An Undergraduate Fuzzy Logic Control Lab Using a Line Following Robot

DOGAN IBRAHIM, TAYSEER ALSHANABLEH

Faculty of Engineering, Department of Computer Engineering, Near East University, Lefkosa, Mersin 10, Turkey

Received 9 November 2008; accepted 18 March 2009

ABSTRACT: Fuzzy logic controllers have gained popularity in the past few decades with highly successful implementation in many fields. Fuzzy logic enables designers to control complex systems more effectively than traditional methods. Teaching students fuzzy logic in a laboratory can be a time-consuming and an expensive task. This paper presents a low-cost educational microcontroller-based tool for fuzzy logic controlled line following mobile robot. The robot is used in the second year of undergraduate teaching in an elective course in the department of computer engineering of the Near East University. Hardware details of the robot and the software implementing the fuzzy logic control algorithm are given in the paper.
 Comput Appl Eng Educ; Published online in Wiley InterScience (www.interscience.wiley.com); DOI 10.1002/cae.20347

Keywords: fuzzy logic controller; robotics; control systems; line following robot; educational tool

INTRODUCTION

Traditional methods to control any dynamic system require the use of some knowledge, or *model*, of the system to be controlled. An accurate model is crucial for the successful implementation of a control algorithm. Unfortunately, most systems in real life are nonlinear, highly complex, and too difficult or impossible to model accurately. Fuzzy logic, a mathematical system developed by Zadeh [1], helps to reduce the complexity of controlling nonlinear systems. Fuzzy logic expresses operational laws of a control system in *linguistic* terms instead of the traditionally used mathematical equations. The linguistic terms are most often expressed in the form of logical expressions, such as IF−THEN. For example, considering a temperature control system one can write

> IF room_temperature is WARM, THEN set fan_speed to MEDIUM

It is important to notice that the terms WARM and MEDIUM do not define single discrete values, but rather a range of values. For example, WARM could define the temperature range 20−25°C, and MEDIUM could define the speed range

Correspondence to D. Ibrahim (dogan@neu.edu.tr).

500−800 rpm. The range of input values are known as membership functions and are usually shown graphically.

It is interesting to notice that a traditional non-fuzzy controller would probably implement the above control function as

> IF room_temperature >25°C THEN set fan_speed to 800 rpm

Perhaps the main difference between the two control functions is that in the non-fuzzy controller the output is activated after the input reaches an exact value, and small changes about this point can result in major changes in the output value. In the fuzzy controller, on the other hand, a range of input values result in the same output, thus eliminating wild changes in the output values.

Fuzzy control is used in many commercial, domestic, and automotive control applications. A cross-section of applications that have successfully used fuzzy control includes

- Washing machines.
- Vacuum cleaners.
- Refrigerators.
- Microwave ovens.
- Hi-Fi systems.
- Televisions.
- Photocopiers.
- Vehicle climate control.
- Vehicle seat and mirror control systems.

The number of commercially available educational fuzzy logic control laboratory kits is rather limited and the existing ones are either too expensive or complex. There are many sites on the Internet related to the fuzzy logic and fuzzy logic controllers. Some of the available educational tools are software packages only, usually in the form of simulation. A well-known commercial software package in this field is the MATLAB, developed by Mathwork, Inc. [2] and offering a fuzzy logic toolbox for the design and simulation of fuzzy logic control systems. ARISTOTLE, developed by Fuzzy Logic Laboratorium Linz [3], Togai Infra Logic, developed by Ortech Engineering, Inc. [4], and FULDEK, developed by Bell Helicopter Textron, Inc. [5] are some of the fuzzy logic software packages.

There are also commercially available fuzzy logic educational laboratory kits in the form of hardware, and these kits offer practical hands-on experience to students. G.U.N.T. [6] in collaboration with the Department of Automation and Information Technology at the Harz University of Applied Studies and Research [7] has developed a ball-and-beam experiment based on a microcontroller implementing fuzzy algorithms. A fuzzy control is used to attempt to hold a ball in a specific position on a beam by tilting the beam, even when the position of the ball is modified by external influences. The aim of this experimental unit called the RT 121 has been to teach the fundamentals of fuzzy logic control and microcontroller technology. Although the RT 121 is an excellent fuzzy logic experimental and learning kit, it has the disadvantage that the kit is rather expensive.

