Control Tunning Approach and Digital Filter Application for Competitive Line Follower Robot

[1]Johann S. J. C. C. Amorim, [1]Jefferson H. O. Fernandes, [1]André L. C. Canella, [2]Tatiana M. B. Santos, [3,4]José Lima, [1] Milena F. Pinto
[1]*Federal Center for Technological Education of Rio de Janeiro - CEFET/RJ*
Rio de Janeiro, RJ, Brazil
[2]*Federal Fluminense University - UFF*
Niteroi, RJ, Brazil
[3] *Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança, Portugal*
[4] *INESC Technology and Science, Porto, Portugal*
correspondent author: johann.amorim@aluno.cefet-rj.br

2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE) | 979-8-3503-1538-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/LARS/SBR/WRE59448.2023.10332987

***Abstract*—This research describes the development of a control strategy to optimize a competitive line follower robot for standard races. The innovative approach stems from the WolfBotz team at CEFET/RJ, presenting a thorough exploration of mathematical foundations, hardware design, control analysis, and how to implement this system in a microcontroller. This research complements a previous work that shows all the regulations used in Brazilian competitions and describes the controllers used in the system, such as angular and linear control. This research emphasizes all the changes between the two versions of Line Follower robots. The emphasis on mathematical foundations and integrating digital signal processing techniques like digital filters set the stage for robust sensor data interpretation. The tuning and optimization of dual controllers for track stability and linear velocity regulation represent a significant innovation, augmenting the robot's overall performance.**

***Index Terms*—Line Follower, PID, Digital Filters, Wheeled Robot**

I. INTRODUCTION

Robots are being implemented massively around the world. One type of robot frequently used for automation is the Line Follower robot, which aims to follow a pre-determined track using some line [1]. This robot can follow a specific track in a factory to transport some products autonomously without needing to map the infrastructure and use expensive technologies, such as GPS. However, any autonomous mobile system must use a controller that will make the system stable on the working unit system, in this situation, angular velocity and linear velocity. Therefore, to implement autonomous systems, it is paramount to carry out studies to apply controls in the system so it can be implemented [2].

For this purpose, a competitive line follower robot was used in the previous article [3] that makes research to improve the robot's performance. The authors used a line follower robot applied in competitions in Brazil following the regulation [4] to verify the advantage of using two controllers on the robot's system. Now, this present research will be described the development of the new version of the line follower robot, its design, changes, and improvements in hardware and software, digital filters applied to sensor readings, new changes on the controllers, and how these factors influenced the robot.

This study draws inspiration from existing literature on line follower robots, exemplified by several notable articles. For instance, Serrano et al. [5] delved into creating a budget-friendly robot, shedding light on cost-effective design methodologies. Similarly, Babu et al. [6] contributed by investigating a PID control tailored for line follower robots, harnessing a sole input variable for enhanced control efficacy. Notably, Nugraha et al. [7] delved into the realm of fuzzy logic applications in line follower robots, introducing an alternative dimension to the domain of control strategies. Furthermore, prior research has explored the intricacies of line identification, grappling with sensor-induced path detection errors as evidenced by Engin et al. [8] and the work of Amorim et al. [3], which notably introduced dual controls to elevate the performance of competitive line follower robots.

In addition, the motivation for this research is to improve the competitive line follower robot of the WolfBotz team of CEFET-RJ. By comprehensively grasping the needs of a competitive line follower robot's functioning, the potential emerges to address the industry's requisites for diverse mobile robotic platforms that mirror the dynamic essence of line follower robots [9]. This endeavor not only holds the promise of enhancing line follower performance but also extending their impact across various industrial domains.

To improve the line follower robot's performance, this research will emphasize the tuning of two control strategies used on this system and the importance of filtering digital signals. Both controls are used to stabilize the robot. Regarding digital signals, reading some information with sensors is susceptible to noise due to electrical contacts, electromagnetic interference, sample time, etc. To avoid or to reduce this problem, some concepts of digital signal processing were used in this research. This research presents an innovative approach to enhancing competitive line follower robots in standard races. Emerging from the WolfBotz teams of CEFET/RJ, this research goes beyond the conventional by thoroughly exploring mathematical foundations and hardware designs, and, by showing the use of advanced control strategies and digital signal processing techniques to other researchers of line follower robots that compete in standard races. Incorporating digital filters and the depth of hardware and software enhancements collectively underscore the collaborative essence of this study for other teams in this league of line-follower robots.

II. Concepts of a Line Follower Robot

A. Differential Robot

For this line follower robot, it was used the concept of a differential robot. This concept uses differential speeds in our rotation direction [10]. For this research, the robot was built in this concept because the mathematical model is simpler to understand and to model the control. Furthermore, it is possible to get some information about the track. For example, using two motorized wheels with fixed axles is only necessary to need the wheel rotation and its influence on the system.

As depicted in Figure 1, the equations dictating angular and linear velocities can be deduced, as detailed in Equations 1 and 2. Moreover, by referring to Figure 2, it becomes possible to derive an equation for radius curves that holds significance in system control. Employing Equation 3 facilitates the determination of the robot's maximum attainable linear velocity while staying on the track, encompassing radius information incorporated in Equation 4. Note that the variable L is the robot's length, ΔS_R is the right wheel displacement, and ΔS_L is the left wheel displacement. R is the curve radius, g is the gravity acceleration, and μ_s is the coefficient of static friction, which is 0.46 on this robot.

