UDK 621.8

DIFFERENTIALLY DRIVEN WHEELED ROBOT CONSTRUCTED USING LEGO MINDSTORMS COMPONENTS

M. Farhan[1]
[1]Baghdad University

Abstract: The design, modeling, and implementation of mobile robotic systems have emerged as a prominent research direction in the field of automation and intelligent systems. With continuous advancements in embedded control systems, sensors, and computational platforms, mobile robots have become increasingly capable of performing complex tasks in dynamic environments. These systems not only provide a practical foundation for theoretical research in kinematics, control theory, and artificial intelligence, but also offer real-world applicability in areas such as manufacturing, logistics, search and rescue operations, and educational robotics.

One of the most commonly used platforms for experimentation and demonstration of autonomous behavior is the line-following robot (LFR). The LFR serves as a foundational model for studying feedback control, real-time processing, and sensor integration. In order to ensure efficient line tracking, the robot must exhibit fast response times, precise path correction capabilities, and consistent behavior under varying environmental conditions. These requirements impose challenges on both hardware design and the development of robust software algorithms for motion control and sensor data processing.

This paper presents a comprehensive study on the design and realization of a line-following robot utilizing a differential drive mechanism built with Lego Mindstorms EV3 components. The robot is equipped with a color sensor for real-time line detection and employs a digital feedback control algorithm to adjust wheel velocities accordingly. A key focus of the work is to demonstrate that high-quality line-following behavior can be achieved using relatively simple control strategies, provided that proper system modeling and parameter tuning are carried out.

To validate the performance of the control system, simulation models were developed using MATLAB Simulink. These models allow for visualization and testing of the robot's dynamic response under various track geometries and sensor conditions, enabling refinement of the control parameters prior to physical implementation. The use of the EV3 software environment for programming ensures compatibility with the Lego hardware while maintaining flexibility in algorithm development.

The results of both simulation and real-world experiments confirm that even with low-complexity control logic, the robot is capable of achieving stable, accurate, and responsive tracking of predefined paths. The simplicity of the system also makes it suitable for educational purposes, allowing students and researchers to explore core principles of robotics, including sensor fusion, feedback control, and mechatronic integration. Furthermore, this work lays the groundwork for future research in enhancing the performance of mobile robots through the integration of machine learning, adaptive control techniques, and multi-sensor systems.

Keywords: Digital control, line following robot.

ДИФЕРЕНЦІАЛЬНО-ПРИВОДНИЙ КОЛІСНИЙ РОБОТ, СКОНСТРУЙОВАНИЙ З ВИКОРИСТАННЯМ КОМПОНЕНТІВ LEGO MINDSTORMS

Фархан М.[1]
[1]Багдадський університет

Анотація: Проєктування, моделювання та реалізація мобільних робототехнічних систем стали одним із провідних напрямів досліджень у галузі автоматизації та інтелектуальних

https://doi.org/10.31650/2618-0650-2025-7-2-45-58

систем. Завдяки безперервному розвитку вбудованих систем керування, сенсорних технологій і обчислювальних платформ, мобільні роботи отримали здатність виконувати складні завдання в динамічних середовищах. Такі системи не лише забезпечують практичну базу для теоретичних досліджень у галузях кінематики, теорії керування та штучного інтелекту, але й мають широке застосування в реальному житті – зокрема, в промисловості, логістиці, пошуково-рятувальних операціях та освітніх проєктах.

Однією з найпоширеніших платформ для експериментів і демонстрації автономної поведінки є робот, що слідує по лінії (LFR – Line-Following Robot). Такий тип роботів є базовою моделлю для вивчення систем зворотного зв'язку, обробки даних у реальному часі та інтеграції сенсорів. Для забезпечення ефективного слідування за лінією робот повинен демонструвати швидке реагування, точну корекцію траєкторії та стабільну роботу за різних зовнішніх умов. Ці вимоги створюють виклики як для апаратної реалізації, так і для розробки надійних алгоритмів керування та обробки даних із сенсорів.

У даній роботі представлено комплексне дослідження процесу проєктування та реалізації робота, що слідує за лінією, з використанням диференціального приводу, зібраного на основі компонентів Lego Mindstorms EV3. Робот оснащено кольоровим сенсором для виявлення лінії в реальному часі, а також цифровим алгоритмом зворотного зв'язку для регулювання швидкості обертання коліс. Основна увага зосереджена на демонстрації того, що високоякісне слідування за лінією можна забезпечити за допомогою відносно простих алгоритмів керування, за умови коректного моделювання системи та налаштування параметрів.

