Electr Eng (2018) 100:695–709
https://doi.org/10.1007/s00202-017-0602-7

ORIGINAL PAPER

Designing driving and control circuits of four-phase variable reluctance stepper motor using fuzzy logic control

Auday Al-Mayyahi[1] · **Ramzy S. Ali**[1] · **Rabee' H. Thejel**[1]

Received: 23 December 2015 / Accepted: 3 July 2017 / Published online: 21 July 2017

Abstract Precise positioning and repeatability of movement for stepper motors require designing a robust control system. To achieve that, an analytical model of a four-phase variable reluctance stepper motor is presented. A proposed open-loop driving circuit is designed to control the motion of a variable reluctance stepper motor. The driving circuit has an ability to drive the motor into two-step angles, i.e. a full step (15°) and a half step (7.5°). The direction of movement can be either into clockwise or counterclockwise direction. The operation of the variable reluctance stepper motor in an open-loop control circuit has demonstrated disadvantages of an oscillation and a relatively high settling time. Therefore, a closed-loop control circuit has been introduced using fuzzy logic control to overcome the oscillation problem and to obtain on a precise positioning within a reasonable settling time. The fuzzy logic control is used to improve and enhance the behaviour of the step position response based on oscillatory response and hence to reduce the overshoot significantly. The comparisons between the open- and closed-loop circuits are presented to demonstrate the disparity between both control circuits. The simulation results of the open-loop and the closed-loop circuits show that the time responses have been improved using different loads conditions. The simulation experiments are conducted and investigated using MATLAB–SIMULINK software package.

Keywords Four-phase variable reluctance stepper motor · Driving and control circuits · Proportional derivative (PD) · Fuzzy logic control (FLC) · Step response · Settling time and overshoot

✉ Auday Al-Mayyahi
auday.almayyahi@hotmail.com

[1] Department of Electrical Engineering, College of Engineering, University of Basrah, Basra, Iraq

List of symbols

V_a	Voltage applied to phase A
i_a	Current in phase A
r_a	Resistant of phase A
λ_a	Flux linkage of phase A
l_{aa}	Self-inductance of phase A
l_{ab}	Mutual inductance between phases A and B
l_{ac}	Mutual inductance between phases A and C
l_{ad}	Mutual inductance between phases A and D

1 Introduction

Stepper motors have various motion control applications in many systems such as industrial machines and electronics. Nowadays, they are used in a wide variety of applications that require a position driving system including computer peripherals and robotics. There are three main types of stepping motors, i.e. variable reluctance (VR), permanent magnet (PM), and hybrid stepper motors. Generally, stepper motors can be defined as an electro-mechanical device that converts a pattern of inputs and the rate of change of those inputs into a precise rotational motion [1,2]. In the state of the art, many pieces of research have been reported to tackle the positioning control of PM stepper motor [3–6] and hybrid stepper motors [7,8]. The variable reluctance stepper motors have been around for a long time. Nonetheless, they have not been widely applied into real-world situations that demand a quite precise positioning. One of the major concerns regarding the influence for the response behaviour of the rotor position is the oscillatory response and the high overshoot and undershoot. It can be described as the variation of the rotor response to stationary poles. In fact, the construction of the VR stepper motor is much simpler than other types; hence, the complexity of control circuits can be minimised and cost can be reduced. As a result, a development of the VR stepper motor is needed to cope with the current limitations that imply with the operation of such type of motors.

Practically, a VR stepper motor has a number of functional differences when compared with a PM stepper motor. Because the rotor is not magnetised, it produces less torque rather than a similar sized PM stepper motor. In addition, it has no detent torque when there is no electrical power supply, which can be an advantage or disadvantage depending on required applications. Finally, because of the small step size and reduced detent torque, the VR stepper motor has more of a tendency to overshoot and the destination step might not be reached. The step position response in VR stepper motor demonstrates a poor damping characteristic, which in turn leads to generate a significant overshoot and extended settling time. In addition, resonance phenomenon may occur in such motors. These characteristics disturb the normal operation of the stepper motor under certain conditions in operation. This is a significant disadvantage when it occurs. In particular, if the VR motor is operated in an open-loop circuit. Moreover, in some applications related to a high-speed repetitive motion, it is essential to reduce the overshoot and settling time in order to guarantee that rotor attends a required step. Hence, this would positively increase the speed of the response and enhance the behaviour of the VR stepper motor. Furthermore, it is necessary to diminish significantly the oscillatory response and maintain zero tracking.

Previous studies demonstrated an application of robust nonlinear controller of the VR stepper motor. The control design had shown to be applicable to other motor models wherein the flux linkage was modelled as separable products of functions of the rotor position and winding currents [9]. Recently, many control circuits have been implemented to drive a stepper motor. For example, in reference [10], a driving system was designed for a stepper motor using an AT89C51 microcontroller. This driving system is to drive the stepper motor in an efficient manner so that the user has the provision to select the forward or reverse motions, full or half step, and speed or fixed number of steps of movement. In [11], a driving circuit was implemented using a switch mode H-bridge method to derive a stepper motor. However, the implemented circuit did not consider constructing a mathematical model for a specific type of stepper motors. Hence, it might not guarantee an accurate step response. In [12], an adaptive artificial neural network controller applied on a PM stepper motor for regulating its speed. The dynamic response of the PM stepper motor with the proposed controller was studied during the starting process under the full-load torque and under load disturbance. The stepper motor in an open-loop driving circuit presented in a reasonable performance [13].

In the state of the art, a few pieces of research have been introduced based on a fuzzy logic control to achieve an accurate position tracking of stepper motors. For instance, in reference [14], a simple transfer function based on second order was used to achieve the position tracking. Several limitations are observed in that work such as lacking of sequence stepping-based system behaviour. The motor was represented as single-loop electrical circuit. Hence, this does not reflect an accurate representation of the system modelling. In principle, that would be likewise to a direct current (DC) motor. Also, a driving circuit was given to drive the stepper motor based on specific sequences. Similarly, in reference [15], although the application of the fuzzy logic controller was demonstrated. However, some constraints are noticed such as a single step was only conducted without achieving further sequences to mimic the practical performance of stepper motors. In reference [16], another simplification of a stepper motor was conducted. The mathematical model was accomplished using so-called direct-quadrature (DQ) model to analyse a system. A linear representation of the model did not take into consideration the influence of nonlinearity for variables of stepper motor such as self and mutual inductance. Hence, it suffers from lacking of accuracy in a representation of the model.

In this paper, we have introduced a new approach based on a fuzzy logic controller. This controller will be integrated into a proposed open-loop driving circuit and a presented model of the VR stepper motor. Such controller is utilised due to its wide range of applications to control nonlinear systems. By using the FLC, the overshoot will be diminished dramatically and the settling time will be reduced. The contribution can be understood as follows: the mathematical model of the four-phase VR stepper motor is derived and analysed to provide a comprehensive and accurate modelling of the multiple-phase stepper motor. In addition, a thorough and diligent open-loop driving circuit is designed to drive the motion of stepping sequence. Finally, a closed-loop control circuit is proposed based on fuzzy logic controller to overcome the overshoot problem and reduce the settling time of a response. The position responses of the rotor using the open- and closed-loop circuits are conducted based on many operational scenarios. The simulation results are established under different load conditions starting from the no-load to full-load situation using MATLAB software package.

The rest of the paper is structured as follows; Sect. 2 studies the mathematical model of the four-phase VR stepper motor. Section 3 illustrates the design of open-loop driving circuit. The designing of the closed-loop PD fuzzy controller is presented in Sect. 4. The simulation results are reported and explained in Sect. 5. Finally, the conclusion is given in Sect. 6.

2 Mathematical model of a VR stepper motor

The VR stepper motor is characterised by the fact that there is no permanent magnet either on the rotor or the stator. The rotor carries no windings either. It is of salient pole type and is made entirely of soft iron stampings. The stator carries the windings, and the number of poles on the stator is an even multiple of the number of phases for which the stator windings are wound [17]. The construction of a four-phase VR stepper motor made of eight teeth on the stator and six teeth on the rotor is shown in Fig. 1. In this section, the modelling and analysing of the mathematical model for the VR stepper motor are presented. The aim is to obtain a group of differential equations that represent the dynamic behaviour of the motor under different operating conditions. This model has four input voltages, which are provided from pulses in different sequences, and forms by utilising a driving circuit. The equations are derived after series of calculations for the VR stepper motor given below [18].

The calculation of the step angle 'R' can be conducted using the equation given below:

$$R = \frac{360^\circ}{n} \tag{1}$$

where 'n' represents the number of steps per revolution and can be calculated by the following equation:

$$n = \frac{1}{\left[\left(\frac{1}{N_\mathrm{r}}\right) - \left(\frac{1}{N_\mathrm{s}}\right)\right]} \tag{2}$$

where N_s, number of teeth on the stator, and N_r, number of teeth on the rotor.

