IJEIS (Indonesian Journal of Electronics and Instrumentations Systems)
Vol.x, No.x, April xxxx, pp. 1~5
ISSN (print): 2088-3714, ISSN (online): 2460-7681
DOI: 10.22146/ijeis.xxxx

Model Reference Adaptive Control of a DC Motor Speed with Encoder using Hardware in the Loop

Muhammad Aqsal Mufhaddhal[1], Erwin Susanto[2], Muhammad Ridho Rosa[3]
[1,2,3] School of Electrical Engineering, Telkom University, Bandung, Indonesia
e-mail: [1] aqsalmufhaddhal@student.telkomuniversity.ac.id, [2] erwinelektro@telkomuniversity.ac.id
[3] mridhorosa@telkomuniversity.ac.id

Abstrak

Studi pada naskah ini bertujuan untuk merancang pengaturan kecepatan motor DC dengan encoder menggunakan Model Referrence Adaptive Control (MRAC) melalui sistem hardware in the loop (HIL). Meskipun saat ini kendali konvensional seperti kendali PID masih banyak ditemukan penggunaanya untuk kendali motor dc, namun kendali tersebut mempunyai keterbatasan dalam kemampuan beradaptasi terhadap ketidakpastian dan gangguan. Dengan demikian, makalah ini mengusulkan teknik kendali yang adaptif yang dapat memaksa sistem mengikuti perilaku model referensi meskipun gangguan dan ketidakpastian muncul dalam pengoperasiannya. Penelitian ini meliputi perancangan sistem perangkat keras secara lup tertutup yang terdiri dari motor DC plant, pengontrol MRAC, dan tampilan kontrol. Selain itu juga dilakukan percobaan untuk menguji kinerja kontrol dengan sinyal referensi sinusoidal dan unit step. Hasil pengujian menunjukkan bahwa kontrol MRAC berhasil mengikuti sinyal referensi dengan nilai selisih kuadrat rata-rata akar (Root Mean Squared Error, RMSE) yang rendah. Kesimpulannya, penelitian ini berhasil merancang suatu pengendalian pada motor DC dengan sensor encoder menggunakan Model Reference Adaptive Control pada perangkat keras sistem dan membuahkan hasil yang memuaskan pada pengujian eksperimen.
Kata kunci*—Motor DC, enkoder, hardware in the loop, MRAC, RMSE*

Abstract

This study aims to design speed control of a DC motor with an encoder using the Model Reference Adaptive Control (MRAC) via hardware in the loop (HIL) system. Although conventional control such as a PID control is still found in many dc motor control applications today, but they have limitations in adaptiveness from any uncertainties and noises. Therefore, this paper proposes a more adaptive control technique. Model Reference Adaptive Control which forces the real system to follow the behavior of the reference model system even though there is uncertainty in the system dynamics. This research includes the design of a hardware system on a loop consisting of a DC motor plant, MRAC controller, and control display. In addition, experiments were also carried out to test the performance control with sinusoidal reference signals and signal reference steps. The test results show that the MRAC control successfully follows the reference signal with low Root Mean Squared Error (RMSE) values. In conclusion, this study succeeded in designing a control on a DC encoder motor using the Model Reference Adaptive Control on the system hardware in a loop and yields satisfactory results in experimental testing.
Keywords—*DC motor, encoder, hardware in the loop, MRAC, RMSE*

1. INTRODUCTION

DC motors are the most used electric machines in various industrial and consumer applications [1], because of their ability to produce stable rotation at adjustable speeds, including in dc/dc boost converter and microgrids implementations [2][3]. In general, DC motor speed control is

Received June 1st,2012; Revised June 25th, 2012; Accepted October 10th, 2012

done by changing the value of the terminal voltage. One of the most used control methods is Proportional-Integral-Derivative (PID) Control [4]. The PID control considers the difference between the reference signal (set point) and the feedback signal and then produces a suitable output to adjust the motor speed. Although PID control is relatively simple and easy to implement, it has several drawbacks. Some of these drawbacks include sensitivity to load changes, inability to adapt to changing system parameters, and inability to cope with unwanted disturbances, for example a report on an iterative learning control to reduce the drawback of PD controller [5]. To overcome the drawbacks of PID control, several more advanced control methods have been developed. One of the promising control methods is Model Reference Adaptive Control (MRAC) [6]. MRAC is an adaptive control method that can modify controller parameters and can adapt to changes that occur in the system [7].

This research will design a tool to control DC motors using the adaptive control reference model on hardware in the loop. The performance of the system is measured by RMSE calculation [8],[9], [10]. The purpose of this study is to design a DC motor speed control with an encoder using hardware in the loop and to design a reference adaptive control model control method to control DC motor speed. The results show that MRAC can be applied to dc motor speed control with good performance i.e. low root mean square error values.