Для верифікації ефективності системи керування розроблено і проаналізовано моделі у середовищі MATLAB Simulink. Ці моделі дали змогу візуалізувати динамічну реакцію робота за різних конфігурацій траєкторії та умов роботи сенсорів, що дало змогу оптимізувати параметри керування до фізичної реалізації. Програмування здійснювалось у середовищі EV3 Software, що забезпечує сумісність з апаратною платформою Lego та гнучкість у розробці алгоритмів.

Результати моделювання та експериментів підтверджують, що навіть за використання алгоритмів з низькою обчислювальною складністю, робот здатен забезпечити стабільне, точне та швидке слідування за заданою траєкторією. Простота системи також робить її придатною для використання в освітньому середовищі, дозволяючи студентам і дослідникам вивчати основи робототехніки, включаючи сенсорну інтеграцію, зворотний зв'язок та мехатроніку. Крім того, ця робота створює основу для подальших досліджень у напрямі підвищення продуктивності мобільних роботів за рахунок впровадження методів машинного навчання, адаптивного керування та мультисенсорних систем.

Ключові слова: Цифрове керування, робот, що слідує за лінією.

1 INTRODUCTION

Robotics is a branch of engineering that involves the conception, design, manufacture and operation of robots. The field overlaps with electronics, computer science, artificial intelligence, mechatronics, nanotechnology, bioengineering and control engineering. Robots are mechatronic engineering products, capable of acting autonomously while implementing assigned behaviors in various physical environments. The developed use of robots in many areas makes the fundamental understanding of them fundamental [1-3].

In recent years there has been a rapid increase in the use of digital controllers in control system. It has become routinely practicable to design very complicated digital controllers and to carry out the extensive calculations required for their design. The current adoption of digital rather than analog control in robotics is due to the genuine advantages found in working with digital signals rather than continuous time signals[4-7].

The use of analog controllers in control engineering poses problems such as limited accuracy, susceptibility to noise and drift of power supply, cost ineffectiveness and less flexibility. Digital control systems are more suitable for modern control systems because of reduced cost, noise immunity and speed [8].

Line following robots need to adapt accurately, faster, efficiently and cheaply to changing operating conditions. The drawbacks prominent in analog controllers reduce their suitability in robotics. Hence, the necessity for digital controllers which provide better performance capabilities [9].

2 ANALYSIS OF LITERARY DATA AND RESOLVING THE PROBLEM

2.1 Lego Mindstorms line follower robot design

A line follower shown in Fig.1 is a mobile robot which is able to follow a visible line on a surface consisting of contrasting colours. To build and run the robot, the required hardware included; Lego EV3 brick, power supply, 2 large servo motors, a set of wheels, colour sensor, connector cables, beams, axles, bushes and pins. The EV3 brick formed part of the chassis, equipped with wheels. The servo motors are used to drive the two front wheels. Two rear small castor wheels supported the robot. The robot had a colour sensor mounted at the front end to identify the line. It is centered between the two front wheels, which are separated by a distance of 7.4 cm. It is designed to follow an oval track made of black electrical tape (18 mm wide) on a white surface.

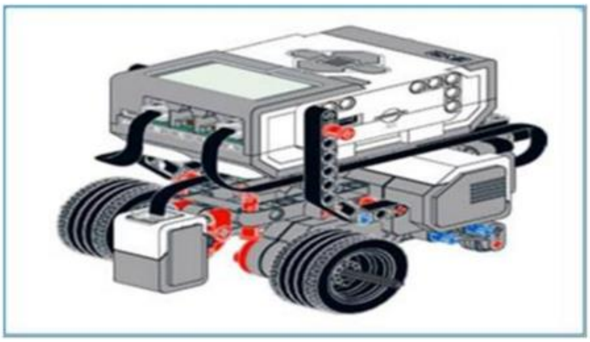

Fig. 1. Line following robot

2.2 Study of Lego Mindstorms EV3 motor

Lego Mindstorms has not published the EV3 motor's electromechanical characteristics. Table 1 shows the proposed parameters used in this paper, while, Table 2 shows the operational specifications.

Table 1

Lego Mindstorms EV3 large motor characteristics

Motor Parameter	Unit	Value
Torque constant	N.m/A	0.2
Back e.m.f. constant	V.s/rad	0.5
Armature resistance	Ω	5
Armature inductance	H	0.005
Viscous damping coefficient, B	N.m/rad.s	0.0006
Rotor inertia coefficient, J	N.m	0.001

Table 2

Operational specifications

Nominal Voltage	7.2V or 9V
Rotation Speed at no load	160 – 170 rpm
Running Torque	0.20 N-m
Stall Torque	0.40 N-m

Fig.2 shows the motor model simulated using MATLAB Simulink.