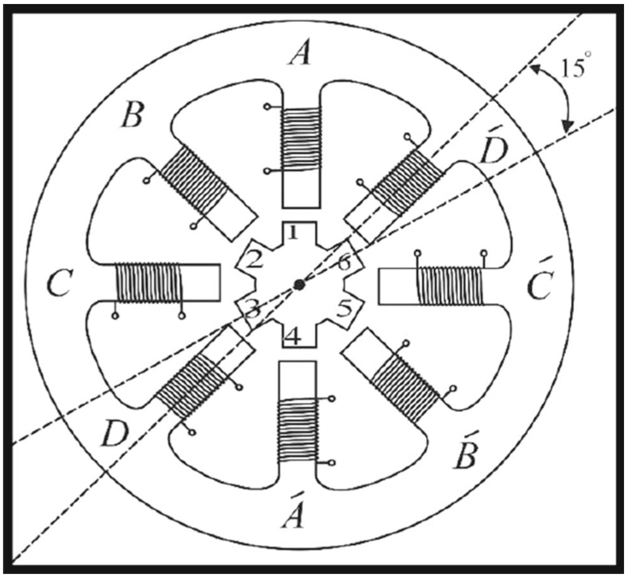


Fig. 1 A cross section of four-phase VR stepper motor

The developed model in this paper has eight teeth on the stator and six teeth on the rotor as shown above. Therefore, the step angle can be calculated using both Eqs. (1) and (2) that will yield '15°' step angle. The voltage of each phase of the four-phase VR stepper motor can be written as:

$$\left.\begin{aligned} V_a &= i_a r_a + \tfrac{\mathrm{d}}{\mathrm{d}t}(\lambda_a) \\ V_b &= i_b r_b + \tfrac{\mathrm{d}}{\mathrm{d}t}(\lambda_b) \\ V_c &= i_c r_c + \tfrac{\mathrm{d}}{\mathrm{d}t}(\lambda_c) \\ V_d &= i_d r_d + \tfrac{\mathrm{d}}{\mathrm{d}t}(\lambda_d) \end{aligned}\right\} \tag{3}$$

The flux linkage of each phase depends on the self-inductance of its phase and mutual inductance with other phases as given below.

$$\left.\begin{aligned} \lambda_a &= l_{aa} i_a + l_{ab} i_b + l_{ac} i_c + l_{ad} i_d \\ \lambda_b &= l_{ba} i_a + l_{bb} i_b + l_{bc} i_c + l_{bd} i_d \\ \lambda_c &= l_{ca} i_a + l_{cb} i_b + l_{cc} i_c + l_{cd} i_d \\ \lambda_d &= l_{da} i_a + l_{db} i_b + l_{dc} i_c + l_{dd} i_d \end{aligned}\right\} \tag{4}$$

Following equations can be derived by substituting the flux linkages from Eq. (4) into (3):

$$\begin{aligned} \frac{\mathrm{d}i_a}{\mathrm{d}t} = \frac{1}{l_{aa}}\Bigg[& V_a - i_a r_a - i_a \frac{\mathrm{d}l_{aa}}{\mathrm{d}\theta}\omega - l_{ab}\frac{\mathrm{d}i_b}{\mathrm{d}t} - i_b \frac{\mathrm{d}l_{ab}}{\mathrm{d}\theta}\omega \\ & - l_{ac}\frac{\mathrm{d}i_c}{\mathrm{d}t} - i_c \frac{\mathrm{d}l_{ac}}{\mathrm{d}\theta}\omega - l_{ad}\frac{\mathrm{d}i_d}{\mathrm{d}t} - i_d \frac{\mathrm{d}l_{ad}}{\mathrm{d}\theta}\omega \Bigg] \end{aligned} \tag{5}$$

$$\begin{aligned} \frac{\mathrm{d}i_b}{\mathrm{d}t} = \frac{1}{l_{bb}}\Bigg[& V_b - i_b r_b - i_a \frac{\mathrm{d}l_{ba}}{\mathrm{d}\theta}\omega - l_{ba}\frac{\mathrm{d}i_a}{\mathrm{d}t} \\ & - i_b \frac{\mathrm{d}l_{bb}}{\mathrm{d}\theta}\omega - l_{bc}\frac{\mathrm{d}i_c}{\mathrm{d}t} - i_c \frac{\mathrm{d}l_{bc}}{\mathrm{d}\theta}\omega - l_{bd}\frac{\mathrm{d}i_d}{\mathrm{d}t} - i_d \frac{\mathrm{d}l_{bd}}{\mathrm{d}\theta}\omega \Bigg] \end{aligned} \tag{6}$$

$$\begin{aligned} \frac{\mathrm{d}i_c}{\mathrm{d}t} = \frac{1}{l_{cc}}\Bigg[& V_c - i_c r_c - i_a \frac{\mathrm{d}l_{ca}}{\mathrm{d}\theta}\omega - l_{ca}\frac{\mathrm{d}i_a}{\mathrm{d}t} \\ & - i_b \frac{\mathrm{d}l_{cb}}{\mathrm{d}\theta}\omega - l_{cb}\frac{\mathrm{d}i_b}{\mathrm{d}t} - i_c \frac{\mathrm{d}l_{cc}}{\mathrm{d}\theta}\omega - l_{cd}\frac{\mathrm{d}i_d}{\mathrm{d}t} - i_d \frac{\mathrm{d}l_{cd}}{\mathrm{d}\theta}\omega \Bigg] \end{aligned} \tag{7}$$

$$\begin{aligned} \frac{\mathrm{d}i_d}{\mathrm{d}t} = \frac{1}{l_{dd}}\Bigg[& V_d - i_d r_d - i_a \frac{\mathrm{d}l_{da}}{\mathrm{d}\theta}\omega - l_{da}\frac{\mathrm{d}i_a}{\mathrm{d}t} \\ & - i_b \frac{\mathrm{d}l_{db}}{\mathrm{d}\theta}\omega - l_{db}\frac{\mathrm{d}i_b}{\mathrm{d}t} - i_c \frac{\mathrm{d}l_{dc}}{\mathrm{d}\theta}\omega - l_{dc}\frac{\mathrm{d}i_c}{\mathrm{d}t} - i_d \frac{\mathrm{d}l_{dd}}{\mathrm{d}\theta}\omega \Bigg] \end{aligned} \tag{8}$$

Similar statements can be written for the voltages, currents, and so on for the phases B, C, and D.

For the VR stepper motor type, the self and mutual inductances equations can be written as follows:

$$l_{bb} = [30 - 19.4|i_b|] + [13.5 - 23.1|i_b|]\cos\left(6\theta + \frac{\pi}{2}\right) \quad (9)$$

$$l_{cc} = [30 - 19.4|i_c|] + [13.5 - 23.1|i_d|]\cos\left(6\theta + \frac{3\pi}{2}\right) \quad (10)$$

$$l_{dd} = [30 - 19.4|i_d|] + [13.5 - 23.1|i_d|]\cos\left(6\theta\frac{3\pi}{4}\right) \quad (11)$$

$$l_{ab} = \left[-10.6 + 11.4 \times \frac{|i_a| + |i_b}{2}\right] + \left[7 - 11.3\frac{|i_a| + |i_b|}{2}\right]\cos\left(6\theta - \frac{3\pi}{4}\right) \quad (12)$$

$$l_{ac} = \left[-1.4 + 2\frac{|i_a| + |i_c|}{2}\right] + \left[3.7 - 3.6\frac{|i_a| + |i_c|}{2}\right]\cos\left(6\theta + \frac{\pi}{2}\right) \quad (13)$$

$$l_{ad} = \left[8.1 - 7.1\frac{|i_a| + |i_d|}{2}\right] + \left[4.5 - 7.6\frac{|i_a| + |i_d|}{2}\right]\cos\left(6\theta - \frac{\pi}{4}\right) \quad (14)$$

$$l_{bd} = \left[-1.4 + 2\frac{|i_b| + |i_d|}{2}\right] + \left[3.7 - 3.6\frac{|i_b| + |i_d|}{2}\right]\cos(6\theta + \pi) \quad (15)$$

$$l_{bc} = \left[-10.6 + 11.4\frac{|i_b| + |i_c|}{2}\right] + \left[7 - 11.3\frac{|i_b| + |i_c|}{2}\right]\cos\left(6\theta - \frac{\pi}{4}\right) \quad (16)$$

$$l_{cd} = \left[-10.6 + 11.4\frac{|i_c| + |i_d|}{2}\right] + \left[7 - 11.3\frac{|i| + |i_d|}{2}\right]\cos\left(6\theta + \frac{\pi}{4}\right) \quad (17)$$

The total developed torque is composed of self and mutual components. Using the same subscript convention as is used for the self and mutual inductances, the total developed torque is written as:

$$T_e = T_{aa} + T_{bb} + T_{cc} + T_{dd} + T_{ab} + T_{ac} + T_{ad} + T_{bc} + T_{bd} + T_{cd} \quad (18)$$

The self-torque can be written as:

$$\left.\begin{aligned} T_{aa} &= -17.8\,|i_a|\sin(6\theta) \\ T_{bb} &= -17.8\,|i_b|\sin\left(6\theta + \tfrac{\pi}{2}\right) \\ T_{cc} &= -17.8\,|i_c|\sin(6\theta + \pi) \\ T_{dd} &= -17.8\,|i_d|\sin\left(6\theta + \tfrac{3\pi}{2}\right) \end{aligned}\right\} \quad (19)$$

Table 1 Parameters of a single-phase variable reluctance stepper motor

Parameter	Definition	Value	Unit
V	Applied phase voltage	5	V
J	Moment of inertia	3.677×10^{-6}	N ms^2
r_a	Motor phase resistance	6	Ω
B	Viscous friction constant	3.5×10^{-3}	N ms
K_w	Load torque constant	7×10^{-5}	N ms

In order to determine the mutual torque, it is necessary to energise two phases together and measure the developed torque as a function of the rotor position for different values of current. These torques can be approximated as given below:

$$\left.\begin{aligned} T_{ab} &= -2.2\sqrt{|i_a i_b|}\sin\left(6\theta - \tfrac{3\pi}{4}\right) \\ T_{ac} &= -2.5\sqrt{|i_a i_c|}\sin(6\theta) \\ T_{ad} &= -2.2\sqrt{|i_a i_d|}\sin\left(6\theta + \tfrac{3\pi}{4}\right) \\ T_{bc} &= -2.2\sqrt{|i_b i_c|}\sin\left(6\theta - \tfrac{\pi}{4}\right) \\ T_{bd} &= -2.5\sqrt{|i_b i_d|}\sin\left(6\theta + \tfrac{\pi}{2}\right) \\ T_{cd} &= -2.2\sqrt{|i_c i_d|}\sin\left(6\theta + \tfrac{\pi}{4}\right) \end{aligned}\right\} \quad (20)$$

The motor developed torque can also be written as:

$$T_e = J\frac{\mathrm{d}\omega}{\mathrm{d}t} + B\omega + T_m \quad (21)$$

The shaft load can be given by the following equation:

$$T_m = K_w\,\omega \quad (22)$$

where K_w, Load torque constant (N ms).
The parameters of this motor are given in Table 1.