CE124 is a fuzzy logic trainer kit developed by Megachem Co Ltd [8]. Students learn by physically connecting fuzzy blocks together in the same way as when learning Boolean logic or when connecting operational amplifiers in an analogue computer. The kit is supplied with a manual that contains a series of structured experiments, which lead the student to progress from an introduction into fuzzy principles and circuits to real-time fuzzy control. The experiments cover the topics of fuzzy membership, defuzzification, fuzzy logic operators, and case studies are provided for fuzzy washing machine cycle and fuzzy camera exposure compensation. In addition, the kit can be used as an external fuzzy control system for other products of the company, such as thermal process control, coupled tasks, and servo control trainer. The CE124 can also be connected to a PC using the supplied icon-based software package. Although the CE124 allows rapid implementation, practical use, and learning of fuzzy logic control systems, the cost of the kit is rather high.

Akcayol et al. [9] describe the development of an educational tool for fuzzy logic controllers and classical controllers, aimed for undergraduate and graduate level students. The tool was designed with the aim that students could establish a thorough understanding and be able to compare the fuzzy logic controllers with classical controllers. A DC motor speed control experiment has been used to compare both methods of control where students can practice on both controllers and draw conclusions related to various system parameters and working conditions.

A fuzzy logic controlled LEGO robot for undergraduate teaching is presented by Azlan et al. [10]. In this paper, the study is divided into two parts: in the first part, objects were picked and loaded to a fuzzy controlled line following robot which carried the objects to a goal. In the second part, a fuzzy controlled robot with the capability to navigate in a maze was developed. The system described in this paper is complex for a first course in fuzzy logic control teaching.

Elmas and Akcayol [11] describe a PC-based educational tool for a switched reluctance drive with fuzzy logic, developed for undergraduate and graduate level students. This application is complex and very specific and cannot be considered to be a general-purpose fuzzy logic control teaching tool.

The aim of the study presented in this paper has been to use a low-cost robot with an on-board embedded microcontroller and then develop fuzzy control algorithms to control the movements of the robot to follow a track made of a black line drawn on a white surface. The main aim has been to develop a specially low-cost and simple to use fuzzy logic control teaching kit that can be used in the early stages of undergraduate engineering laboratory sessions.

FUZZY MOBILE ROBOT CONTROL

The introduction of fuzzy logic control concepts to undergraduates is challenging and the utilisation of an experimental component such as a line following robot helps to understand the practical implications of fuzzy logic control. The use of fuzzy logic in mobile robot control is not a new concept. Many researchers have reported in the past successful implementation of the fuzzy control algorithms for the control of mobile robots. A quick literature search reveals many papers on this topic and the contents of some of these papers are given very briefly in this section.

Saffiotti [12] describes the use of fuzzy logic in the control of autonomous mobile robots. Saffiotti analyses the mobile robot control as behavioural with three types of control: ''classical'' control regime where the robot is programmed to track a path by knowing its position with respect to the path, and with no external sensors. ''Sensor-based'' control regime where the aim is still to move the robot along a path, but this time by using external sensors. Typical examples include moving along a wall or contour, reaching a light source or a beacon, and avoiding obstacles. The third type is ''Complex'' control which takes into account possible obstacles blocking the path of the robot.

Olsen [13] describes the fuzzy control of an autonomous robot using the Motorola MC68HC12 microcontroller. This microcontroller has four built-in fuzzy logic instructions, allowing the development of low-level applications that can utilise the unique features of fuzzy logic. It is shown in the paper that the development of a fuzzy-logic-based robotic application is considerably simplified by using these specific instructions.

Hurley et al. [14] describe the use of fuzzy logic for controlling an educational mobile robot that can avoid obstacles. The robot called Rug-Warrior Pro was developed by the Massachusetts Institute of Technology for use in robotics courses, and was based on the Motorola MC68HC811E2 microcontroller with extended memory and real-time operating system. The objective was to teach students how to develop fuzzy-control-based robot control and obstacle avoidance algorithms in an unmapped and changing environment.

Surmann et al. [15] describe an autonomous robot that copes with uncertain, or approximate information and has to identify sudden perceptual situations to manoeuvre in real time. The paper describes a fuzzy-rule-based system approach controlling the movement of an autonomous mobile robot. Guiding and controlling of the robot are achieved by combining local actions and global strategies within the fuzzy controller. Different behaviours and perceptions are detected with the help of fuzzy rules and stored in fuzzy state variables. These state variables activate different fuzzy rule sets, which in turn change the behaviour of the fuzzy controller.