2. METHODS

In this section, we present the modelling of dc motor with encoder and how to derive the controller. Literature review is also conducted to show the previous reports regarding to our study, MRAC of dc motor for various applications

2.1 Literature Review and Modelling

Control design needs system modeling and describing control methods. DC motor is one type of actuator that is often used in various control systems, including in maximum power point tracking [11], gas industry [12], and robotics applications [13]. Its function is to provide rotary motion, which can later be combined with wheels or drums and cables to produce translational motion. To explain the working principle of a DC motor, it can be seen in Figure 1 which shows the armature electrical equivalent circuit and the rotor free body diagram.

The transfer function for DC motors can be seen in equation (1).

$$P(s) = \frac{\theta(s)}{V(s)} = \frac{K}{(Ls)b)(Ls)R))K^2} \quad (1)$$

State-space of the dc motor according to equation (1) can be written in equation (2) and equation (3).

$$\frac{d}{dt}\begin{pmatrix}\dot{\theta} \\ i\end{pmatrix} = \begin{pmatrix}-\frac{b}{J} & \frac{K}{J} \\ -\frac{K}{L} & -\frac{R}{L}\end{pmatrix}\begin{pmatrix}\dot{\theta} \\ i\end{pmatrix} + \begin{pmatrix}0 \\ \frac{1}{L}\end{pmatrix}V \quad (2)$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{pmatrix}\dot{\theta} \\ i\end{pmatrix} \quad (3)$$

Hardware in the loop (HIL) is a control technique that involves operating physical components connected to real-time simulations. Control systems that combine hardware and software usually fall into the category of physical systems.
In HIL, all, or part of the controlled process, consisting of actuators, physical processes, and sensors, can be simulated [14-17].

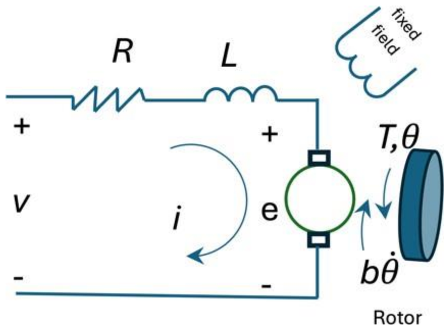


Figure 1 DC motor schematic

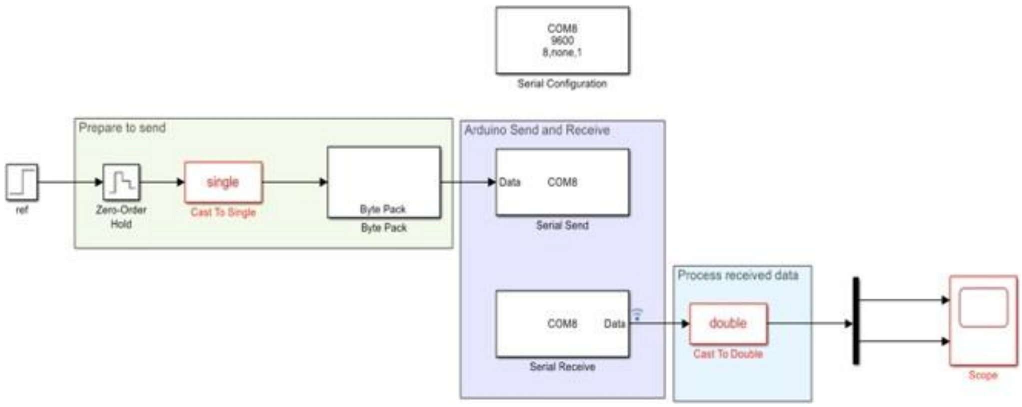


Figure 2 Serial communication block between Simulink and Arduino

Figure 2 shows the serial Simulink-Arduino as communication between Arduino and Simulink to build hardware in the loop system. There is a configuration to connect the laptop USB port with Arduino, serial send to send data from Simulink to Arduino and serial receive to receive data from Arduino to Simulink [18], [19].
The direct model reference adaptive control has an adaptive control reference that directly estimates control parameters adaptively. Many published reports on model reference adaptive, both direct and direct can be found [20-21].

The parameter symbols in Figure 1 can be described in Table 1.
This study uses state feedback direct MRAC high order with a system of order 2 with following explanations. System ordinary differential equation is:

$$\dot{x}(t) = Ax(t) + Bu(t) \qquad (4)$$

The reference model is:

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \qquad (5)$$

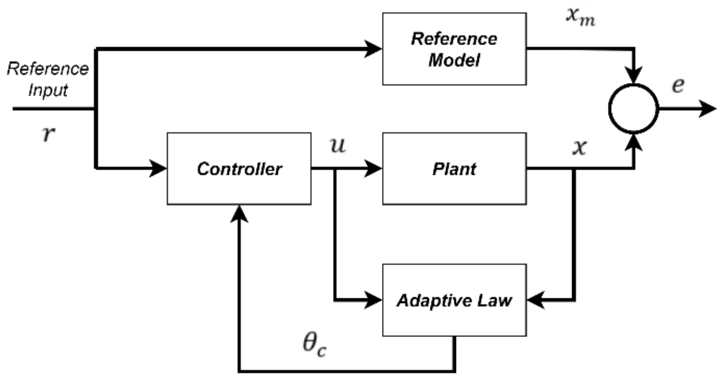


Figure 3 DC motor schematic Diagram of Direct Model Reference Adaptive Control

Direct MRAC control is given in equations (6) as follows:

$$u(t) = -k'(t)x(t) + l(t)r(t) \quad (6)$$

where k, l are state and reference gains.
The error dynamic is obtained from (4) and (5) with control action (6).