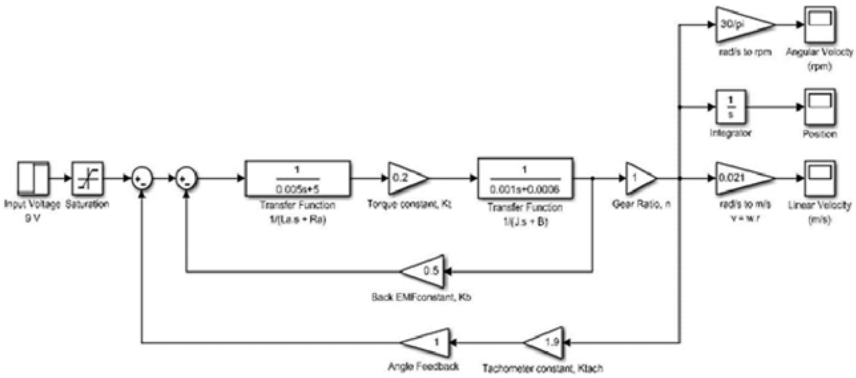

Fig. 2. Motor implementation on Simulink

2.3 Line following robot algorithms

Line following works by using the colour sensor (in reflected light intensity mode) to read the changes in the reflected light levels along the edge of a dark and light surface. The reflected light intensity is measured as a percentage from 0% (very low reflectivity) to 100% (very high reflectivity). More light is reflected from a white surface compared to the black surface. Depending on the light sensor value, the motors are directed to vary the speed.

In a program, white and black values are defined using a threshold value. Threshold is the average of the sensor value with the sensor on the black line and one found on the white area. Different measurements for black and white depend on factors such as the light level in the room, the robot's battery level, and the type of surface used.

The light sensor will read the light value. Then the robot can be programmed such that if the sensor sees black, which is when the sensor value is less than the threshold, the robot should turn right, else it should turn left. The basic line following approach is shown in Fig.3, and can be summarized as follows:

1. The robot will be started. It will then be set to move forward. It will be made to steer right until it detects the line edge.
2. Once the sensor sees black, the robot will continue to go forward while turning left gradually.
3. Whenever the sensor will see white (i.e. the robot leaving the line), the robot will turn to the right until the sensor finds black again.
4. The sequence then will be repeated in a loop, unless the robot is stopped.

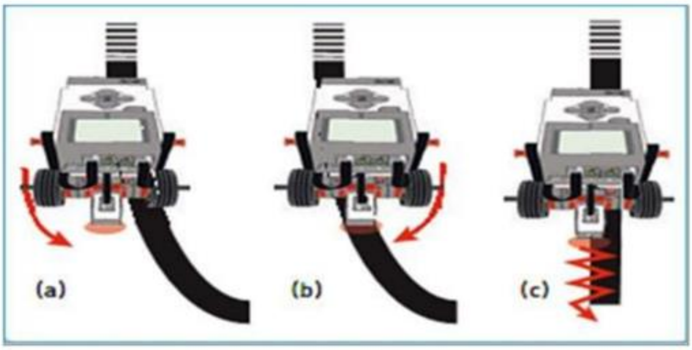

Fig. 3. Basic line following approach

2.4 Digital controller design

A robot without a controller will oscillate a lot about the line, leading to more consumption of battery power, less speed and following the line less efficiently.

When designing a line following robot, the transient response specifications are defined as:

1. Rise time: It is how fast the robot will try to get back to the line after it has drifted off.
2. Overshoot: The distance past the line edge the robot will tend to go as it is responding to an error.
3. The amount of overshoot indicates the relative stability of the system.
4. Steady-state error: The offset from the line as the robot follows a long straight line.
5. Settling time: The time the LFR will take to settle down when it encounters a turn.

The performance criteria are stipulated as follows:

1. Constant speed of 0.1m/s to be maintained despite the presence of turns.
2. Steady-state error: Less than 2%
3. Settling time of less than 0.1 seconds
4. Overshoot (%) of less than 1.0
5. Finite phase margin

The robot controller to be designed is to be modified until the transient response met is satisfactory.

2.4.1 Proposed controller design

The proposed controller is a Proportional-plus-Integral-plus-Derivative (PID) digital controller.

The PID controller would control the position of the robot with quick response time and minimize the overshoot. The proportional part would determine the magnitude of turn required to correct the error sensed. The integral part would improve the steady state error (proportional offset) which increases while the robot is not on the line. The derivative part would measure the deviation from the path and minimize overshoot. It would reduce the oscillating effect about the line. The derivative control is used to provide anticipative action.