Another fuzzy-control-based mobile robot with obstacle avoidance algorithm is given by Bento et al. [16]. In this application the distance, angle, linear speed, and angular speed are controlled to move a robot in a planned path while avoiding obstacles. It is reported that the fuzzy controller showed good performance in path tracking and simultaneously good obstacle avoidance for different trajectories and obstacles configurations.

A microprocessor-based fuzzy logic controlled line following robot is described by Reuss and Lee [17]. The robot is based on the RCX Lego Mindstorms which incorporates an on-board Hitachi H8 microprocessor. Two light sensors are used under the robot to sense a white line drawn on a black surface and a fuzzy logic algorithm is used to move the robot to follow the line.

Ishikawa [18] describes a fuzzy-control-based autonomous and automated mobile robot. This paper presents a sensor-based navigation method using fuzzy control, of which the purpose is to construct an expert knowledge for efficient and better piloting of the robot. Here, functions are provided for tracing a planned path by sensing the robot and its difference angle from the planned path. Stationary and moving obstacles are also avoided by sensing the free area ahead of the robot. Fuzzy control is also used to select suitable rules (tracing a path/avoiding obstacles) according to a situation, which is derived from sensor information by using fuzzy logic control.

Another interesting paper on fuzzy logic and robot control is by Pawlikowski [19] where the development of a fuzzy logic speed and steering control system for an autonomous vehicle is described. Using an integrated vision system, the vehicle senses position relative to the angle of a line drawn on the ground, and processes that information through a fuzzy logic algorithm. The algorithm selects drive speeds for two independent motors, thereby providing the ability to go forward, or turn left or right while following a path.

FUZZY LOGIC CONTROL TEACHING

Fuzzy logic control is an elective course at the second year of the undergraduate Computer Engineering Department of the Near East University. The course attracts about 20 students and is offered in every semester, twice an academic year. The text used in the course is the *Fuzzy Logic With Engineering Applications* by Ross, which is a rather comprehensive text with real working examples. The main goals of the program has been to introduce the basic principles of fuzzy logic control to students and provide them with a simple fuzzy logic control experiment that they can carry out with little knowledge of software and hardware skills. The grading for the course consisted of a single final exam, given at the end of the course, an experimental laboratory work, and a number of written assignments in the form of homeworks given throughout the course.

THE MOBILE ROBOT USED

As the aim of this study has been to use a line following robot to teach the principles of fuzzy logic control, it is worthwhile to look at the details of the mobile robot used in the project. The robot used in this project is the Robo-PICA [20], an educational mobile robot which is sold in disassembled kit form (see Fig. 1). Robo-PICA is controlled by an on-board PIC16F887 type [21] microcontroller and the robot system has the following features:

- Mobile robot assembly with two geared DC motors.
- Track-belt assembly set.
- Experimenter board with
  - PIC16F887 microcontroller,
  - 20 MHz ceramic resonator,
  - LCD,
  - DC motor drive circuits,
  - Piezo buzzer,
  - LED,
  - ICD programming interface,
  - USART interface,
  - Servo motor interface,
  - I^2C bus interface, and
  - Switching power supply.
- Remote control transmitter and receiver modules.
- Distance measurement module.
- Infrared reflector modules.

Figure 1 Robo-PICA educational robot.

In addition to the basic robot assembly and the experimenter module, an ICD in-circuit programming module is supplied with the kit. This module is connected to the USB port of a PC and it is used to download program HEX code to the program memory of the on-board PIC16F887 microcontroller.

Robo-PICA can be programmed by either using the PIC assembly language, or using a high-level language, such as BASIC, PASCAL, or C. The kit is supplied with a CDROM which contains a 2K program size limited DEMO version of the mikroC compiler [22]. mikroC is a popular PIC microcontroller C compiler developed by mikroElektronika. The DEMO version should be sufficient to carry out many robot experiments, but if required, the user can purchase the full unlimited version of the compiler at a low cost from the manufacturers or their distributors.

One of the very important features of Robo-PICA is the ICD in-circuit programming interface and the PICkit2 PIC microcontroller chip programming software supplied on the CDROM. Users can, for example, develop their programs using the mikroC language and then download the resulting HEX

program code directly into the microcontroller program memory using the ICD in-circuit programming interface and the PICkit2 programmer software. This eliminates the need to have separate programmer device and programmer software.