After deriving and implementing the mathematical model based on SIMULINK, it has been tested manually to verify its response to the given inputs. The next step in this paper is to design and construct a control circuit which is capable of automatically taking the VR stepper motor of reaching the required position. The pipeline of the designed control circuit is given in the next section.

3 Design an open-loop driving logic circuit

The driving of the four-phase VR stepper motor requires designing a driving logic circuit. The main task of this design is to enable the rotor of the four-phase VR stepper of rotating in both directions clockwise (CW) and counterclockwise (CCW) with two steps: full step (15°) and a half step (7.5°). The presented driving circuit consists of two inputs and four outputs. The input layer is set by a human user. The output layer is obtained based on the internal interfacing among different components so that the needed phases will be energised. The general construction of proposed logic driving

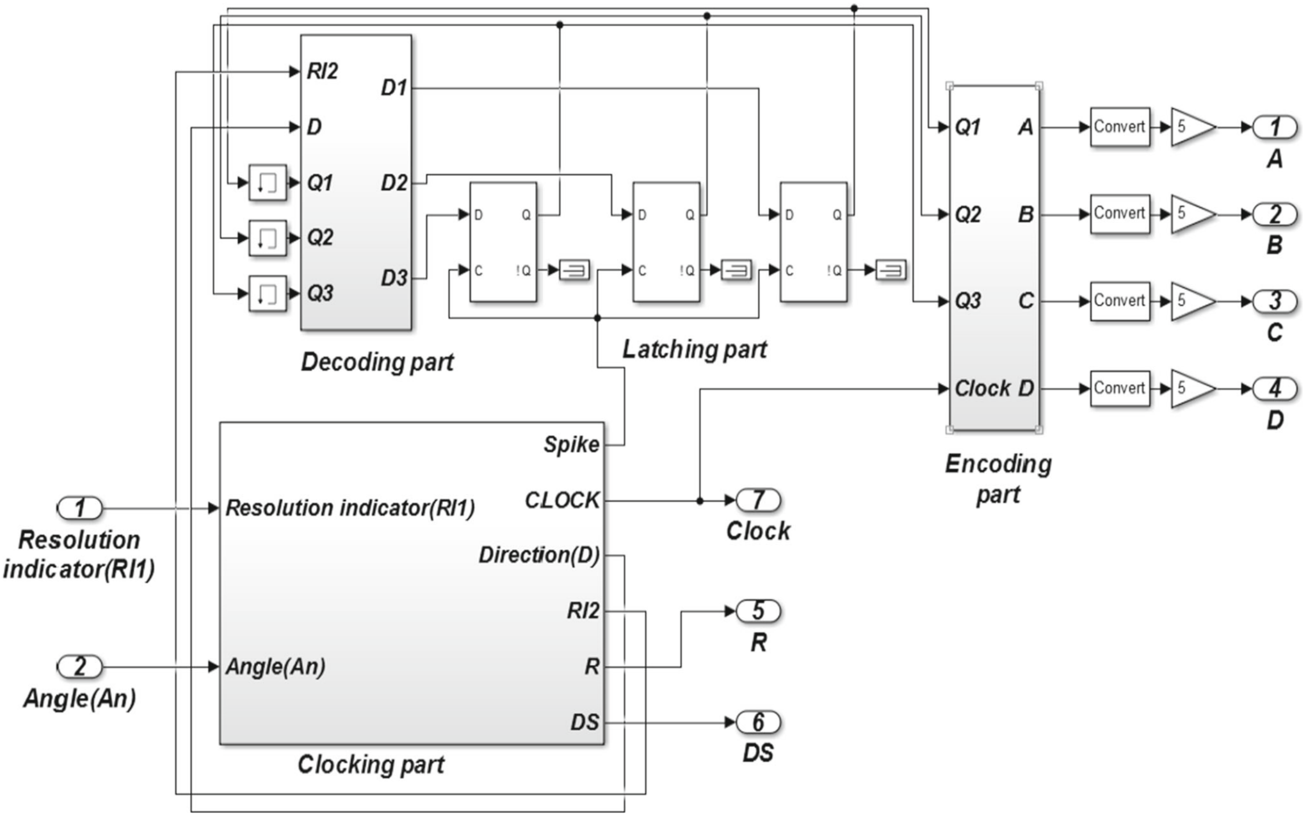


Fig. 2 Construction of the proposed of driving logic circuit

circuit is shown in Fig. 2. It demonstrates that the internal implementation of the driving circuit has three main parts, i.e. a decoding part, an encoding part, and a clocking part.

The angle (A_n) represents the required angular position, and it could take any real number. It is compared with the actual angular position taking into consideration both the magnitude and the direction. By this comparison, the required stepping direction and number of steps will be obtained. A full-step movement can be achieved when a single-phase excitation is used. The clockwise direction happens when the sequence of a single-phase excitation is ABCDA, whereas anticlockwise direction obtains when the sequence of switching ADCBA is applied. Similarly, the full-step movement can also be gained when double-phase excitation is applied. A clockwise stepping occurs when a sequence of AB–BC–CD–DA signals is supplied. Likewise, an anticlockwise stepping can be obtained when a sequence of AD–DC–CB–BA signals provided.

A half-step movement is achieved using both of single-phase and double-phase excitation systems. The sequence of clockwise direction is A–AB–B–BC–C–CD–D–DA–A, and the sequence of anticlockwise direction is A–AD–D–DC–C–CB–B–BA–A. Input (RI_1) is to specify that the step angle of the rotor is '7.5°' or '15°', if '$\mathrm{RI}_1 = 0$' the motor will rotate in 15° step angle only. In contrast, it will rotate in '7.5°' step angle only if '$\mathrm{RI}_1 = 1$'. Therefore, we introduced another option which is when '$\mathrm{RI}_1 = 2$', in this case, the rotor will rotate in any step angle based on the required angle in order to reach the final position as fast as possible. The decoding part is designed using the following truth table given in Table 2.

Using Karnaugh map, the outputs of the decoding parts can be analysed and obtained as in the equations given below. These equations are implemented logically to attain this part.

$$D_1 = \mathrm{RI}_2 \oplus Q_1 \tag{23}$$

$$D_2 = \overline{\mathrm{RI}_2 Q_2} + \left[\mathrm{RI}_2 \left(\overline{\overline{D_i} \oplus (Q_2 \oplus Q_1)}\right)\right] \tag{24}$$

$$\begin{aligned} D_3 = &\left(\overline{\mathrm{RI}_2} \oplus Q_2\right) \overline{D_i} \left(\overline{\mathrm{RI}_2 \oplus Q_1}\right) \\ &+ \left(\overline{\overline{\mathrm{RI}_2} \oplus Q_2}\right) D_i \left(\overline{Q_1}\right) + \left(Q_1 \oplus \overline{D_i}\right) \mathrm{RI}_2 Q_3 \\ &+ \left[(\overline{\mathrm{RI}_2} \oplus Q_2) \oplus D_i\right] \overline{\mathrm{RI}_2} Q_1. \end{aligned} \tag{25}$$

The encoding part is the final part of the driving circuit. By this part, the four phases of output voltages are generated. The implementation of this part is made by using the truth table given in Table 3. Hence, the output equations of this part will also be obtained logically using Karnaugh map and Boolean algebra.

$$A = \left[\overline{Q_3}\,\overline{Q_2} + Q_3 Q_2 Q_1\right] C_k \tag{26}$$

Table 2 Truth table of decoding part of driving logic circuit

State	Inputs of control		Present state of counter			Next state of counter			Outputs of decoding part			Description of each state
	RI_2	D	Q_3	Q_2	Q_1	Q_3+	Q_2+	Q_1+	D_3	D_2	D_1	
Single-phase excitation	0	0	0	0	0	0	1	0	0	1	0	A to B 15° CW
	0	0	0	1	0	1	0	0	1	0	0	B–C 15° CW
	0	0	1	0	0	1	1	0	1	1	0	C–D 15° CW
	0	0	1	1	0	0	0	0	0	0	0	D–A 15° CW
	0	1	0	0	0	1	1	0	1	1	0	A–D 15° CCW
	0	1	1	1	0	1	0	0	1	0	0	D–C 15° CCW
	0	1	1	0	0	0	1	0	0	1	0	C–B 15° CCW
	0	1	0	1	0	0	0	0	0	0	0	B–A 15° CCW
Single-phase-double-phase excitation	1	0	0	0	0	0	0	1	0	0	1	A–AB 7.5° CW
	1	0	0	0	1	0	1	0	0	1	0	AB–B 7.5° CW
	1	0	0	1	0	0	1	1	0	1	1	B–BC 7.5° CW
	1	0	0	1	1	1	0	0	1	0	0	BC–C 7.5° CW
	1	0	1	0	0	1	0	1	1	0	1	C–CD 7.5° CW
	1	0	1	0	1	1	1	0	1	1	0	CD–D 7.5° CW
	1	0	1	1	0	1	1	1	1	1	1	D–DA 7.5° CW
	1	0	1	1	1	0	0	0	0	0	0	DA–A 7.5° CW
	1	1	0	0	0	1	1	1	1	1	1	A–AD 7.5° CCW
	1	1	1	1	1	1	1	0	1	1	0	AD–D 7.5° CCW
	1	1	1	1	0	1	0	1	1	0	1	D–DC 7.5° CCW
	1	1	1	0	1	1	0	0	1	0	0	DC–C 7.5° CCW
	1	1	1	0	0	0	1	1	0	1	1	C–CB 7.5° CCW
	1	1	0	1	1	0	1	0	0	1	0	CB–B 7.5° CCW
	1	1	0	1	0	0	0	1	0	0	1	B–BA 7.5° CCW
	1	1	0	0	1	0	0	0	0	0	0	BA–A 7.5° CCW
Double-phase excitation	0	0	0	0	1	0	1	1	0	1	1	AB–BC 15° CW
	0	0	0	1	1	1	0	1	1	0	1	BC–CD 15° CW
	0	0	1	0	1	1	1	1	1	1	1	CD–DA 15° CW
	0	0	1	1	1	0	0	1	0	0	1	DA–AB 15° CW
	0	1	1	1	1	1	0	1	1	0	1	AD–DC 15° CCW
	0	1	1	0	1	0	1	1	0	1	1	DC–CB 15° CCW
	0	1	0	1	1	0	0	1	0	0	1	CB–BA 15° CCW
	0	1	0	0	1	1	1	1	1	1	1	BA–AD 15° CCW