2.5 Implementation of line following control algorithm for Lego Mindstorms EV3 hardware

Fig.4 shows the Simulink line tracking program with PID controller, while Fig.5 shows EV3 software line following program with PID controller. Sensors and motors contain blocks that interface with the EV3 hardware. Actual speed values block uses the values from each motor encoder to calculate the position and velocity of the robot. Desired velocity takes the user-provided velocity (m/s) and converts it into the desired state values for the velocity controller. Desired light takes the color sensor's white and black values to choose an appropriate reference value for the light. Velocity control has the PID controller implementation to control the forward velocity. Line tracking controller has the PID controller implementation to control the turning.

However, to download and run a line tracking Simulink model on the Lego Mindstorms EV3 robot, EV3 Wi-Fi Dongle or USB Ethernet Adaptor, and Wi-Fi Router are required to set up a network connection between EV3 brick and host computer.

The line following program is then written in EV3 software programming language. The black and white light intensity values are calibrated accordingly for the robot and the track. Using the provided USB cable, the program is downloaded and run on the robot. PID parameters (K_p, K_i and K_d) tuning is done experimentally to achieve smoother line tracking.

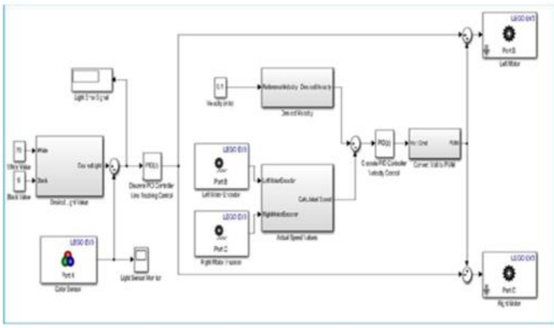

Fig. 4. Simulink line tracking program with PID controller

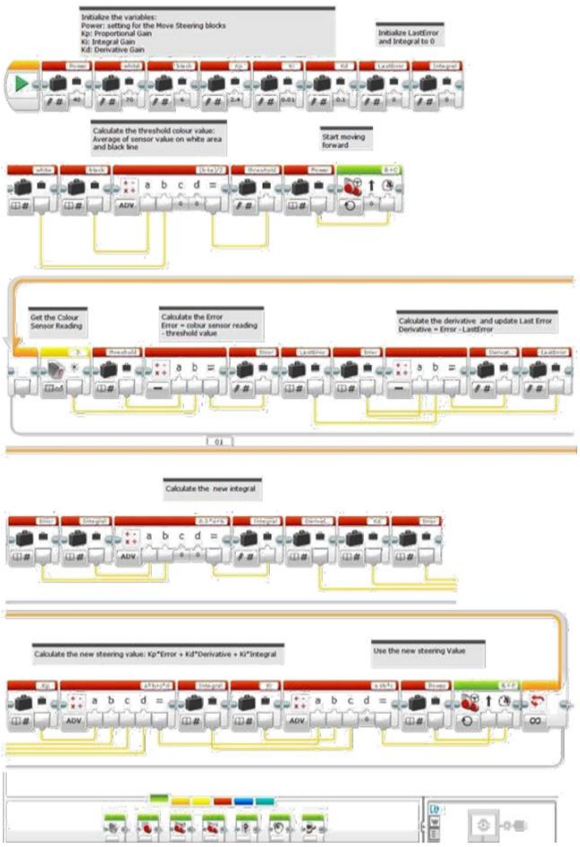

Fig. 5. EV3 Software line following program with PID controller

3 PURPOSE AND TASKS OF STADY

3.1 EV3 large motor characteristics

Table 3 shows the EV3 motor load characteristics, from which the linear relationship between power level and EV3 large motor speed noticeable as shown in Fig.6. Also, from table 3, the rotation speed of the EV3 large motor is proportional to the input voltage.