$$\dot{e}(t) = A_m e + B(-k'(t)x(t) + l(t)r(t)) \quad (7)$$

then the adaptive law becomes:

$$\dot{k}(t) = \gamma_1 B_m' Pe(x - x_m)x\, sgn(l) \quad (8)$$

$$\dot{l}(t) = \gamma_2 B_m' Pe(x - x_m)r\, sgn(l) \quad (9)$$

with γ_1, γ_2 are given constants.

2.2 Controller Design

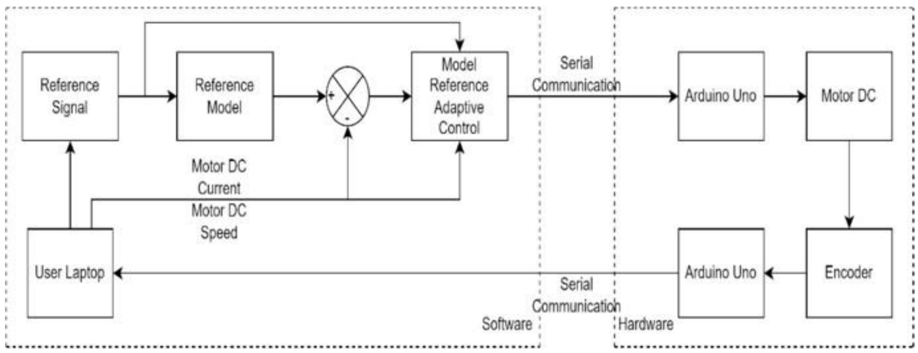


Figure 4 Construction of the controlled system with hardware in the loop

The system block diagram in Figure 4 shows the input in the form of a reference signal as the desired motor speed set point. The Reference Model is the reference for the adaptive control model to regulate the speed of the DC motor. The actual plant must follow the performance of the

Reference Model [22]. The reference adaptive control model uses speed and current to output a control signal (voltage) to regulate the speed of the DC motor. This control signal is sent to Arduino Uno to drive the DC motor, as usually found in many applications [23], [24], [25]. The encoder on the DC motor sends the motor speed (rad/s) to the Arduino. Finally, Arduino sends the motor current and speed (rad/s) to MATLAB as state feedback controller and the process is continuing looped as shown in process flowchart, Figure 5.

Figure 6 shows the hardware realization. It is shown that the hardware design consists of dc motor with encoder (component 1) as the plant to be controlled. Component 2 L298N motor driver functions to regulate the pulse and direction of the dc motor. Component 3 is the Arduino Uno as the processor data from the encoder to be sent to Simulink MATLAB and receive control data to regulate the motor speed.
The DC motor, motor driver and Arduino Uno hardware specifications used are:

- Brushed dc motor: 12 V, 1000 rpm (nominal speed), power consumption 5.4 – 15.6 W.
- Motor driver L298N: motor supply Voltage (maximum) 46 V, motor supply current (maximum) 2A, logic voltage 5V, driver voltage 5-35V, driver current 2A, logical current 0-36mA.
- Arduino Uno: input voltage 7-12V, 14 digital I/O with 6 PWM output, 6 analog input pin, 32KB flash memory, EEPROM, 16 MHz Clock Speed.

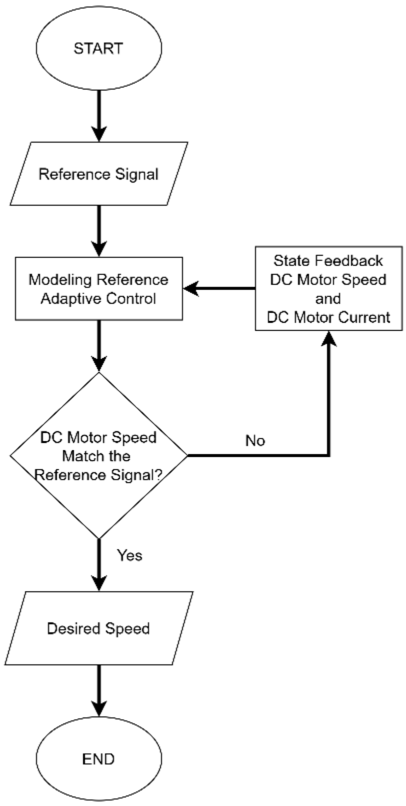


Figure 5 The flowchart of adaptive control for dc motor speed control

3. RESULTS AND DISCUSSION

In the testing, we control the DC motor with direct MRAC state feedback to see the response of the DC motor output to the reference system output. It is shown how the reference system output, the DC motor output adaptation to the reference system output and the effect of adaptive law on signal).