A two-row character LCD is provided on-board which can be used for many purposes. For example, information such as robot movements can be displayed on the LCD second by second, or the LCD can be used for the calibration of the sensors.

An on-board LED and a piezo buzzer provide the programmer with visual and audio outputs that can be useful for either program debugging purposes or for indicating the status of various conditions linked to the movement of the robot.

An infrared transmitter–receiver module pair is provided with the robot with a physical range of up to 10 m. The receiver module is mounted on the robot chassis in a visible high position. The transmitter module is basically a small handheld unit with four push-button switches mounted on top of it. Robot movements can be controlled remotely by developing programs to use the infrared transmitter–receiver pair module.

An infrared distance sensor module is provided with the Robo-PICA robot having a useful range of 4–30 cm. This module is normally mounted directly infront of the robot and it can be used in collision detection and collision avoidance applications. For example, the robot can be programmed to detect obstacles in front of it and then avoid colliding with these obstacles possibly by moving back and changing its direction of movement. The distance sensor can also be used to create a map of the obstacles surrounding the robot.

A pair of infrared reflectors are mounted at the bottom, and at both corners of the robot. With the aid of these reflectors the robot can be programmed and controlled to follow, for example, a dark line drawn on a white background. An example of programming the Robo-PICA for a fuzzy logic control based line following application is given in a later section of the paper.

A servo motor can be connected to the Robo-PICA robot instead of the supplied DC motor for more precise movement and navigation applications. The experimenter board provides the required signals for driving a servo motor. Thus, students are able to experiment with both DC motor and stepping motor based control and navigation applications.

Additional sensors such as compass, accelerometer, electromagnetic sensor, temperature sensor, and GPS can easily be connected to the robot chassis. Connectors are provided on the experimenter board where analogue and digital I/O signals of the microcontroller are terminated, making it easy to connect external sensors to extend the sensing abilities and hence the intelligence of the robot.

The Robo-PICA robot is powered from four AA type batteries for stand-alone operation. A switching regulator is used to drive the +5 V supply required by the microcontroller, the motors, and the associated interface circuitry. A re-chargeable battery pack is recommended for educational and experimental applications where it may be required to experiment with the robot for long durations. The batteries can easily be re-charged using the pair of battery terminals provided on the experimenter board.

The movement of the Robo-PICA robot is achieved using a pair of geared DC motors, coupled to wheels, plastic track-belt assembly, sprockets, and axes. Using track-belt assembly enables the robot to move easily on almost any kind of surface.

FUZZY CONTROL OF THE LINE FOLLOWING ROBOT

Figure 2 shows the block diagram of the line following robot system developed. Left and right infrared reflectors detect the line under the robot and feed the received signals to the microcontroller

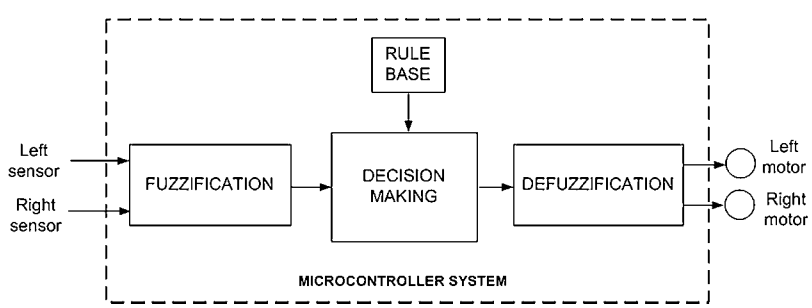

Figure 2 Block diagram of the system.

system. The microcontroller implements the fuzzy logic control algorithm and sends drive control signals to the left and right motors so that the robot is kept on the line.

A simplified circuit diagram of the system is given in Figure 3, showing the microcontroller, the infrared reflectors, and the motor drive circuitry. The left and right reflectors are connected to port pins RA0 and RA1 of the microcontroller, respectively. The motors are controlled using a L293D-type H-bridge motor driver IC which controls the direction as well as the speed of each motor.

Fuzzy control logic is implemented in three phases as shown in Figure 4:

- Fuzzification.
- Inference.
- Defuzzification.