Table 3

EV3 motor load characteristics

Input Voltage	Torque	Rotation speed	Current	Mechanical power	Electrical power	Efficiency
4.5 V	17.3 N.cm	24 rpm	0.69 A	0.43 W	3.10 W	14 %
6.0 V	17.3 N.cm	51 rpm	0.69 A	0.92 W	4.14 W	22 %
7.5 V	17.3 N.cm	78 rpm	0.69 A	1.41 W	5.17 W	27 %
9.0 V	17.3 N.cm	105 rpm	0.69 A	1.90 W	6.21 W	31 %
10.5 V	17.3 N.cm	132 rpm	0.69 A	2.39 W	7.24 W	33 %
12.0 V	17.3 N.cm	153 rpm	0.69 A	2.77 W	8.28 W	33 %

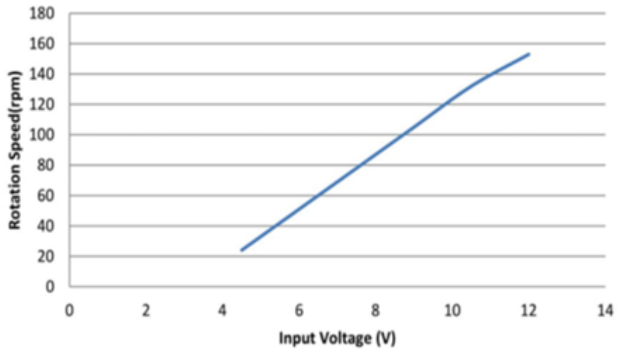

Fig. 6. Graph of rotation speed against applied voltage

3.2 PID parameters tuning

Different values of PID parameters (K_p, K_i, K_d) are chosen in order to get the step response.

1. For $K_p = 1,\ K_i = 0,\ K_d = 0$

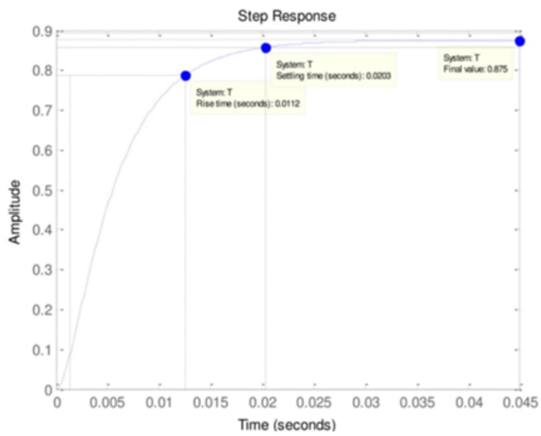

Fig. 7. Step response: $K_p = 1$

https://doi.org/10.31650/2618-0650-2025-7-2-45-57

Observations:
Rise time = 0.0112 seconds.
Settling time = 0.0203 seconds.
Final value = 0.875.

2. For $K_p = 5,\ K_i = 0,\ K_d = 0$

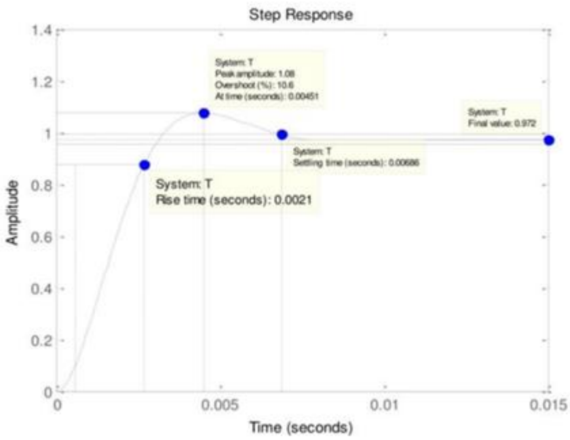

Fig. 8. Step response: $K_p = 5$

Observations:
Rise time = 0.0021 seconds.
Settling time = 0.00686 seconds.
Final value = 0.972.
Overshoot (%) = 10.6.
Peak amplitude = 1.08.

3. For $K_p = 2.4,\ K_i = 0,\ K_d = 0$

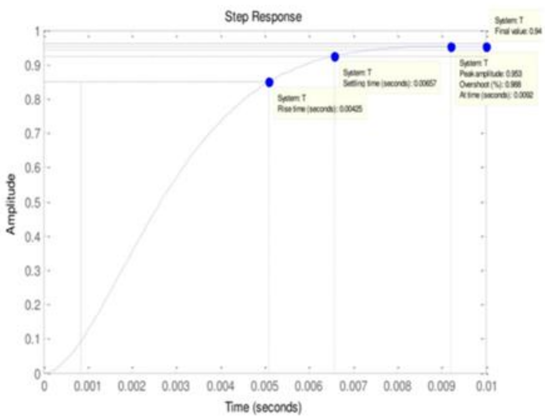

Fig. 9. Step response: $K_p = 2.4$

Observations:
Rise time = 0.00425 seconds.
Settling time = 0.00657.
Final value = 0.94.
Overshoot (%) = 0.988.
Peak amplitude = 0.953.