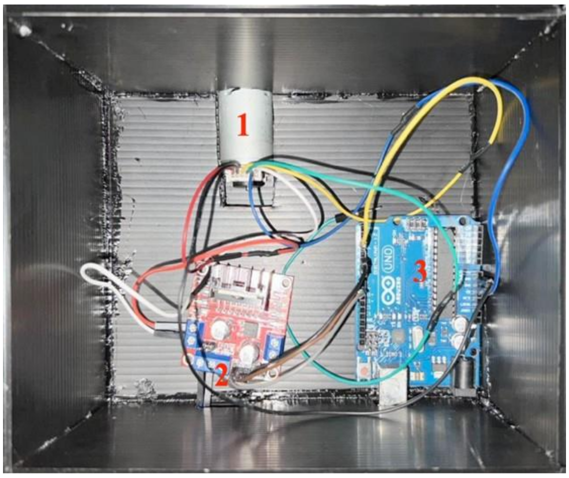
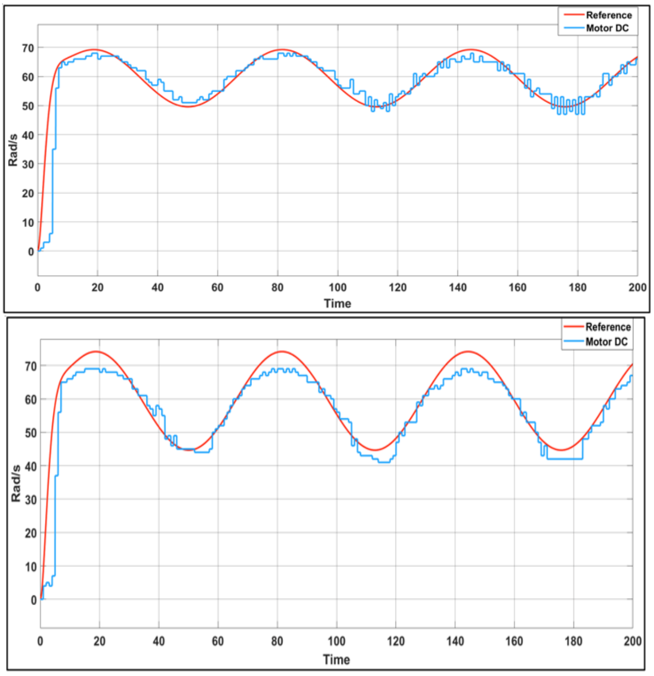


Figure 6 A DC motor (1), motor driver (2) and Arduino Uno(3) as the hardware components.

A dc motor parameter is available to provide the state space model of reference model i.e.: $J = 0.01$, $b = 0.01$, $K = 1$, $R = 1$, $L = 250$.MRAC on the DC motor output adaptation to the reference system. Two different signal inputs will be used, namely a sinusoidal signal and a step signal.



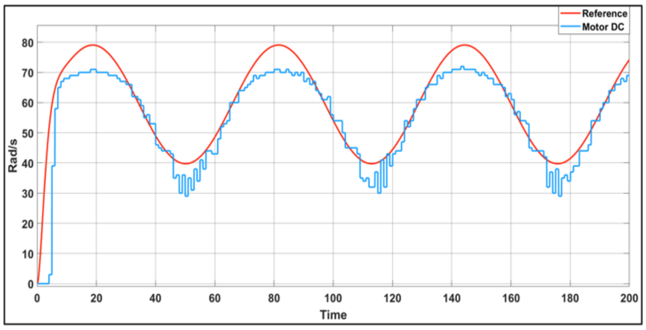


Figure 7 Trajectories of DC motor speed and reference input, sinusoidal input 10 rad/s, 15 rad/s and 20 rad/s

3.1 Sinusoidal Signal Input

Three different sinusoidal signals, 10, 15 and 20 rad/s are carried out. Parameters of each signal are bias of 60 rad/s, a frequency of 0.1 rad/s, a phase of 1 rad, and a sample time of 0 (continues time signal).
Therefore, the reference model used is:

$$\dot{x}_m = \begin{pmatrix} -1 & 100 \\ -0.004 & -0.004 \end{pmatrix} x_m + \begin{pmatrix} 0 \\ 0.004 \end{pmatrix} u \quad (10)$$

with constants $\gamma_1 = \begin{pmatrix} 0.00005 \\ 0.00005 \end{pmatrix}$, $\gamma_2 = [-0.00005]$, and weighting matrix $Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

Figure 7 is the process of a reference adaptive control model working on a DC motor real plant with a sinusoidal signal input. It is shown that the speed of the DC motor follows the reference signal in the form of a sinusoidal signal that has been designed.
The performance of the developed control is determined with the root mean squared error (RMSE) formula. The RMSE value for the sinusoidal signal input test is provided in Table 1.