Fuzzification

Fuzzification is the process of mapping crisp inputs to fuzzy membership functions. In fuzzy logic, it is important to distinguish not only which membership functions a variable belongs to, but also the relative degree to which it is a member. Figure 5 shows the input crisp variables to the robot sensors. As shown in the figure, fuzzy membership functions span a range of values and can overlap. Three sets of membership values are defined for the robot sensor inputs depending on the type of surface below the sensors: BLACK, GREY, and WHITE. The degree of membership is found by finding the intersection point of a distinct input value on the horizontal axis with the line defining one or more fuzzy membership functions. The sensor reading is analogue and is converted into digital form using the on-board A/D converter of the PIC microcontroller. A test performed on the sensors showed that as the shading of the surface under a sensor is changed, the sensor reading changes between 200 for BLACK surface to 900 for WHITE surface.

Inference Rule Definition

After defining the membership functions, we can generate the fuzzy rule definitions to relate the output actions of the controller to the observed sensor inputs. The rule definition is usually in the form of IF−THEN statements, but the rules can also be shown in the table format.

The following rules can be developed for the line following robot:

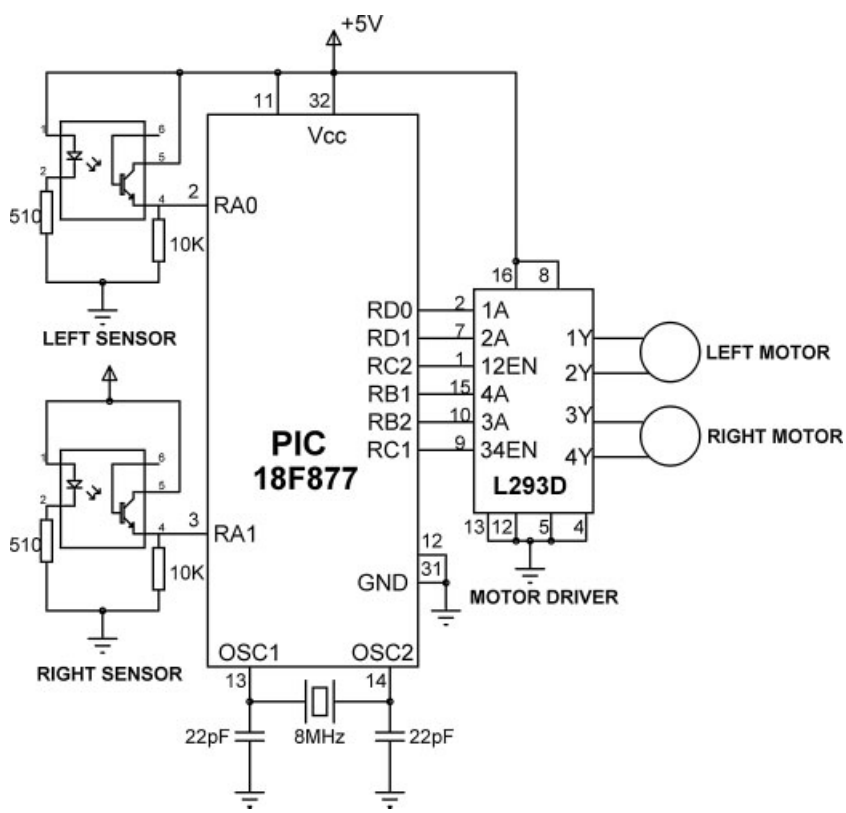

Figure 3 Circuit diagram of the system.

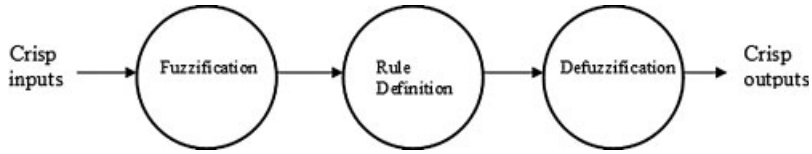

Figure 4 Fuzzy logic phases.