4. For $K_p = 2.4,\ K_i = 0.01,\ K_d = 0$

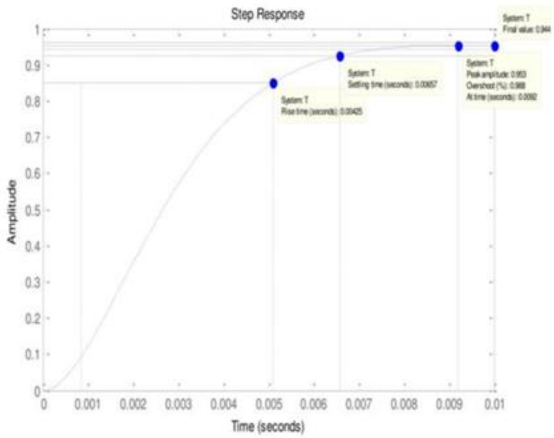

Fig. 10. Step response: $K_p = 2.4,\ K_i = 0.01,\ K_d = 0$

Observations:
Rise time = 0.00425 seconds.
Settling time = 0.00657 seconds.
Final value = 0.944.
Overshoot (%) = 0.988.
Peak amplitude = 0.988.

5. For $K_p = 2.4,\ K_i = 0.01,\ K_d = 0.1$

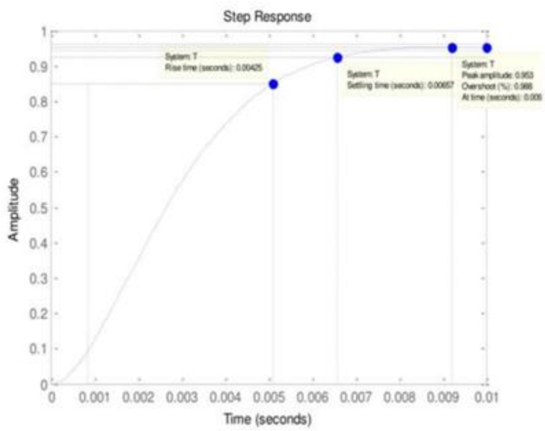

Fig. 11. Step response: $K_p = 2.4,\ K_i = 0.01,\ K_d = 0$

Observations:
Rise time = 0.00425 seconds.
Settling time = 0.00657 seconds.
Final value = 0.944.
Overshoot (%) = 0.988.
Peak amplitude = 0.953.

3.3 Frequency response

Frequency responses are obtained for different values of PID parameters.

1. For $K_p = 5,\ K_i = 0,\ K_d = 0$

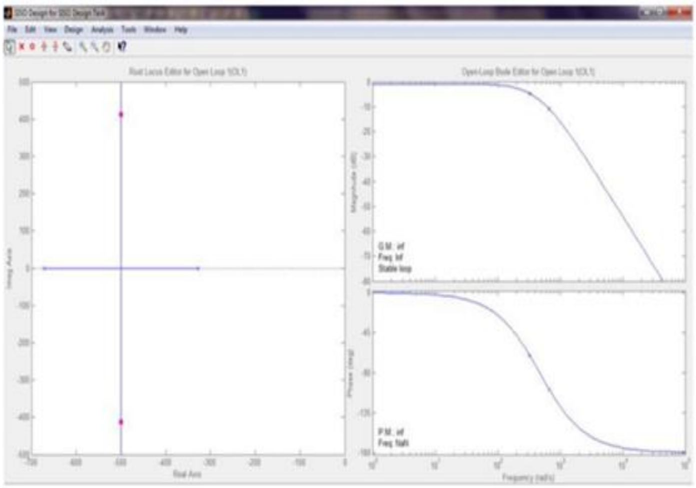

Fig. 12. Various frequency plots for the compensated system during tuning ($K_p = 5$)

<u>Observations:</u>
The root locus exhibited complex closed loop poles.
Both the phase and gain margin are infinite.
The system is stable.

2. For $K_p = 2.4,\ K_i = 0.01,\ K_d = 0.1$

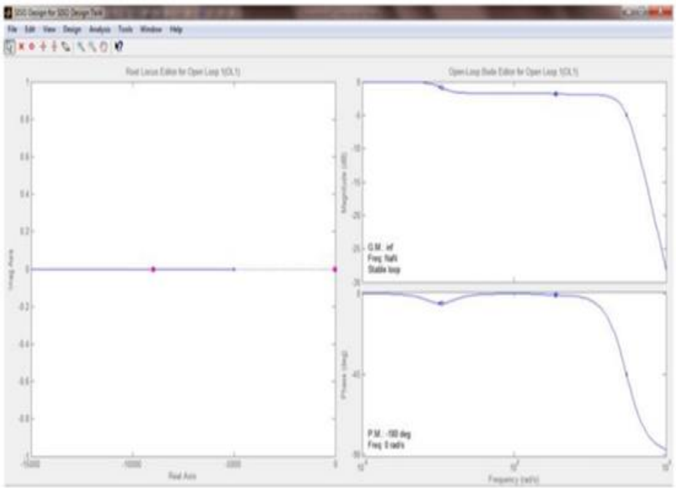

Fig. 13. Various frequency plots for the compensated system ($K_p = 2.4,\ K_i = 0.01,\ K_d = 0.1$)

<u>Observations:</u>
The root locus closed-loop poles changed from complex to real.
The system is still stable.
The infinite gain margin showed inherent stability.