Table 1 RMSE for sinusoidal rpm signal

RMSE of DC Motor Speed against Reference Model		
Amplitude 10 rad/s	Amplitude 15 rad/s	Amplitude 20 rad/s
2.15	3.25	4.94

Furthermore, adaptive speeds/rpm occur because of the influence of the constants k and l on the MRAC adaptive law equations (8) and (9), see Figure 8. Figure 8 shows how the constants k and l work for adaptation mechanism. Adaptive law affects the MRAC output as a value that drives the DC motor. According to the Table 1, it is shown that the speed of the DC motor can track the reference speed and it can be concluded that adaptive law works well.

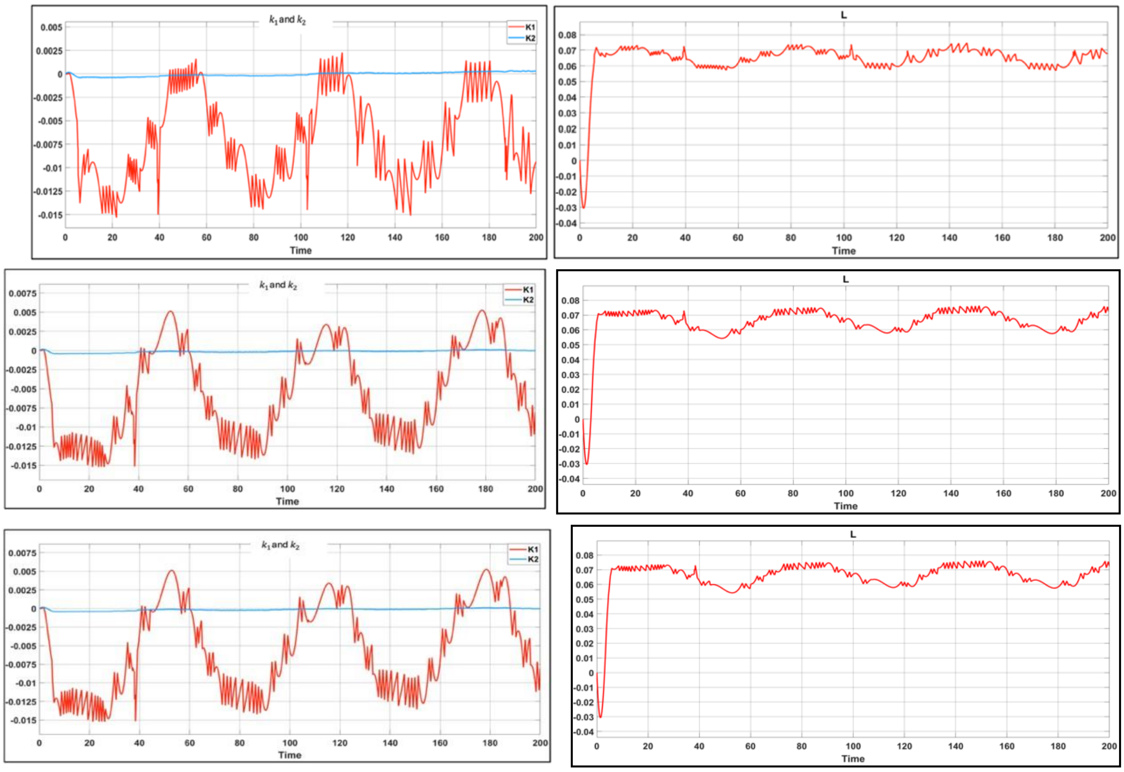


Figure 8 Trajectories of constants k_1, k_2 and L, sinusoidal input 10 rad/s, 15 rad/s and 20 rad/s respectively

3.2 *Unit step Signal Input*

This test uses a step signal input with a step time of 0 s, an initial value of 0 rad/s and three different final values of 45, and 75 rad/s.
Reference Model used is as in equation (10):

$$\dot{x}_m = \begin{pmatrix} -1 & 100 \\ -0.004 & -0.004 \end{pmatrix} x_m + \begin{pmatrix} 0 \\ 0.004 \end{pmatrix} u \qquad (11)$$

with constants
$\gamma_1 = \begin{bmatrix} 0.0001 \\ 0.0001 \end{bmatrix}$, $\gamma_2 = [-0.0001]$, and weighting matrix $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Figure 9 is the process of the reference adaptive control model working on a real plant DC motor with step signal input. The speed of the DC motor follows the reference signal that has been designed. The RMSE value for the step signal input test is provided in Table 2.