- IF (Left Sensor is WHITE) AND (Right Sensor is WHITE) Then Move Hard Forward
- IF (Left Sensor is BLACK) AND (Right Sensor is WHITE) THEN Move Hard Left
- IF (Left Sensor is GREY) AND (Right Sensor is WHITE) THEN Move Soft Left
- IF (Right Sensor is BLACK) AND (Right Sensor is WHITE) THEN Move Hard Right
- IF (Right Sensor is GREY) AND (Left Sensor is WHITE) THEN Move Soft Right

The fuzzy AND is obtained as the *minimum* of the two membership values. Thus, if μ_α and μ_β are two membership values for inputs α and β, then the fuzzy AND is described as

$$\mu_\alpha \quad \text{AND} \quad \mu_\beta = \min(\mu_\alpha, \mu_\beta) \tag{1}$$

Similarly, the fuzzy OR is obtained as the *maximum* of the two membership values. Thus, if μ_α and μ_β are two membership values for inputs α and β, then the fuzzy OR is described as

$$\mu_\alpha \quad \text{OR} \quad \mu_\beta = \max(\mu_\alpha, \mu_\beta) \tag{2}$$

Because the rules are based on linguistic terms instead of mathematical equations, any relationship can be defined by a fuzzy logic controller. This also means that even nonlinear systems can be described and controlled with a fuzzy logic controller.

Figure 6 shows the fuzzy logic rule definitions for the line following robot in the form of a table.The motor movements are controlled by the speed and the direction of each motor.

Forward. The robot is moved FORWARD when the right motor is turned clockwise and at the same time the left motor is turned anti-clockwise at fast speed.

Soft Left. The robot is turned SOFT LEFT when both the right motor and left motor are turned clockwise at low speed.

Hard Left. The robot is turned HARD LEFT when both the right motor and left motor are turned clockwise at high speed.

Soft Right. The robot is turned SOFT RIGHT when both the right motor and left motor are turned anti-clockwise at low speed.

Hard Right. The robot is turned HARD RIGHT when both the right motor and left motor are turned anti-clockwise at high speed.

Defuzzification

The last stage of a fuzzy controller is the defuzzification where a crisp output is generated based on the inputs and the rule base. In the case of the line following robot, the output is the control of the two robot motors. There are several methods available to obtain a crisp output from a fuzzy system. The most commonly used methods are

- Maximum defuzzification method.
- Centroid calculation defuzzification method.

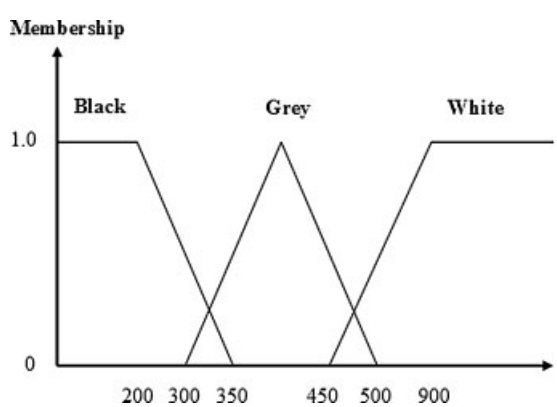

Figure 5 Sensor input membership functions.

	White	Grey	Black
White	F	SL	HL
Grey	SR	X	X
Black	HR	X	F

F: Forward
SL: Soft Left
HL: Hard Left
SR: Soft Right
HR: Hard Right

Figure 6 Fuzzy logic rules for the line following robot.

The maximum defuzzification method simply selects the rule which satisfies the maximum relative membership when more than one rule is active. The output is then set to the value specified by the selected rule. In this application, the maximum defuzzification rule is used for simplicity to set the speed of the robot's motors. Expressed mathematically, if the membership functions are μ_i, then the output is selected for the rule satisfying:

$$\max(\mu_i) \quad \text{for} i = 1 \text{to} n \tag{3}$$

where n is the number of rules.

Another method for calculating the output value is the centroid method. Here, a weighted average of all the active rules determines an output by summing all of the output variables over their relative membership values. The centroid method is one of the commonly used defuzzification methods. Expressed mathematically, if the membership functions are μ_i and the outputs are μ_i, then the crisp centroid output u is defined as

$$u = \frac{\sum_{i=1}^{n} u_i \mu_i}{\sum_{i=1}^{n} \mu_i} \tag{4}$$

THE SOFTWARE

The software is based on the mikroC language, which is a cut-down microcontroller version of the standard C language. Using a high-level language makes it easy for students to develop programs and concentrate on the algorithm rather than the actual coding task. The algorithm of the fuzzy control is shown in Figure 7 as a Program Description Language (PDL).