Table 3 Truth table of encoding part of driving logic circuit

Case no.	The inputs of encoding part				The outputs of encoding part			
	Clock	Q_3	Q_2	Q_1	A	B	C	D
1	1	0	0	0	1	0	0	0
2	1	0	0	1	1	1	0	0
3	1	0	1	0	0	1	0	0
4	1	0	1	1	0	1	1	0
5	1	1	0	0	0	0	1	0
6	1	1	0	1	0	0	1	1
7	1	1	1	0	0	0	0	1
8	1	1	1	1	1	0	0	1
9	0	x	x	x	0	0	0	0
10	1	x	x	x	0	0	0	0
11	0	x	x	x	0	0	0	0

$$B = \left[\overline{Q_3}Q_2 + \overline{Q_3}\,\overline{Q_2}Q_1\right]C_k \tag{27}$$

$$C = \left[Q_3\overline{Q_2} + \overline{Q_3}Q_2Q_1\right]C_k \tag{28}$$

$$D = \left[Q_3Q_2 + Q_3\overline{Q_2}Q_1\right]C_k \tag{29}$$

The final part of the driving circuit is the clocking part. It is built based on a SIMULINK model as shown in Fig. 3. It consists of up-counting counter, s-function block, sample and hold circuits (S/H), and various logic circuits. The up-counting counter has two inputs, i.e. reset and clock. The counter can count from zero up to ‘126’ when the reset input is logic ‘0’ and goes to reset if it is logic ‘1’. The sample and hold (S/H) circuit’s outputs represent the temporary memory storage that will store the initial and the required angular positions. The s-function block so-called step control is a MATLAB software program. It was coded for calculating the maximum number of steps (M_S), the step angle (R), and the direction of stepping (D).

The input count (C) is used when ‘RI_1’ equals ‘2’ and ‘A_n’ equals ‘7.5’ or the multiple of ‘7.5’. This will be utilised to convert the step angle from ‘15°’ to ‘7.5°’ for the final step of rotation. The maximum step (MS) used to specify the maximum number of steps during the rotation of the rotor. The direction (D) is used to specify the direction of rotation in CW or CCW (if $D = 0$, the motor will rotate in CW and it will rotate in CCW when $D = 1$). The resolution indicator 2 (RI_2) on the output side of the step control function is employed to specify the step angle of the motor which is either ‘7.5°’ or ‘15°’. This can be specified when the resolution indicator 1 (RI_1) on the input side of the step control function is set to ‘2’. When the $RI_1 = 0$, the motor will rotate in ‘15°’ step angle, and it will rotate in ‘7.5°’ step angle if equals ‘1’ as aforementioned. However, choosing $RI_1 = 2$ will leave the driving circuit to decide and select which step angle is the best to be used in order to reduce the number of steps to reach a particular destination. Consequently, this will lead to reduce the required time to reach the target angle of the rotor’s position. The output (R)

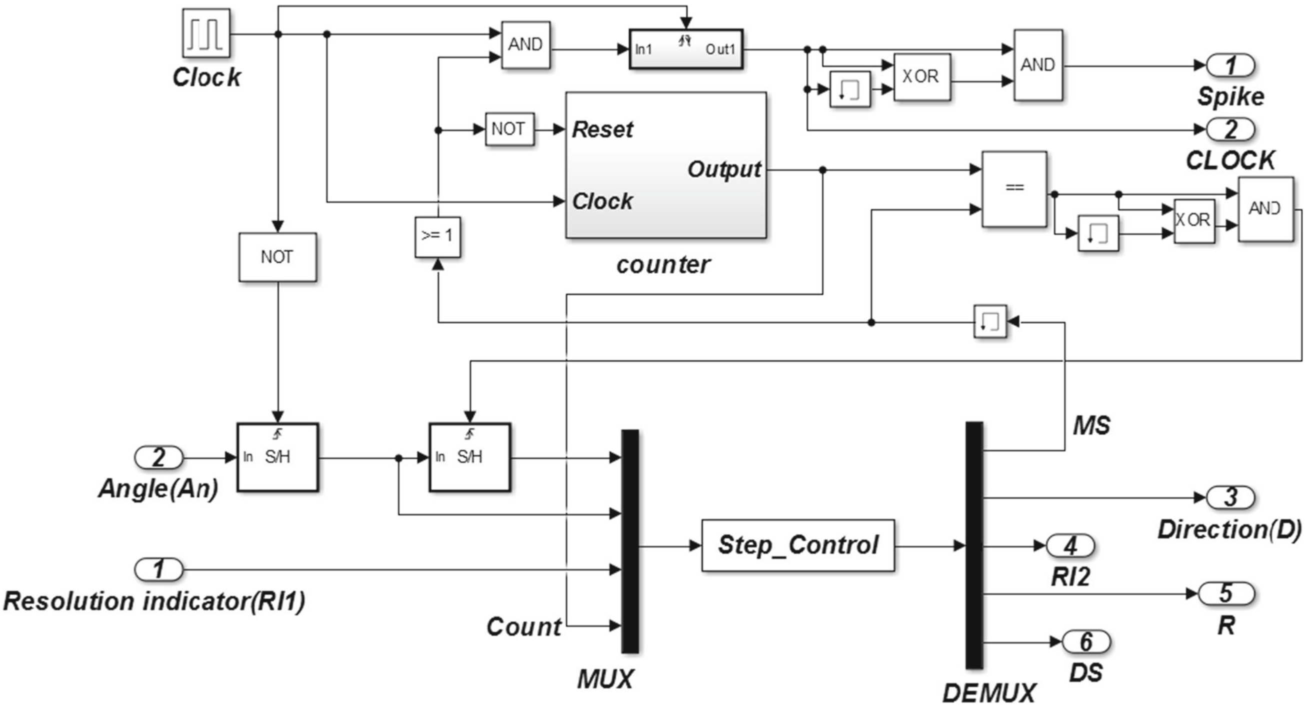
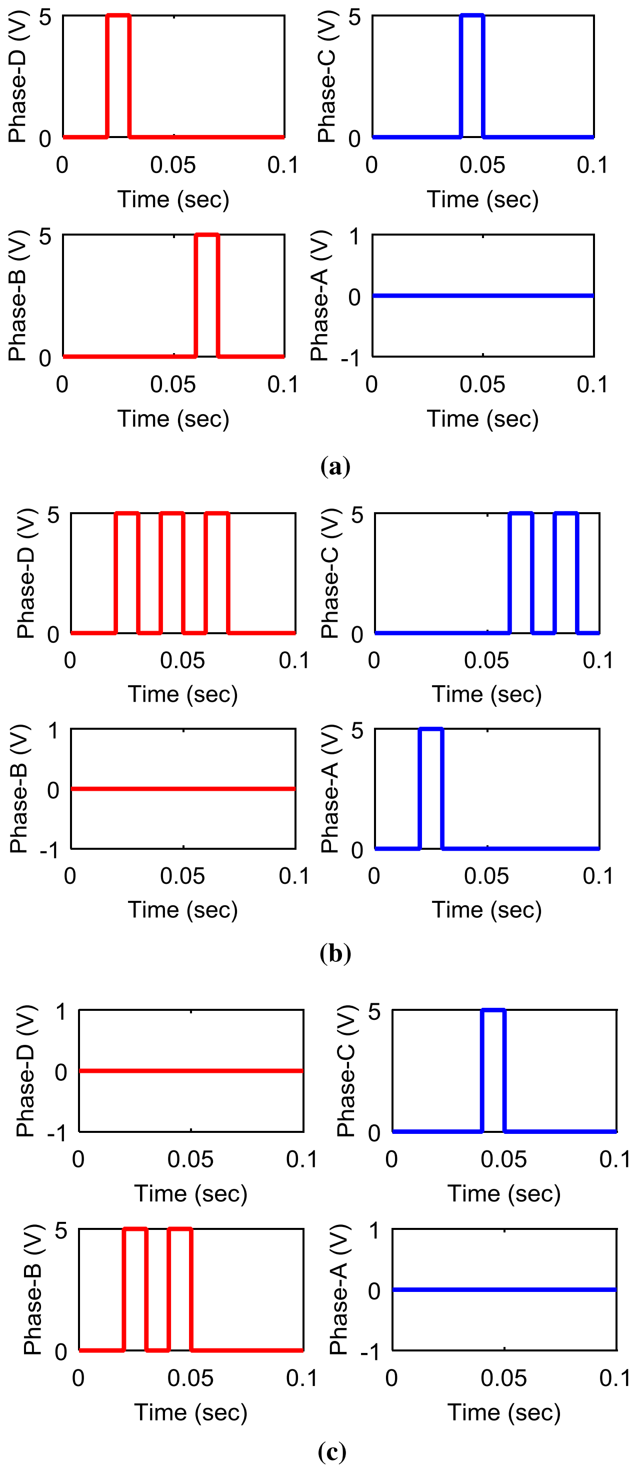


Fig. 3 Proposed SIMULINK model of the clocking part

provides the value of the step angle which is either ‘7.5°’ or ‘15°’. The direction sign (DS) equals ‘−1’ if the motor rotates in CW, and equals ‘1’ when the motor rotates in CCW.