Table 2 RMSE for step rpm signal

RMSE of DC Motor Speed against Reference Model		
Step 45 rad/s	Step 60 rad/s	Step 75 rad/s
2.77	0.78	5.64

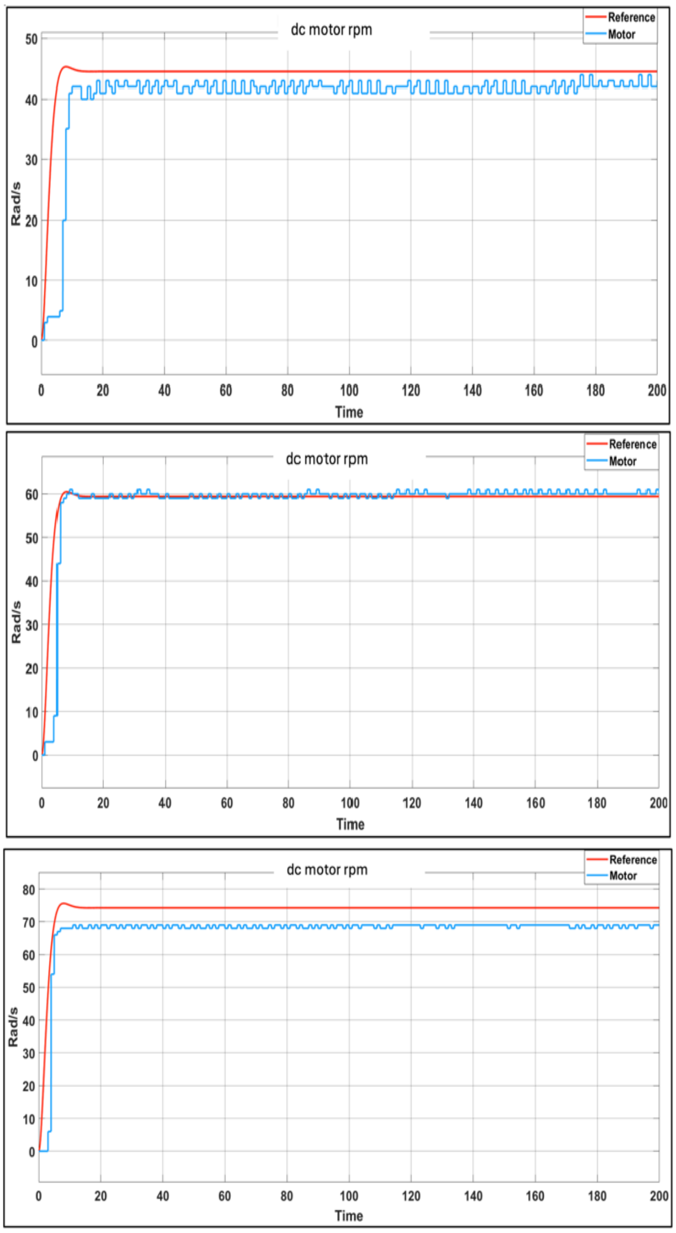


Figure 9 Trajectories of DC motor speed and reference input, step input 0-45 rad/s, 0-60 rad/s and 0-75 rad/s

Figure 10 shows how the constants k and l work on adaptive law. In experimental simulation with the step signal input, the constants k and l reach the steady values show the adaptive law affect the MRAC output as a value that drives the DC motor. Therefore, it can be concluded that adaptive law works well.