The program initially reads the two analogue sensors and converts the data into digital format using the ADC_Read() function of the mikroC language:

```
Left_Sensor = ADC_Read(0);
Right_Sensor = ADC_Read(1);
```

```
BEGIN
    DO FOREVER
        Read left sensor and convert to digital
        Read right sensor and convert to digital
        Calculate membership function for left sensor
        Calculate membership function for right sensor
        Apply the fuzzy logic rules
        Calculate the crisp outputs
        Control the motors using the crisp outputs
    ENDDO
END
```

Figure 7 Implementation of the fuzzy control algorithm.

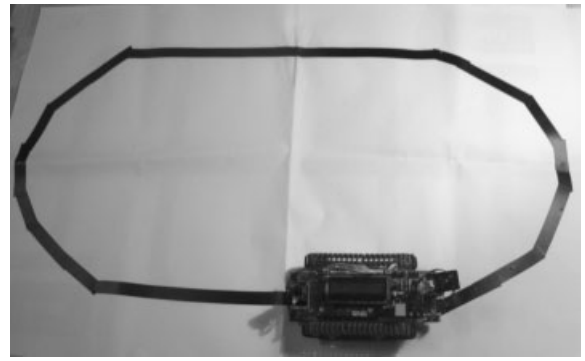

Figure 8 Fuzzy controlled line following robot.

where the *Left_Sensor* and *Right_Sensor* are connected to channel 0 and channel 1 of the A/D converter, respectively. The membership functions are then evaluated as shown graphically in Figure 8:For the left sensor:

```
Left_Black = BLACK(Left_Sensor);
Left_Grey = GREY(Left_Sensor);
Left_White = WHITE(Left_Sensor);
```

And for the right sensor:

```
Right_Black = BLACK(Right_Sensor);
Right_Grey = GREY(Right_Sensor);
Right_White = WHITE(Right_Sensor);
```

The software then implements the fuzzy rule base by calculating the *Min* function for each pair of rules joined with the *AND* operator. The minimum for each pair is stored in an array called MU:

```
MU[1] = MINF(left_white, right_white);
MU[2] = MINF(left_dark, right_white);
MU[3] = MINF(left_grey, right_white);
MU[4] = MINF(right_dark, left_white);
MU[5] = MINF(right_grey, left_white);
```

The last part of the program calculates the crisp output by finding the maximum, where function MAXF returns an integer number corresponding to the maximum of its parameters:

```
Crisp_Output = MAXF(MU[1],MU[2],MU[3],
MU[4],MU[5]);
```

The speed and direction of each motor are then controlled using functions *Forward*, *Hard_Left*, *Soft_Left*, and *Soft_Right*.

```
switch(Crisp_Output)
{
  case 1:
    Forward( );
    break;
  case 2:
    Hard_Left( );
    break;
  case 3:
    Soft_Left( );
    break;
  case 4:
    Hard_Right( );
    break;
  case 5:
    Soft_Right( );
    break;
}
```

The above operations are repeated forever in a *while* loop.

CONCLUSIONS

Fuzzy logic controllers have become popular in recent decades with successful implementation in many diverse fields and consequently many technical colleges and universities are now offering fuzzy logic courses. By using fuzzy logic control, one can simplify the design of complex and nonlinear control systems without having a model of the system. This paper has described the development of a low-cost educational fuzzy logic controller system for a line following robot. The system enables students to implement various control algorithms by simply programming in a high-level language. Results show that students could easily develop fuzzy logic control algorithms to move the robot to follow a line. Robot movements were smooth both in the forward track and at the corners. Figure 8 shows a picture of the robot following a black line drawn using black tape.

The fuzzy logic controller teaching system described in this paper can be enhanced by the addition of the following features:

- More sensors can be added to the robot and more complex fuzzy logic controller software can be developed to control the robot movements.
- Students can design ''classical'' control algorithms to control the robot and then compare these algorithms with the fuzzy logic controller algorithms.
- Obstacles can be introduced into the robot path and fuzzy logic controller algorithms can be developed to detect and avoid these obstacles.

REFERENCES

[1] L. A. Zadeh, Fuzzy sets. Information and control, Vol. 8, Academic, New York, 1965, pp 338–353.

[2] MATLAB fuzzy logic toolbox, www.mathworks.com/products/fuzzylogic [accessed on October 19, 2008].

[3] Introducing Aristotle, Fuzzy Logic Laboratorium Linz-Hagenberg. www.f11.uni-linz.ac.at/pdw/aristotle/aristotle.html [accessed on October 3, 2008].