4 BASIC RESULTS

The effects of each of controller parameters, K_p, K_i, and K_d on the line following robot are summarized in the table 4.

https://doi.org/10.31650/2618-0650-2025-7-2-45-57

Table 4

Effects of increasing PID parameters

Parameter	Rise Time	Overshoot	Settling Time	Steady-state Error
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Eliminate
K_d	Small change	Decrease	Increase	No change

The difficulty of tuning increased with the number of parameters that are to be adjusted. To observe the response that resulted from the tuning adjustments, it is necessary to wait for several minutes. This made the tuning by trial-and-error a tedious and time-consuming task.

In practice, the stability of a mathematical model is not sufficient to guarantee acceptable system performance or even to guarantee the stability of the physical system that the model represented. This is because of the approximate nature of mathematical models.

The main problems associated with the implementation of digital control are related to the effects of quantization and sampling. The advantages of digital control outweigh its implementation problems for most of the applications.

5 DISCUSSION OF THE RESULTS OF THE STUDY

The study demonstrates the effectiveness of a simple line-following robot system combining low-cost hardware with a straightforward digital control algorithm. Both experimental and simulation results showed reliable line-tracking performance across various conditions, with calibration of the color sensor and tuning of control parameters proving critical. Minor variations in lighting or surface reflectivity affected sensor readings, but adjusting detection thresholds and proportional gains minimized deviations. MATLAB Simulink simulations provided insight into dynamic behavior, allowing iterative testing of control strategies and reducing physical debugging time.

Differential drive control with a proportional controller was sufficient for stable tracking on moderately curved paths, though sharper curves revealed limitations, suggesting the potential for PID control or sensor fusion improvements. Implementation on the Lego Mindstorms EV3 platform facilitated rapid development, intuitive debugging, and real-time testing, making it suitable for education and prototyping. Overall, the results show that accessible, low-cost systems can achieve robust autonomy and control accuracy, providing a foundation for future work on adaptive navigation, obstacle avoidance, and machine learning-based path optimization.

6 CONCLUSIONS

A line following robot was designed and built using Lego Mindstorms EV3 components. Digital control algorithms were developed. The advantages and limitations of implementing the digital control on different software were studied. The effectiveness of using PID controller for optimum line tracking was demonstrated by inspecting the movement pattern of the robot while following the track. To obtain the desired control response, K_p, K_i, and K_d were successfully determined by tuning.

7 ETHICAL DECLARATIONS

The author has no relevant financial or non-financial interests to report.

References

1. Mutheke, M. (2015). *Digital control of a line following robot*. Graduation Project, University of Nairobi, Kenya.
2. Pathak, A., Pathan, R. K., Tutul, A. U., Tousi, N. T., Rubaba, A. S., & Bithi, N. Y. (2017). Line follower robot for industrial manufacturing process. *International Journal of Engineering Inventions, 6*(10), 10–17. Retrieved from https://www.ijeijournal.com/papers/Vol.6-Iss.10/B0610011017.pdf
3. Chowdhury, N. H., Khushi, D., & Rashid, M. M. (2017). Algorithm for line follower robots to follow critical paths with minimum number of sensors. *International Journal of Computer, 24*(1), 13–22. Retrieved from https://ijcjournal.org/InternationalJournalOfComputer/article/view/819
4. Kumaresan, P., Priya, G., Kavitha, B. R., Ramya, G., & LawanyaShri, M. (2017). Case study: A line following robot for hospital management. *International Journal of Pure and Applied Mathematics, 116*(24), 529–537.
5. Attar, A., Ansari, A., Desai, A., Khan, S., & Sonawale, D. (2017). Line follower and obstacle avoidance bot using Arduino. *International Journal of Advanced Computational Engineering and Networking, 5*(4), 18–21.
6. Tayal, S., Rao, H. P. G., Bhardwaj, S., & Aggarwal, H. (2020). Line follower robot: Design and hardware application. In *Proceedings of the 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)* (pp. 1–5). IEEE.
7. Amorim, J. S. J. C. C., Fernandes, J. H. O., Canella, A. L. C., Santos, T. M. B., Lima, J., & Pinto, M. F. (2023). Control tuning approach and digital filter application for competitive line follower robot. In *Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE)* (pp. 1–6). IEEE.
8. Pavón, W., Campaña, J., Santamaria, P., Iza, C., & Zapata, R. (2023). Intelligent control system for line follower robot: From theory to implementation on the mechatronics field. In *Proceedings of the 2023 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)* (pp. 1–6). IEEE.
9. Chekuri, R. R., Shivaram, V., Shravan, K., Sai, T. V., Chinnaiah, M. C., & Krishna, D. H. (2024). Design and implementation of line follower robot on SoC with SDK. In *Proceedings of the 2024 3rd International Conference for Innovation in Technology (INOCON)* (pp. 1–6). IEEE.
10. Kaiser, M., & Islam, M. (2017). Line follower robot: Fabrication and accuracy measurement by data analysis. *International Journal of Robotics and Automation, 4*(1), 1–10.
11. Baharuddin, M. S., & Abidin, M. Z. (2017). Analysis of line sensor configuration for the advanced line follower robot. *Journal of Robotics and Control, 3*(2), 45–50.