For investigation and validation of a full operation for the driving circuit, we introduced four examples of supplied pulses as depicted in Fig. 4 to show the motor input voltages obtained from the driving circuit under different cases. For instance, in Fig. 4a, when phase D winding is energised, the rotor moves to the first ‘15°’. When phase D is de-energised and phase C is energised, the motor moves ‘15°’ in CW direction. Next, de-energising phase C and energising phase B cause the rotor to move to a further ‘15°’ in CW direction. Similarly, in Fig. 4b, when both phase A and phase D are energised at the same time, the rotor will move ‘7.5°’. Next, when phase A is de-energised and phase D is energised, the rotor will move ‘7.5°’ and so on until reaching the final position. The same rotation will be happening if and only if the same energising principle is applied. Another example is illustrated in Fig. 4c when different sequence of pluses are applied to inputs of the stepper motor. However, these sequences will lead to ‘15°’ movement and later on ‘7.5°’. This is due to the fact that RI_2 is set to ‘2’ which means both of step angles could be utilised to complete the required angle.



(a)

(b)

(c)

Fig. 4 Motor input voltages **a** $A = 45°$ and $\mathrm{RI}_1 = 0$, **b** $A = 30°$ and $\mathrm{RI}_1 = 1$, **c** $A = -22.5°$ and $\mathrm{RI}_1 = 2$

4 Design closed-loop proportional derivative (PD) fuzzy logic control

The essential elements in designing a fuzzy controller are defining input and output variables, choosing fuzzification and defuzzification methods, and determining the rule base of the controller. Different types of fuzzy controller can be implemented based on problem requirements such as proportional derivative (PD) and proportional derivative integral (PID) fuzzy controller [19]. In the literature survey, the PD fuzzy controller has been used in many real-world industrial applications [20,21]. In this paper, we utilise the PD type due to the fact that the derivative action helps to predict the future error, and the proportional derivative controller uses the derivative action to attenuate oscillations [22].

In PD-like fuzzy controller, the mapping is established between the control action $u(t)$ on one side, the error $e(t)$, and its change $\Delta e(t)$; on the other hand, in conventional case: $u(t) = f(e(t), \Delta e(t))$, but in this design the effect of load torque constant (K_w) is added in the structure of PD-like fuzzy controller. Therefore, the mapping now has an additional factor which is the load torque constant (K_w). In the new considered case, the output can be given as: $u(t) = f(e(t), \Delta e(t), K_w)$. The rotor position error is used as an input signal to this controller. The output control signal of the FLC is a switching signal that will be supplied to the motor's windings. The block diagram of this fuzzy controller is shown in Fig. 5. In this controller, the Sugeno fuzzy approach is used to obtain a single-tone output. The architecture of the closed-loop PD-like fuzzy controller for VR stepper motor is demonstrated in Fig. 6.

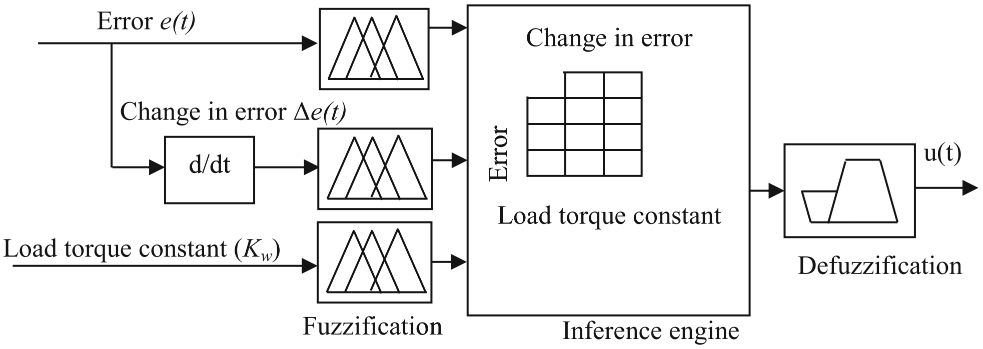


Fig. 5 Block diagram of the proposed fuzzy logic controller

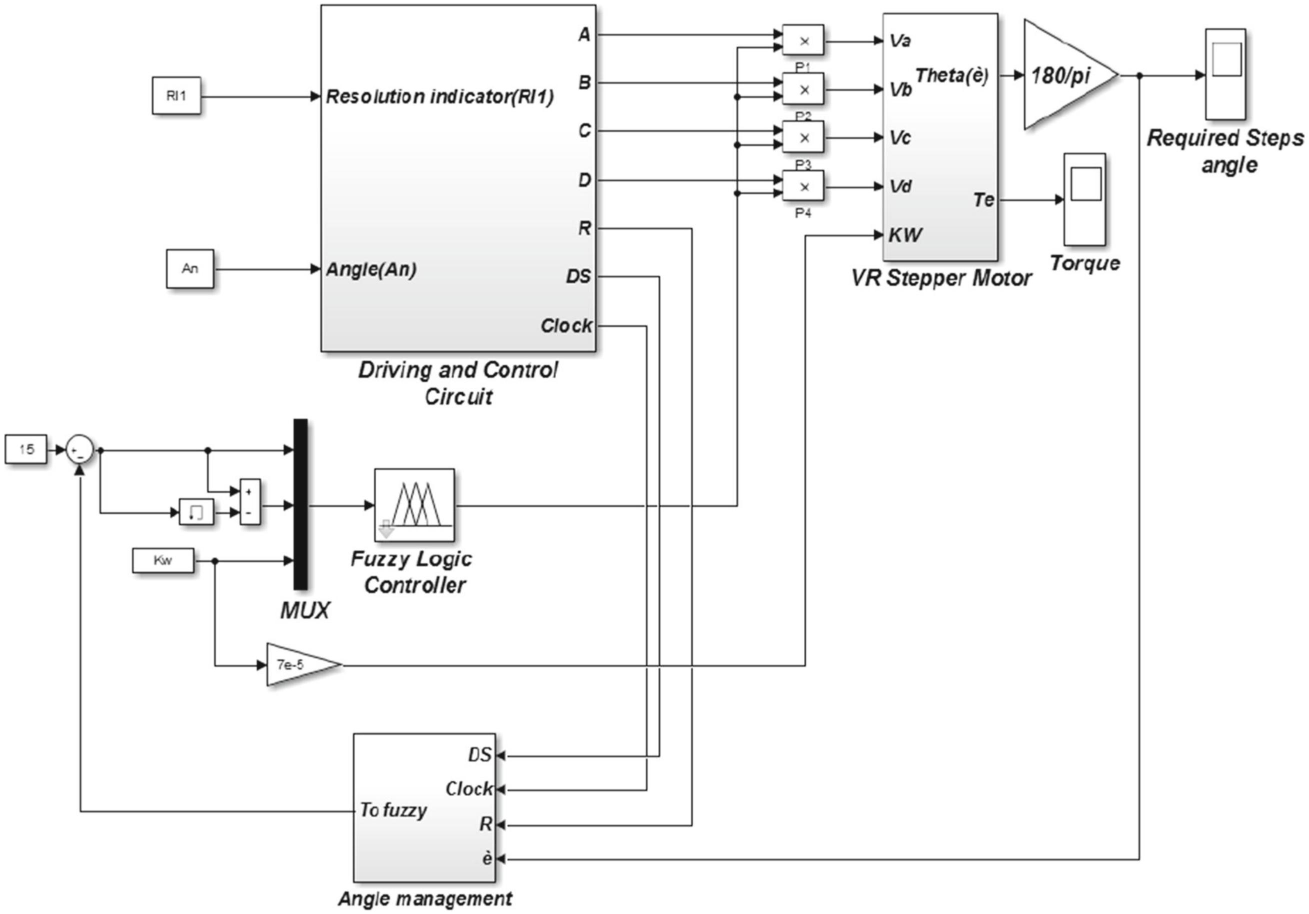


Fig. 6 SIMULINK model for closed-loop PD fuzzy controller

The block diagram of the angle management circuit is introduced in Fig. 7. It is noticeable that the angle management has four inputs, i.e. the direction signal (DS), step angle (R), clock, and rotor angular position (θ). This circuit is used to ensure that the range of the current angle which will be compared with the reference value (15°), is always between ‘0’ and ‘15°’, because the FLC is designed using this range.

