued dependent variables, regression trees are employed; for discrete-valued dependent variables, classification trees are utilized [18].

Basic Theory: Each node in the decision tree has a condition over a feature, which is obtained from the independent variables in the decision tree. On the basis of the condition, the nodes choose which node they will traverse to next. When the leaf node is reached, a projected output is generated. The tree is efficient when the requirements are met in the proper order. Nodes are selected based on their entropy/information gain, which is measured as a function of time. The tree structure is created by the use of a recursive, greedy-based method.

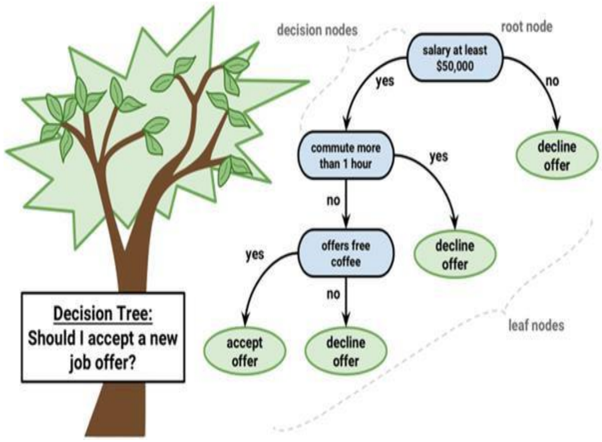

Fig iii: Illustration of a Decision tree

Advantages:

• There is no requirement for data preparation.
• There are no assumptions about the distribution of data.
• Colinearity is dealt with effectively.
• Decision trees have the ability to offer an intelligible explanation for a forecast.

Disadvantages include:

• There is a risk of overfitting the model if we continue to grow the tree in order to obtain high purity. This problem may be resolved by using decision tree pruning techniques.
• Likely to be exposed to outliers.
• When working with large and complex datasets, the tree may become quite complex.

ii. K-nearest neighbors

Using K-nearest neighbors, a non-parametric classification and regression approach may help identify relationships between variables. As far as machine learning techniques go, this is one of the simplest to utilize. The local approximation is used in a lazy learning model.

Theoretical Framework:

Essentially, KNN works on the premise that the test data point is similar to your area, so you assume it is [18]. When using KNN, we search for k nearby objects and make a forecast. A majority vote is used in KNN classification, but in KNN regression, the result is the mean of the k closest data points. We choose k as an odd number as a rule of thumb. KNN is a computationally inefficient learning paradigm [18].

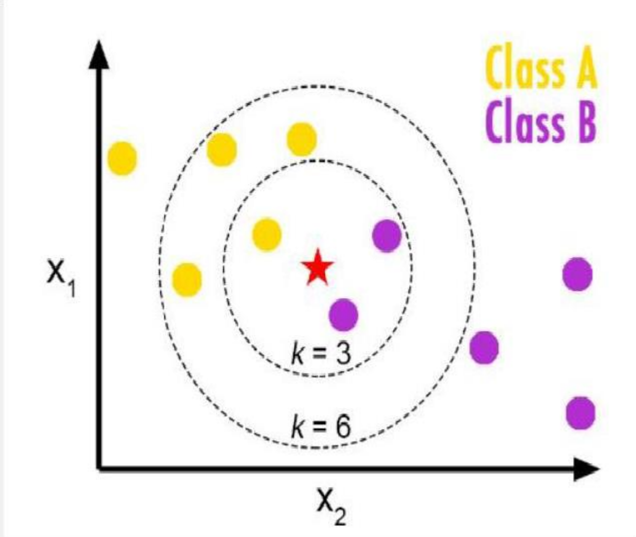

Fig ii: Illustration of a basic theory for K-nearest neighbors

Advantages

• Simple and straightforward machine learning model;
• Few hyperparameters to modify.

Disadvantages:

• If the sample size is big, the calculation cost will be high;
• Appropriate scaling should be supplied to ensure that all characteristics are treated equally.

iii. Logistic Regression Model

This method is a machine learning algorithm that examines the connection between dependent and independent variables, similar to linear regression, but with the primary distinction being data categorization. Although the approach is named regression, it is utilized for classification rather than estimation. Consequently, binary classification is another name for logistic regression [18]. Because of its basic operation structure, logistic regression has a low variance and is less prone to overfitting. Regression using Logistic Equations Logistic regression, like linear regression, is an excellent place to begin learning about classification methods. A regression model may appear; however, this is really a categorization model. a binary output model is built around a logistic function. Using the logistic regression's output, which is a probability (0x1), we can use to forecast whether the binary output would be 0 or 1 [18].