Література

1. Mutheke, M. (2015). *Digital control of a line following robot*. Graduation Project, University of Nairobi, Kenya.
2. Pathak, A., Pathan, R. K., Tutul, A. U., Tousi, N. T., Rubaba, A. S., & Bithi, N. Y. (2017). Line follower robot for industrial manufacturing process. *International Journal of Engineering Inventions, 6*(10), 10–17. Retrieved from https://www.ijeijournal.com/papers/Vol.6-Iss.10/B0610011017.pdf
3. Chowdhury, N. H., Khushi, D., & Rashid, M. M. (2017). Algorithm for line follower robots to follow critical paths with minimum number of sensors. *International Journal of Computer, 24*(1), 13–22. Retrieved from https://ijcjournal.org/InternationalJournalOfComputer/article/view/819
4. Kumaresan, P., Priya, G., Kavitha, B. R., Ramya, G., & LawanyaShri, M. (2017). Case study: A line following robot for hospital management. *International Journal of Pure and Applied Mathematics, 116*(24), 529–537.

5. Attar, A., Ansari, A., Desai, A., Khan, S., & Sonawale, D. (2017). Line follower and obstacle avoidance bot using Arduino. *International Journal of Advanced Computational Engineering and Networking, 5*(4), 18–21.
6. Tayal, S., Rao, H. P. G., Bhardwaj, S., & Aggarwal, H. (2020). Line follower robot: Design and hardware application. In *Proceedings of the 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)* (pp. 1–5). IEEE.
7. Amorim, J. S. J. C. C., Fernandes, J. H. O., Canella, A. L. C., Santos, T. M. B., Lima, J., & Pinto, M. F. (2023). Control tuning approach and digital filter application for competitive line follower robot. In *Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE)* (pp. 1–6). IEEE.
8. Pavón, W., Campaña, J., Santamaria, P., Iza, C., & Zapata, R. (2023). Intelligent control system for line follower robot: From theory to implementation on the mechatronics field. In *Proceedings of the 2023 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)* (pp. 1–6). IEEE.
9. Chekuri, R. R., Shivaram, V., Shravan, K., Sai, T. V., Chinnaiah, M. C., & Krishna, D. H. (2024). Design and implementation of line follower robot on SoC with SDK. In *Proceedings of the 2024 3rd International Conference for Innovation in Technology (INOCON)* (pp. 1–6). IEEE.
10. Kaiser, M., & Islam, M. (2017). Line follower robot: Fabrication and accuracy measurement by data analysis. *International Journal of Robotics and Automation, 4*(1), 1–10.
11. Baharuddin, M. S., & Abidin, M. Z. (2017). Analysis of line sensor configuration for the advanced line follower robot. *Journal of Robotics and Control, 3*(2), 45–50.

Mhnd Farhan
Baghdad University,
assistant prof.
Ph.D.
Iraq, Baghdad
mhndfarhan@yahoo.com
ORCID: 0009-0007-0541-6995

For references:
M. Farhan (2025). Differentially driven wheeled robot constructed using LEGO mindstorms components. Mechanics and mathematical methods. VII (2). 45–57.

Для посилань:
Фархан М.. Диференціально-приводний колісний робот, сконструйований з використанням компонентів LEGO MINDSTORMS. Механіка та математичні методи, 2025. Т. VII. № 2. С. 45–57.