[4] Togai Infra Logic, Inc., Fuzzy C Development System User's Manual, Togai Infra Logic, Inc., Irvine, CA, 1992.

[5] Bell Helicopters Textron, Inc., Fuzzy Logic Development Kit User' Manual, Bell Helicopters Textron, Inc., Fort Worh, TX, 1992.

[6] G.U.N.T. Geratebau, GmbH, RT 121 Fuzzy Control: Ball-on-Beam User's Guide, G.U.N.T. Geratebau, GmbH, Fahrenberg, Barsbuttel, 2006.

[7] Harz University of Applied Studies and Research, www.hs-harz.de [accessed on October 20, 2008].

[8] CE124 Fuzzy Logic Trainer User's Guide, www.mga-chem.co.jp [accessed on October 20, 2008].

[9] M. A. Akcayol, C. Elmas, O. A. Erdem, and M. Kurt, An educational tool for fuzzy logic controller and classical controllers, Comput Appl Eng Educ 12 (2004), 126–135.

[10] N. Z. Azlan, F. Zainudin, H. M. Yusuf, S. F. Toha, S. Z. S. Yusoff, and N. H. Osman, Fuzzy logic controlled miniature LEGO robot for undergraduate training system, 2nd IEEE Conference on Industrial Electronics and Applications, May 23–25, 2007.

[11] C. Elmas and M. A. Akcayol, PC based educational tool for a switched reluctance drive with fuzzy logic, Int J Elect Eng Educ 40 (2003), 208–219.

[12] A. Saffiotti, The uses of fuzzy logic in autonomous robot navigation, Soft Comput 1 (1997), 180–197.

[13] D. Olsen, Fuzzy logic control in autonomous robotics: Investigating the Motorola MC68HC12 on a line following robot, 2003, http://www.d.umm.edu/~olse0203/FuzzyControl/FuzzyPaper.html [accessed on October 20, 2008].

[14] M. D. Hurley, W. L. Xu, and G. Bright, Implementing fuzzy logic for machine intelligence: A case study, Int J Eng Educ 21 (2005), 178–186.

[15] H. Surmann, J. Huser, and L. Peters, A fuzzy system for indoor mobile robot navigation, Fourth IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 95, 1995, pp 83–88.

[16] L. C. Bento, G. Pires, and U. Nunes, A behaviour based fuzzy control architecture for path tracking and obstacle avoidance, Proceedings of the 5th Portuguese Conference on Automatic Control (Controlo 2002), Aveiro, 2002, pp 341–346.

[17] R. F. Reuss and T. Lee, Fuzzy control in a line following robot, Computer Science 791: Fuzzy Systems, 2001, http://www.cse.unr.edu/∼simon/fuzzy/fuzzylogic.htm [accessed on October 15, 2008].
[18] S. Ishikawa, A method of autonomous mobile robot navigation by using fuzzy control, Adv Robotics 9 (1995), 29−52.
[19] S. Pawlikowski, Development of a fuzzy logic speed and steering control system for an autonomous vehicle, MSc Research Project, Department of Mechanical Engineering, University of Cincinnati, 1999.
[20] Innovative Experiment Co. Ltd., Robotics experiments with pic microcontrollers, http://www.inexglobal.com [accessed on September 10, 2008].
[21] PIC16F887 Data Sheet, http://www.microchip.com [accessed on November 2, 2008].
[22] mikroC User Guide, http://www.mikroe.com [accessed on September 10, 2008].

BIOGRAPHIES

Dogan Ibrahim was born in Cyprus in 1954. He received the BSc degree in Electronic Engineering from the University of Salford in the UK, and MSc degree in Automatic Control Engineering from the Manchester University. Prof. Ibrahim received his PhD degree in Electrical Engineering from the City University in London in 1980. He is a professor and head of the Department of Computer Engineering at the Near East University in Nicosia, Cyprus. His areas of interest include microprocessors and microcontrollers, microcontroller-based design, and distant engineering education.

Tayseer Alshanableh was born in Palestine in 1963. He received the BSc degree in Civil Engineering from the Middle East Technical University, and MSc degree in Electrical & Electronic Engineering from the Near East University, Cyprus. Dr. Alshanableh received his PhD degree in Electrical Engineering also from the Near East University in 2007. His areas of interest include communications systems, signal processing, and artificial intelligence.