In order to implement the fuzzy controller, several parameters need to be defined such as memberships and rules of the inference engine. In this controller, the number of the inputs is three so it means that different memberships need to be chosen for these inputs. The choice of these memberships is set based on the expected values of the input that the memberships are set for. For instance, we set the error memberships between ‘−15’ to ‘15’ such as Negative (N), Zero (Z), and Positive (P) as this is the maximum and minimum values of the expected error. Also, the positive side of the error has been partitioned into another six small memberships to

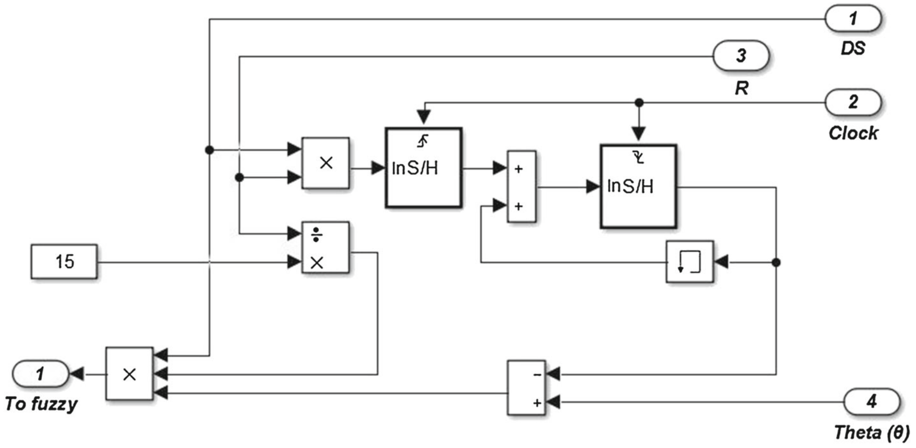


Fig. 7 Block diagram of the angle management circuit

obtain a more accurate range of the positive error. In addition, three memberships such Positive High (PH), Positive Medium (PM), and Positive Low (PL) are added to each of the six positive memberships to cover the three corresponding load conditions. The membership functions for the error, the change in error, the load torque constant, and the output are shown in Fig. 8.

The rules of inference engine are presented for the error and the load torque constant in Table 4 and for the error and change in error are given in Table 5. The surface views that show the relationships between the inputs and the output are depicted in Fig. 9a, b.

5 Simulation results

To test the validation of the proposed driving circuit and fuzzy control, many simulation experiments have been carried out. Various step angles have been applied to verify the target reaching of the rotor position. The simulation results were investigated into three different cases of the applied load. In order to obtain an improvement in the step response of the VR stepper motor, i.e. the overshoot and settling time in the position angular response, the simulation results of the closed-loop control circuit are presented. The three case studies are introduced and explained as follows:

5.1 Case study I

In this case study, no load was applied on the VR stepper motor. In the simplest case, we tested the rotor response with a ‘15°’ step angle to calculate the overshoot and oscillation response in such a case. In fact, the overshoot percentage was greatly high as expected although it takes a short time to diminish. However, this might cause a loss tracking problem in real-world scenarios that require precise position. Figure 10 shows the position angular response of VR stepper motor under no-load condition when ‘$A_n = 15°$’ and ‘$\mathrm{RI}_1 = 0$’. It clearly can be seen that the peak and steady state values are ‘25°’ and ‘15°’, respectively. Therefore, the overshoot percentage can be calculated using the following basic equation;

$$\text{Overshoot} = \frac{\text{Peak Value} - \text{Steady state value}}{\text{Steady state value}} \times 100\% \tag{30}$$

For this experiment, the calculated overshoot is ‘66.66%’ and settling time is ‘10 ms’. The blue line shows the rotor position response of the closed-loop control circuit under the same conditions. In this case, the peak and steady state values of rotor position response are ‘16°’ and ‘15°’, respectively. Also, the settling time was reduced to ‘8 ms’. Again, we can find out how much the overshoot has been improved based on the same Eq. (30) which will be ‘6.66%’.

The new obtain value demonstrates a significant improvement for the rotor step response using the proposed fuzzy control technique. It can be clearly seen that the overshoot percentage is improved greatly by ‘60 %’ and the settling time is improved by ‘2 ms’. The obtained results can confirm the rotor reaching to its target without probability of losing the required step angle.

In Fig. 11, the rotor position rotates into CCW direction because the selected rotational angle is ‘−22.5°’. To accomplish this response, the first step is ‘−15°’ and the second step is ‘−7.5°’. The RI_1 equals ‘2’ in this experiment to enable the designed circuit the freedom of choosing the best step angle to reach the destination. In the closed-loop part, it can be clearly noticed that how the rotor response has been improved for both steps as shown below.

Moreover, the rotor response will be tested in case of the required position is ‘$A_n = 37.5$’. However, the movement direction will be in CW in this case as shown in Fig. 12. Therefore, the faster way to complete this rotation is to move into two full steps and then a half step. In fact, this

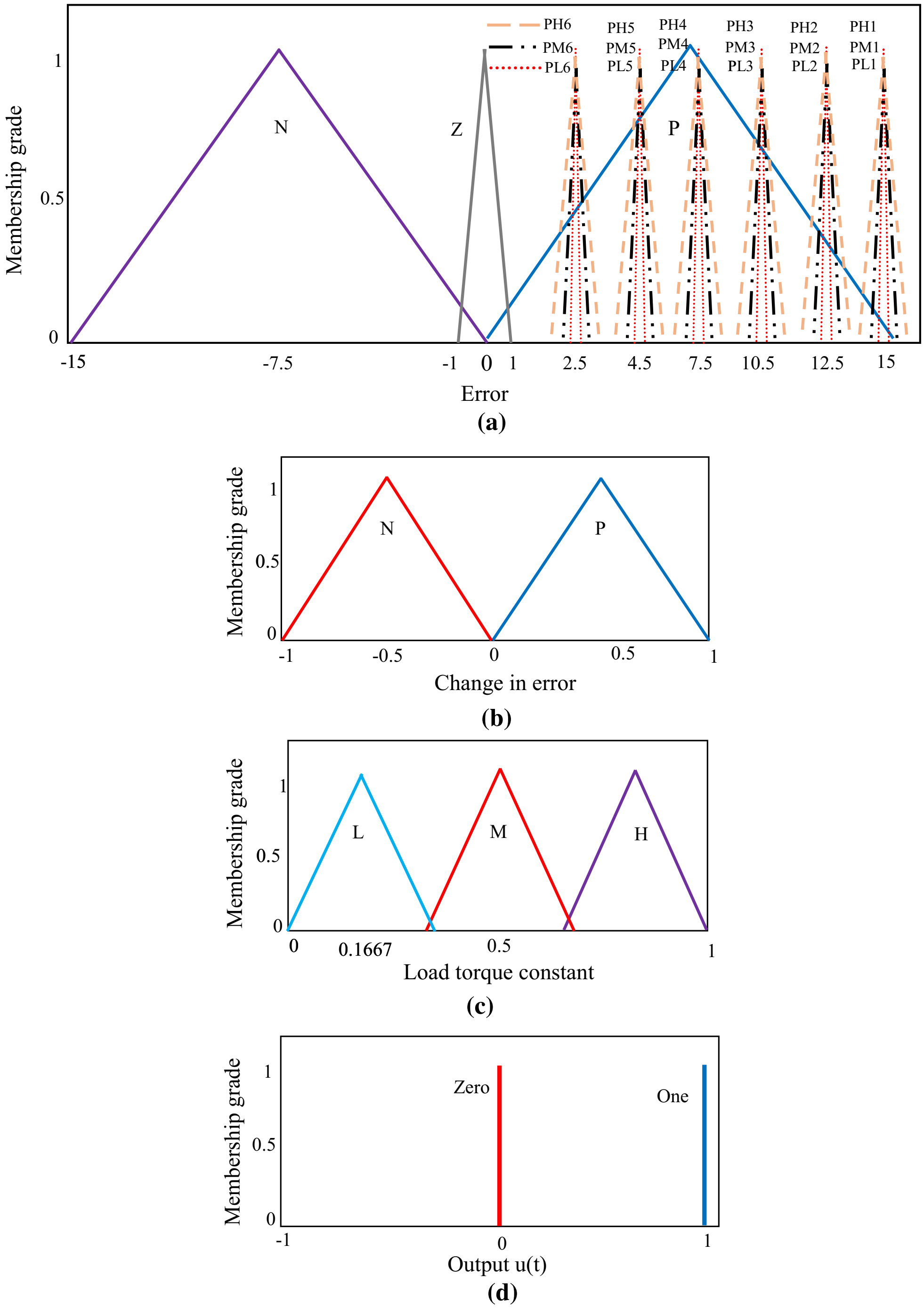


Fig. 8 Fuzzy memberships functions for **a** error signal, **b** change in error signal, **c** load torque constant, and **d** the output $u(t)$

is the faster response that can be achieved due to choosing '$\mathrm{RI}_1 = 2$'.

To make the driving circuit more intelligent, we added a privilege of rotation in opposite direction in case the input angle (An) is greater than '180°' so that the final position can be reached faster. In Fig. 13, nevertheless, the required angle is '330°' in CW direction. However, the rotor will rotate into CCW direction to reach the position because it will take just two steps to reach the same position with '22' steps when it follows the CW direction. Also, the overshoot has been improved significantly using the fuzzy controller as depicted in the blue line.