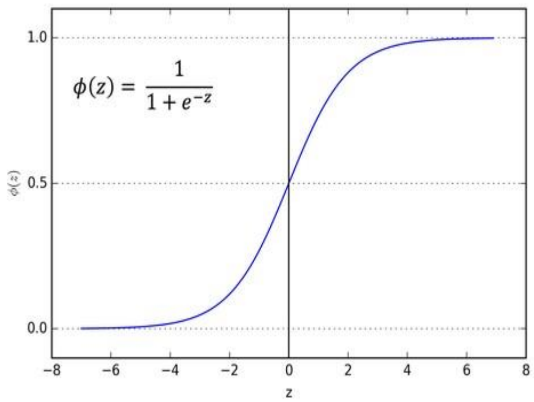

Fig iii: Logistic Regression Model

Theoretical Framework:

When it comes to behavior, logistic regression resembles linear regression rather closely. The linear output is likewise calculated, and then a stashing function is applied to the regression result. It's common to utilize the sigmoid function as a logistic function. Clearly, the z value is identical to the linear regression result in Eqn as seen below (1) [18].

Advantages

• Quick, easy, and straightforward classification approach.
• Parameters describe the direction and magnitude of importance of independent factors over the dependent variable.
• Multiclass classifications are also an option. There is always a loss function that is convex.

Disadvantages:

• It is not applicable to issues involving non-linear classification.
• Proper feature selection is necessary. Assumptions are for a good signal-to-noise ratio
• The presence of colinearity and outliers reduces the LR model's accuracy.

IV. FUTURE IN THE U.S

The future of machine learning in the U.S will continue to grow from the current status. The use of machine learning solutions is increasing in our everyday lives as they continue to integrate improvements into organizations' essential operations. With machine learning now powering everything from Netflix's recommendation system to self-driving vehicles, companies should start paying attention [18]. In this post, we'll talk about machine learning's future and how it may benefit a variety of businesses. This crucial area of artificial intelligence, a subdiscipline, has gotten a lot of attention recently, thanks to both its scientific advances and its rich job prospects. It's no secret these days that search engines depend on machine learning to keep their services up to date. As a result of putting ideas into practice, Google has developed some remarkable new services. As an example, consider speech recognition with picture search [18]. Many of Google's services, such as picture search and translation, make use of cutting-edge machine learning techniques that allow computers to determine, listen, and talk in much the same manner that humans can. The phrase "machine learning" refers to the most cutting-edge uses of AI currently being developed. Time will tell how they continue to develop new and exciting features.

V. ECONOMIC BENEFITS IN THE UNITED STATES

There are many economic benefits of ML algorithms in the U.S market. The adoption of ML in-wall controller robots is shaping the manufacturing industries to develop products that can operate independently. One reason these professions are so profitable is the scarcity of qualified candidates with machine learning expertise. A 34 percent growth rate and a typical pay of $146,085 per year are cited by employment website Indeed.com, which ranks machine learning engineer as the best job in the United States [18]. Overall, employment in computer and information technology is expected to expand by 11% between 2019 and 2029. This report forecasts that artificial intelligence would generate 12 million new employment in 26 nations by 2025 (indicating a net gain of 97 million new jobs produced while 85 million people lose their jobs due to the displacement caused by artificial intelligence) [17,18]. Because of the strong demand for machine learning skills, most occupations pay well over $100,000, with some paying up to $200,000, such as machine learning engineers. Machine learning is still being used by manufacturers at a very early level. Only 9% of survey participants expect to use AI in their business activities by 2020. It's possible to monitor equipment performance and condition using machine learning technologies. These tools may also be used to anticipate product quality and estimate energy usage. Machine learning advances mean that more robots will be found in industrial facilities in the not too distant future. Self-driving car firms like Tesla, Waymo, and the American Honda Motor Company are all looking into how to make their vehicles safer for drivers and passengers [18]. Partially automated vehicles have already been shown off by manufacturers, but completely self-driving automobiles are still decades away from commercialization. One of the most important technologies for making these visions a reality is machine learning.

VI. CONCLUSION

This paper presented a comparison of the performance of various machine learning models for a wall following a robot controller. The findings show that ML has been tremendous in improving the controller robots and increase in the manufacture of automatic devices. For instance, mobile robots, particularly wall-following robots, have been successfully used in a variety of industries, including manufacturing and the office. It is often used in order to mitigate possible dangers. This robot operates in such a way that it can follow the outlines of objects in the environment, such as walls and barriers, and it may also be paired with more sophisticated behavior to execute high-level tasks. Digital marketing teams all around the world are using machine learning. It makes personalization more relevant. In this way, businesses may connect and communicate with their customers. The FLC and PID are the primary control systems for a wall-control robot, according to this study's conclusions. FLC receives the sensor data and uses them to regulate the speed of two DC motors. To move the robot, differentiating the speeds of the two motors is used Results from the experiments reveal that FLC is effective in guiding the robot down a guidance line by following a wall, and it outperforms the PID controller. The process of creating a model of the whole robot's movements starts at the bottom and works its way up. When compared to a PID controller, the FLC has better reaction times. There is less overshoot, shorter settling time, and less inaccuracy with the FLC controller than the PID controller. That is to say, the FLC controller is more suited for operating a wall following robot than the previous model.