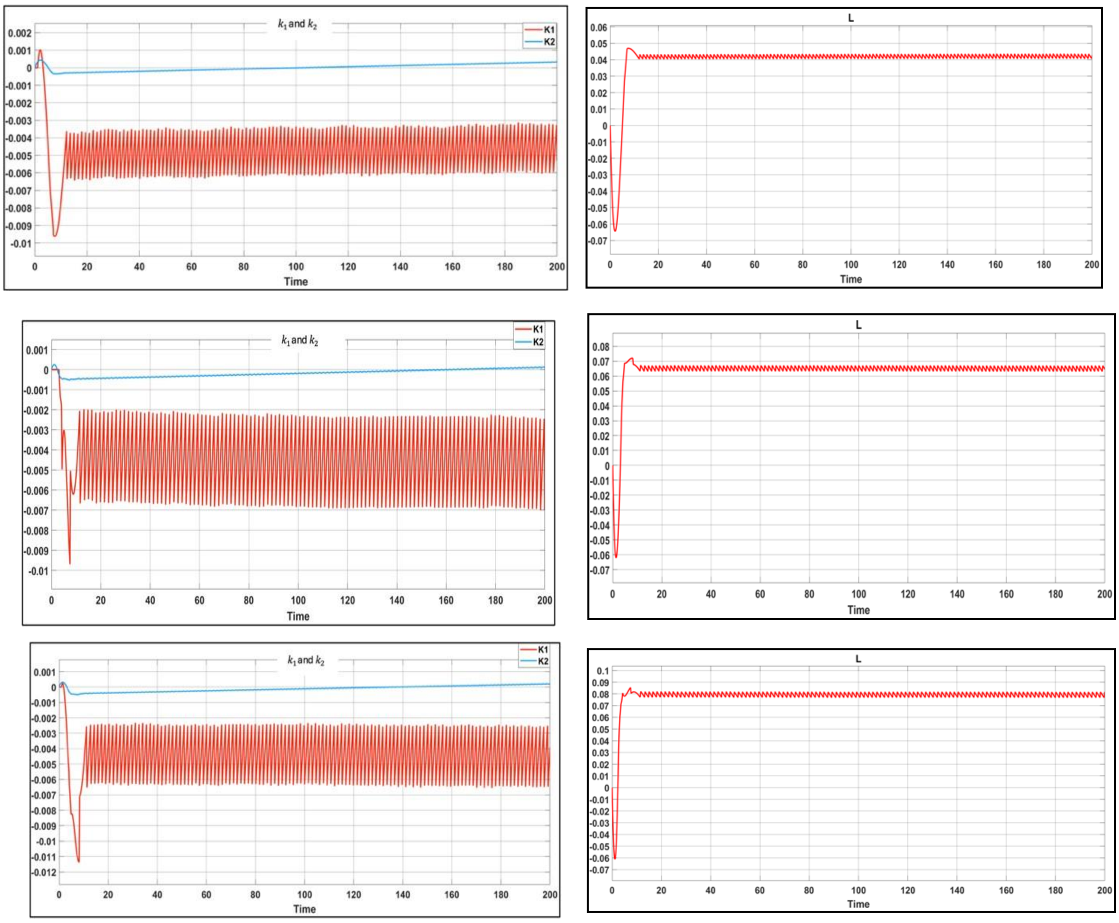


Figure 10 Trajectories of DC motor speed and reference input, step input 0-45 rad/s, 0-60 rad/s and 0-75 rad/s respectively

4. CONCLUSIONS

Based on the experimental simulation that have been carried out, this study concludes:

1. The Model Reference Adaptive controller has been designed for a real plant DC encoder motor using hardware in the loop system which is joining Simulink and microcontroller instrument.
2. In testing the sinusoidal signal reference input, the real plant encoder DC motor can follow the reference signal that has been designed with a RMSE (Root Mean Squared Error) value of 2.15 for an amplitude of 10 rad/s, 3.25 for an amplitude of 15 rad/s and 4.94 for an amplitude of 20 rad /s.
3. In testing the step signal reference input, the real plant encoder DC motor can follow the reference signal that has been designed with a RMSE (Root Mean Squared Error) value of 2.77 for step 45 rad/s, 0.78 for step 60 rad/s and 5.64 for step 75 rad /s.

Therefore, MRAC design shows the successful work, and it is shown that the constants k and l on the adaptive law work well and tend to steady condition in adapting the speed of the DC motor to the reference model as seen in Figures 8 and 10.

ACKNOWLEDGEMENTS

REFERENCES

[1] D. Mohanraj *et al*., “A Review of BLDC motor: state of art, advanced control techniques, and applications,” *IEEE Access*, vol. 10, pp. 54833-54869, 2022, doi: 10.1109/ACCESS.2022.3175011

[2] R. Silva-Ortigoza *et al*., “Hierarchical flatness-based control for velocity trajectory tracking of the “DC/DC boost converter–DC motor system powered by renewable energy,” *IEEE Access*, vol. 11, pp. 32464-32475, 2023, doi: 10.1109/ACCESS.2023.3260188.

[3] P. Wang, J. Zhao, and K. Liu, “Parameter-Adaptation-Based Virtual DC Motor Control Method for Energy Storage Converter,” *IEEE Access*, vol. 9, pp. 90795–90804, 2021, doi: 10.1109/ACCESS.2021.3091699

[4] W. Tan, W. Han, and J. Xu, “State-Space PID: A Missing Link Between Classical and Modern Control,” *IEEE Access*, vol. 10, pp. 116540–116553, 2022, doi: 10.1109/ACCESS.2022.3218657

[5] J. Dong, B. He, M. Ma, C. Zhang and G. Li, “Open-Closed-Loop PD Iterative Learning Control Corrected with the Angular Relationship of Output Vectors for a Flexible Manipulator,” *IEEE Access*, vol. 7, pp. 167815–167822, 2019, doi: 10.1109/ACCESS.2019.2930559

[6] A.A. El-Samahy, and M.A. Shamseldin, “Brushless DC motor tracking control using self-tuning fuzzy PID control and model reference adaptive control,” *Ain Shams Engineering Journal*, vol. 9, no. 3, pp. 341–352, 2018, https://doi.org/10.1016/j.asej.2016.02.004

[7] M.A. Qureshi, S. Musumeci, F. Torelli, A. Reatti, A. Mazza, and G. Chicco, “A novel model reference adaptive control approach investigation for power electronic converter applications,” *International Journal of Electrical Power & Energy Systems*, vol. 156, 109722, 2024, https://doi.org/10.1016/j.ijepes.2023.109722