In the following experiment, we have chosen a value for the angle 'An' which is not multiple of the step angles. Although the chosen value is '$A_n = 42°$', the driving circuit will approximate this angle to the nearest value of the

Table 4 Rules between the error and the load torque constant against the output

Error $e(t)$	Load torque constant (K_w)		
	High (H)	Medium (M)	Low (L)
Positive high 1 (PH1)	One	–	–
Positive high 2 (PH2)	One	–	–
Positive high 3 (PH3)	One	–	–
Positive high 4 (PH4)	One	–	–
Positive high 5 (PH5)	One	–	–
Positive high 6 (PH6)	One	–	–
Positive medium 1 (PM1)	–	One	–
Positive medium 2 (PM2)	–	One	–
Positive medium 3 (PM3)	–	One	–
Positive medium 4 (PM4)	–	One	–
Positive medium 5 (PM5)	–	One	–
Positive medium 6 (PM6)	–	One	–
Positive low 1 (PL1)	–	–	One
Positive low 2 (PL2)	–	–	One
Positive low 3 (PL3)	–	–	One
Positive low 3 (PL3)	–	–	One
Positive low 3 (PL3)	–	–	One
Positive low 3 (PL3)	–	–	One

Table 5 Rules between the error and the change in error against the output

Error $e(t)$	Change in error $\Delta e(t)$	
	Positive	Negative
Positive (P)	One	Zero
Zero (Z)	One	One
Negative (N)	Zero	One

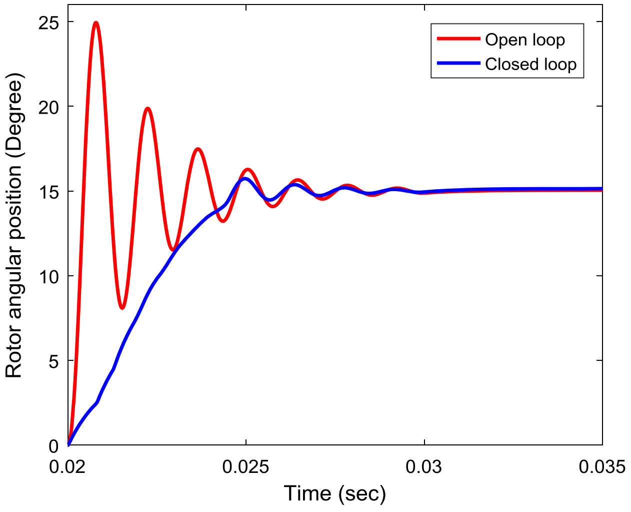
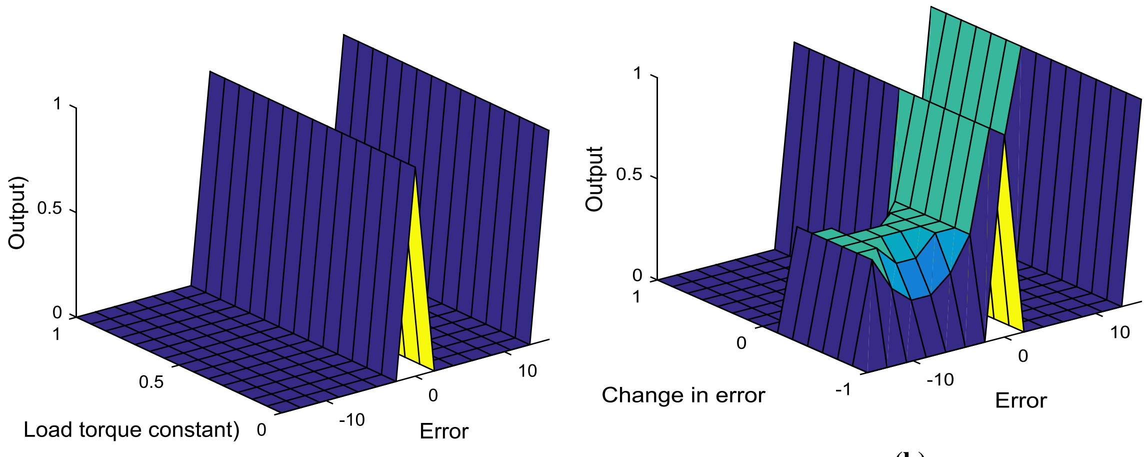


Fig. 10 Angular position response when '$A_n = 15°$' and '$\mathrm{RI}_1 = 0$' at no load

multiple of the step angles based on the resolution indicator 1 (RI_1). In this particular experiment, the approximated angle will be equal to '45°' because of the '$\mathrm{RI}_1 = 2$' and the angle '$A_n = 42°$' is closer to '45°' that is the multiple of the step angles. The open-loop response and the closed-loop response using the proposed PD fuzzy technique are shown in Fig. 14.

5.2 Case study II

In this case, the half of the load was applied when '$A_n = 15°$' and '$\mathrm{RI}_1 = 0$'. The rotor angular response has been reduced slightly due to this addition as shown Fig. 15. Also, the closed-loop system responses effectively to the added load. Hence, the overshoot is better compared to the no-



(a)

(b)

Fig. 9 Surface view of the rules of the proposed fuzzy controller for **a** the error and the load torque constant against the output, **b** the error and the change of error against the output

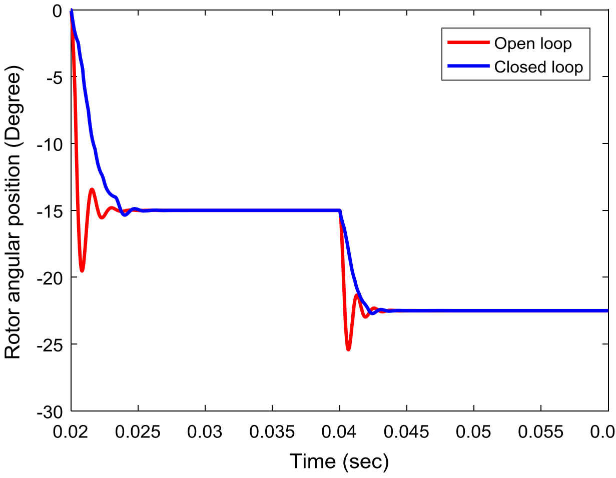


Fig. 11 Angular position response when ‘$A_n = -22.5°$’ and ‘$\mathrm{RI}_1 = 2$’ at no load

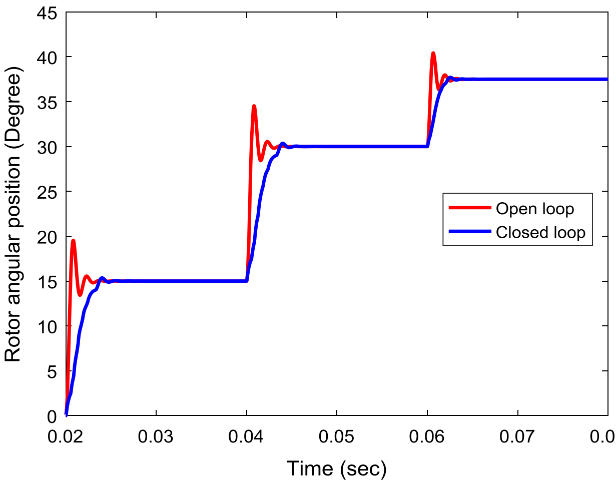


Fig. 12 Angular position response when ‘$A_n = 37.5°$’ and ‘$\mathrm{RI}_1 = 2$’ at no load

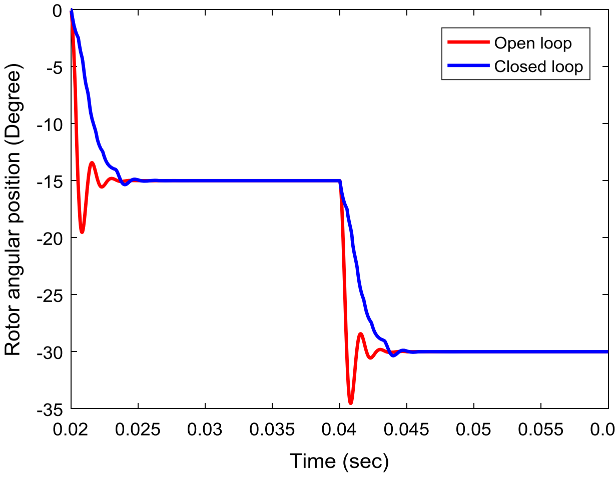


Fig. 13 Angular position response when ‘$A_n = 330°$’ and ‘$RI_1 = 2$’ at no load

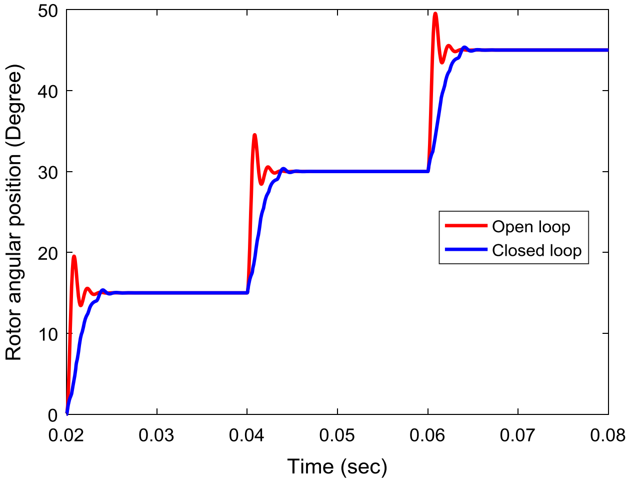


Fig. 14 Angular position response when ‘$A_n = 42°$’ and ‘$\mathrm{RI}_1 = 2$’ at no load

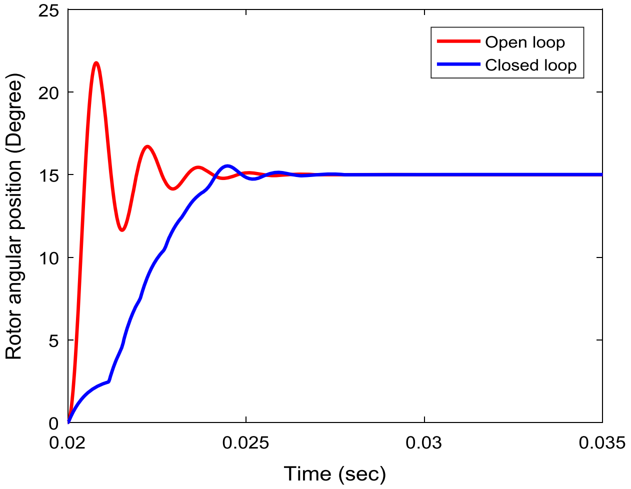


Fig. 15 Angular position response when ‘$A_n = 15°$’ and ‘$\mathrm{RI}_1 = 0$’ at half load

load case. From this conducted experiment, the percentage value of the overshoot was 53.3 at the no-load condition and then it became 4.6% after using the fuzzy algorithm. The results show rotor response has been improved further in case of some load existence. In addition, the settling time was improved which has been reduced from ‘7 ms’ to ‘6 ms’.