REFERENCES

1. D. Simon, "Analyzing Control System Robustness," *IEEE Potentials,* Vol. 21, No. 1, 2002, pp.16-19.
2. B. Davies and W. E. R. Davies, "Practical Robotics: Principles and Applications," Prentice Hall, Upper Saddle River, 1997.
3. Z. Gao, "From Linear to Nonlinear Control Means: A Practical Progression," *ISA Transactions*, Vol. 41, No. 2, 2002, pp. 177-189. doi:10.1016/S0019-0578(07)60077-9.
4. Z. Gao, Y. Huang and J. Han, "An Alternative Paradigm for Control System Design," *Proceeding of IEEE Conference on Decision and Control,* Orlando, 4-7 December 2001, pp. 4578-4585.
5. D. Ratner and P. McKerrow, "Navigation an Outdoor Robot along Continuous Landmarks with Ultrasonic Sen- sing," Robotics and Autonomous Systems, Vol. 45, No. 1, 2003, pp. 73-82. doi:10.1016/S0921-8890(03)00096-4.
6. R. Carelli and E. O. Freire, "Navigation Outdoor and Wall-Following Stable Control for Sonar-Based Mobile Robots," Robotics and Autonomous Systems, Vol. 45, No. 3-4, 2003, pp. 235-247. doi:10.1016/j.robot.2003.09.005
7. R. Benporad, M. Di Marco and A. Tesi, "Wall-Following Controller for Sonar-Based Mobile Robots," Proceeding of IEEE Conference on Decision and Control, San Diego, 10-12 December 1997, pp. 3063-3068.
8. P. M. Peri, "Fuzzy Logic Controller for an Autonomous Mobile Robot," Master Thesis, Cleveland State Univer- sity, Cleveland, 2005.
9. R. Braunsting, J. Mujika and J. P. Uribe, "A Wall Fol- lowing Robot with a Fuzzy Logic Controller Optimized by a Genetic Algorithm," *Fuzzy Systems*, Vol. 5, 1995, pp. 77-82.
10. M. Veerachary and D. Sharma, "Fuzzy Incremental Con- troller for the 3rd Bulk Converter," IEEE Proceeding of International Conference on Power Electronics and Drive Systems, Bangkok, 27-30 November 2007, pp. 768-771.
11. R. Carelli and E. Freire, "Stable Corridor Navigation Controller for Sonar-Based Mobile Robots," Technical Report, INAW, Universidad Nacional de San Juan, San Juan, 2001.
12. R. Malhotra and A. Sarkar, "Development of a Fuzzy Logic Based Mobile Robot for Dynamic Obstacle Avoidance and Goal Acquisition in an Unstructured Enviroment," IEEE/*ASME Proceeding of International Conference on*

Advanced Intelligent Mechatronics, Kobe, 20- 24 July 2003, pp. 235-247.

13. G. McComb and M. Preoko, "The Robot Builder's Bonanza," 3rd Edition, McGraw Hill, New York, 2006.
14. O. J. Sordalen, "Feedback Control of Nonholonomic Mo- bile Robots," Ph.D. Thesis, Department of Engineering Cybernetics, The Norwegian Institute of Technology, 2006.
15. B. Cai and D. Konik, "Intelligent Vehicle Active Suspen- sion Control Using Fuzzy Logic," *Proceeding of IFAC World Congress,* Vol. 2, 1993, pp. 231-236.
16. Z. Yi, H. Y. Khing, C. C. Seng and Z. X. Wei, "Multi-Ultrasonic Sensor Fusion for Mobile Robots," *IEEE Proceeding of the Intelligent Vehicle Symposium,* Dearborn, 3-5 October 2000, pp. 387-391.
17. I. Skrjanc and D. Matko, "Predictive Function Control Based on Fuzzy Model for Heat-Exchanger Pilot Plant," *IEEE Transaction on Fuzzy Systems,* Vol. 8, No. 6, 2000, pp. 705-712. doi:10.1109/91.890329.
18. D. Hanafi, M. N. M. Than, A. A. A. Emhemed, T. Muly- ana, A. M. Zaid and A. Johari, "Heat Exchanger's Shell and Tube Modeling for Intelligent Control Design," *IEEE Proceeding of 3rd International Conference on Communication Software and Networks,* Xi'an, 27-29 May 2001, pp. 37-41.