[8] Q. Pan, X. Fan, and J. Li, “Connectivity matrix-based descriptors with deep learning for estimation of pure component properties,” *Computer Aided Chemical Engineering*, vol. 52, pp. 1957-1962, 2023, https://doi.org/10.1016/B978-0-443-15274-0.50311-5

[9] M. Singh, R.K Dubey, S. Kumar, “Vehicle telematics: An Internet of Things and Big Data approach,” *Artificial Intelligence and Machine Learning for EDGE Computing*, pp. 235-254, 2022, https://doi.org/10.1016/B978-0-12-824054-0.00019-8

[10] K. Rajesh and M. S. Saravanan, “Prediction of customer spending score for the shopping mall using Gaussian mixture model comparing with linear spline regression algorithm to reduce root mean square error,” *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 335-341, 2022, doi: 10.1109/ICICCS53718.2022.9788162

[11] M.M. Ahmed, W.S. Hassanien, M.A. Enany, “Modeling and evaluation of SC MPPT controllers for PVWPS based on DC motor,” *Energy Reports*, vol. 7, pp. 6044-6053, 2021, https://doi.org/10.1016/j.egyr.2021.09.055

[12] S. Ekinci, B. Hekimoğlu, and D. Izci, “Opposition based Henry gas solubility optimization as a novel algorithm for PID control of DC motor,” *Engineering Science and Technology, an International Journal*, vol. 24, no. 2, pp. 331-342, 2021, https://doi.org/10.1016/j.jestch.2020.08.011

[13] N. Promkaew, *et al*., “Development of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments,” *Results in Engineering*, pages 102280, 2024, https://doi.org/10.1016/j.rineng.2024.102280

[14] A. Fontanella *et al*., “Modeling the coupled aero-hydro-servo-dynamic response of 15 MW floating wind turbines with wind tunnel hardware in the loop,” *Renewable Energy*, vol. 219, no. 1, 119442, 2023, https://doi.org/10.1016/j.renene.2023.119442

[15] T. Jung, N. Jazdi, S. Krauß, C. Köllner, and M. Weyrich, “Hardware-in-the-Loop Simulation for a Dynamic Co-Simulation of Internet-of-Things-Components,” *Procedia CIRP*, vol. 93, pp. 1334–1339, 2020, https://doi.org/10.1016/j.procir.2020.03.073.

[16] G. N. Sahu and M. Law, “Hardware-in-the-loop simulator for emulation and active control of chatter,” *HardwareX*, vol. 11, e00273, 2022, https://doi.org/10.1016/j.ohx.2022.e00273.
[17] A. S. Kvalsund and D. Winkler, “Development of an Arduino-based, open-control interface for hardware in the loop applications,” *HardwareX*, vol. 16, e00488, 2023, https://doi.org/10.1016/j.ohx.2023.e00488
[18] A.Sepehri, A.A Asadi, G.K. Costa, andN. Sepehri, “A do-and-see approach for learning mechatronics concepts,” *Proceedings of the 5th International Conference of Control, Dynamic Systems, and Robotics (CDSR'18)*, Niagara Falls, Canada, June 7–9, 2018, doi: 10.11159/cdsr18.124.
[19] Leomariga, “How to communicate Simulink Arduino,” available at https://github.com/leomariga/Simulink-Arduino-Serial.
[20] S. Kersting and M. Buss, “Direct and indirect model reference adaptive control for multivariable piecewise affine systems,” *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5634–5649, Nov. 2017, doi: 10.1109/TAC.2017.2690060
[21] L. Zhang, Z. Zhu, X. Zhou and L. Sun, “Model reference adaptive control of cross-coupling hysteresis in piezoceramics with dynamic loads,” *IEEE Access*, vol. 10, pp. 14691–14697, 2022, doi: 10.1109/ACCESS.2022.3148325.
[22] R. Rajesh, S.N. Deepa, “Design of direct MRAC augmented with 2 DoF PIDD controller: An application to speed control of a servo plant,” *Journal of King Saud University - Engineering Sciences*, vol. 32, no. 5, pp. 310-320, 2020, https://doi.org/10.1016/j.jksues.2019.02.005.
[23] K. A. Jose, et al., “Speed synchronization of multiple BLDC motor drive using Arduino,” *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, Lonavla, India, pp. 1-5, 2023, doi: 10.1109/I2CT57861.2023.10126358.
[24] A. Ma’arif et al., “Control of DC motor using proportional integral derivative (PID): Arduino hardware implementation,” *2020 2nd International Conference on Industrial Electrical and Electronics (ICIEE)*, Lombok, Indonesia, pp. 74-78, 2020, doi: 10.1109/ICIEE49813.2020.9277258.
[25] H. M. Htun, A. N. Yakunin, H. S. Paing and K. Win, “Implementation of ANFIS controller for DC motor on an Arduino due board,” *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, St. Petersburg, Moscow, Russia, pp. 2668-2671, 2021, doi: 10.1109/ElConRus51938.2021.9396740.