5.3 Case study III

Finally, in the case study, a full load has been applied to the VR stepper motor. This case allows to verify the tolerance of the response and shows how the rotor step response might be affected. We noticed that in this case the overshoot was reduced slightly compared to the previously conducted experiments. Figure 16 shows that the driving circuit has successfully guided the VR stepper motor to the required position and the step response has been improved successfully when the fuzzy technique is applied. The measured overshoot error is at a minimum which equals to 3.3% in case of using the proposed fuzzy controller. Also, the settling time is minimum and equals to '4 ms'.

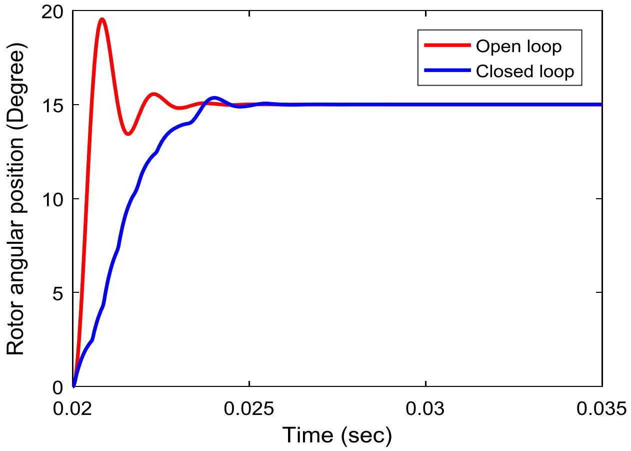


Fig. 16 Angular position response when '$A_n = 15°$' and '$\mathrm{RI}_1 = 0$' at full load

The comparison for three load cases is summarised as in Table 6 taking into consideration the overshoot and the settling time in case of the open-loop and closed-loop circuits. The comparison is also depicted graphically to show the difference between each load for the settling time and overshoot as given in Figs. 17 and 18, respectively.

6 Conclusion

The simulation results have proved successfully that the designed driving logic circuit can effectively control the rotor angular position. In particular, the driving circuit is capable of rotating the rotor in clockwise and anticlockwise directions with two step angles by employing different stepping sequences. It has been observed that the step response of the VR stepper motor suffers markedly from the overshoot and

Table 6 Comparison of the overshoot and settling time between the open- and closed-loop circuits

Parameter	No load		Half load		Full load	
	Open loop without FLC	Closed loop using FLC	Open loop without FLC	Closed loop using FLC	Open loop without FLC	Closed loop using FLC
Overshoot (%)	66.6	6	53.3	4.6	40	3.3
Settling time (ms)	10	8	7	6	5	4

Fig. 17 Settling time for open loop and closed loop under three different loads

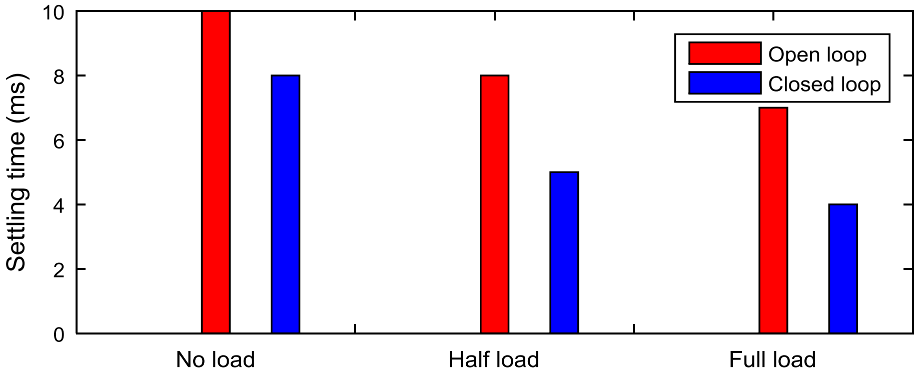
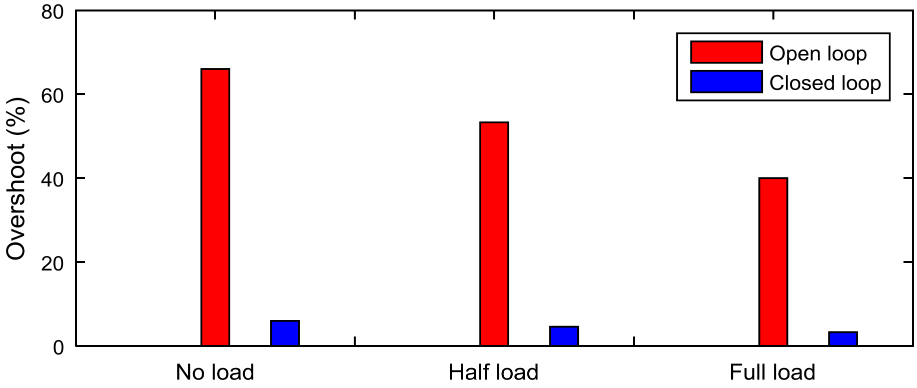


Fig. 18 Overshoot for open loop and closed loop under three different loads



oscillation. Therefore, the fuzzy control technique is introduced to improve the step position response. In addition, the proposed technique is conducted for the increments of the applied load based on three different conditions, i.e. no-load, half-load, and full-load condition. The overshoot has been greatly reduced to '3.3%', and settling time is minimised to '4 ms' at the full-load condition by using the proposed fuzzy technique. In all experiments, the oscillatory response is diminished significantly so that the resonance phenomenon is eliminated.

In the future work, other soft computing techniques such as neural networks and an adaptive neuro-fuzzy inference system can be introduced to compare the performance of the rotor step response with fuzzy control technique.

References

1. Bandyopadhyay B, Thakar V (2004) Multirate output feedback sliding mode control of permanent magnet stepper motor using flatness property. In: International conference on control applications, pp 1370–1375
2. Mort N, Abbod M, Linkens D (1995) Comparative study of fuzzy DC servo motors and stepper motors for mechatronic systems. In: IEE colloquium on innovations in manufacturing control through mechatronics, pp 1–5
3. Elsodany NM, Rezeka SF, Maharem NA (2011) Adaptive PID control of a stepper motor driving a flexible rotor. Alex Eng J 50(2):127–136
4. Kim W, Shin D, Lee Y, Chung CC (2016) Simplified torque modulated microstepping for position control of permanent magnet stepper motors. Mechatronics 35:162–172
5. De Leon-Morales J, Castro-Linares R, Guevara OH (2005) Observer-based controller for position regulation of stepping motor. IEE Proc Control Theory Appl 152(4):465–476
6. Harb AM, Zaher AA (2004) Nonlinear control of permanent magnet stepper motors. Commun Nonlinear Sci Numer Simul 9(4):443–458
7. Henke B, Sawodny O, Schmidt S, Neumann R (2013) Modeling of hybrid stepper motors for closed loop operation. In: 6th IFAC symposium on mechatronic systems and the international federation of automatic control, pp 177–183
8. Elksasy MSM, Gad HH (2010) A new technique for controlling hybrid stepper motor through modified PID controller. Int J Electr Comput Sci 10(02):28–35
9. Melkote H, Khorrami F, Jain S, Mattice MS (1999) Robust adaptive control of variable reluctance stepper motors. IEEE Trans Control Syst Technol 7(2):212–221
10. Aranjo B, Soori PK, Talukder P (2012) Stepper motor drives for robotic applications. In: IEEE international power engineering and optimization conference, pp 361–366
11. Greenough N, Kung CC (2013) A new high-efficiency stepper motor driver for old technology stepper motors. In: The 25th symposium on fusion engineering, pp 1–4
12. Hasanien HM (2011) FPGA implementation of adaptive ANN controller for speed regulation of permanent magnet stepper motor drives. Energy Convers Manag 52(2):1252–1257
13. Ghafari AS, Alasty A (2004) Design and real-time experimental implementation of gain scheduling PID fuzzy controller for hybrid stepper motor in micro-step operation. In: International conference on mechatronics, pp 421–426
14. Kabde AB, Savio AD (2014) Position control of stepping motor. Int J Adv Res Electr Electron Instrum Eng 3(4):8974–8981
15. Abdlaziz W, Salous S (2013) Comparison between PID and fuzzy logic control in the position of stepper motors. Eur J Sci Res 101(1):69–76
16. Ahmadi A, Sedehi MT, Yazdani AM, Haniff MF, Buyamin S, Rahim HA (2012) Designing an optimal fuzzy-PID controller for speed tracking of stepper motor. In: Symposium on industrial electronics and applications, pp 193–198
17. Athani VV (1997) Stepper motors: fundamentals, applications and design. New Age International Publisher, London
18. Kuo BC (1975) Theory and applications of step motors. West Publishing Company, Eagan
19. Shakya R, Rajanwal K, Patel S, Dinkar S (2014) Design and simulation of PD, PID and fuzzy logic controller for industrial application. Int J Inf Comput Technol 4(4):363–368
20. Patel N, Nigam MJ (2013) Design of fuzzy PD controller for inverted pendulum in real time. In: International conference on advances in computing, pp 955–962
21. Ban X, Gao XZ, Huang X (2004) Design of a hybrid fuzzy-PD controller for inverted pendulum. In: International conference on systems, man and cybernetics, pp 2286–2291
22. Jantzen J (2013) Foundations of fuzzy control: a practical approach, 2nd edn. Wiley, New York