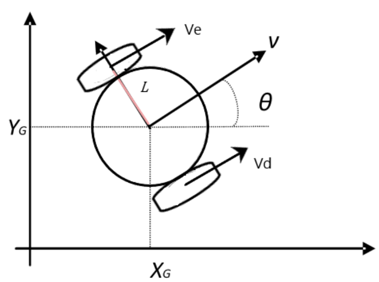


Fig. 1: A differential robot diagram [3].

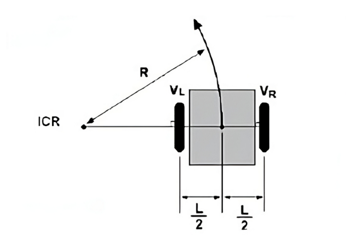


Fig. 2: A differential robot diagram that shows the movement's robot behavior [3].

$$\omega_r = \frac{V_R - V_L}{2} \tag{1}$$

$$v_r = \frac{V_R + V_L}{2} \tag{2}$$

$$R = \frac{L}{2} \cdot \frac{\Delta S_R + \Delta S_L}{|\Delta S_R - \Delta S_L|} \tag{3}$$

$$V_{max} = \sqrt{R \cdot g \cdot \mu_s} \tag{4}$$

B. Mapping

The radius of each curve is essential to know each curve's maximum linear velocity. The robot collects this information before putting it on run mode. To facilitate the microcontroller, code is used as a resource for on-track competition. According to Figure 3, two types of white line marks are on track. One type indicates the track's beginning and end, using two marks. The other one indicates the beginning and the ending of a curve. For this approach, the curve marks were used to delimit the track in segments of straight lines and curves. While the robot travels the track, the microcontroller keeps the information on the displacement of each wheel and saves it on its memory. The information collected is sent to a Bluetooth module when the robot completes the path. This information is used on a computer to calculate the radius of each curve and the maximum linear velocity in each track's segment. For example, the greater radius is 299.02 mm making it possible to reach 1.17 m/s of linear velocity.

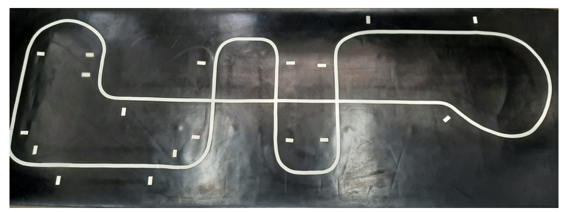


Fig. 3: Photo of the line follower robot track used for this research [3].

C. PID Ccontrol

This section will describe the implemented PID control concept to help the robot have stability and speed control. PID control is most used in the industry for process automation applications [11], [12]. This control consists of reading a reference of your system, that is, a value to be reached, and the reading of a sensor and, through the difference of these data, will return an error that will be used to stabilize the system. In the case of the PID control, a Proportional control is used, which generates a gain in the correction factor. A Derivative control causes a gain in the error variation, acting mainly in transient regimes. The Integrative control acts in the gain of the steady-state error to generate stabilization. The strategy adopted for the robot proposed in the research will be described in stability control and speed control.

Among the two types of control employed, the angular control is responsible for acting in the robot's direction, keeping it on the line. The system has different behaviors, whether on straight lines or curves. So, it is necessary to consider these situations during the development of angular PID control. The linear control is responsible for dictating the maximum speed that the robot can reach in each path. So, when changing from a straight line to a curve, the idea is for this control to make a controlled deceleration and otherwise generate a controlled acceleration. More details about the system diagram of a line follower robot with two controllers can be seen in [3]

D. Tunning Methodology

There are different methods for tuning the PID control. One is by using the Zigler-Nichols reaction curve tuning method.

This method analyzes an engine's reaction curve, for example, from which the closest constants for the control are found at Ogata [13].

The competitive line-follower robot must maintain a certain degree of stability and maximum speed. With that in mind, the control was always placed under-damped so that even if there were a small oscillation in the transient, the transient time would be small. And even without reaching the steady state, the robot would stabilize faster than if it were a critically damped control. However, achieving optimal system tuning mandates a clear understanding of the system's plant. In the subsequent subsection, we will delve into the methods for acquiring the plant of a Line Follower robot.

1) Line Follower Plant: To obtain the plant, the motor's curve was analyzed. For this purpose, the encoder sensors were used to obtain the motor's speed in each PWM set by the microcontroller. In Figure 4, it's possible to note the curve of the motor. Notably, that is a first-order system, making obtaining the plant easier. Using the graph as a basis, it is possible to know that the system's settling time is 100 ms. It is known that a first-order system is given by Equation 5, where k is the constant of the system and b is the system's pole. It is possible to obtain b with a known settling time. The b term is four divided by the settling time in seconds, which results in a b of 40ms. To find the constant k, calculate the transfer function at frequency s = 0. Looking at the graph, the value of G(s) at steady state is 290 rad/s, so the constant k is 11600. So the system's transfer function can be written according to Equation 6.

$$G(s) = \frac{k}{s+b} \tag{5}$$

$$G(s) = \frac{11600}{s+40} \tag{6}$$

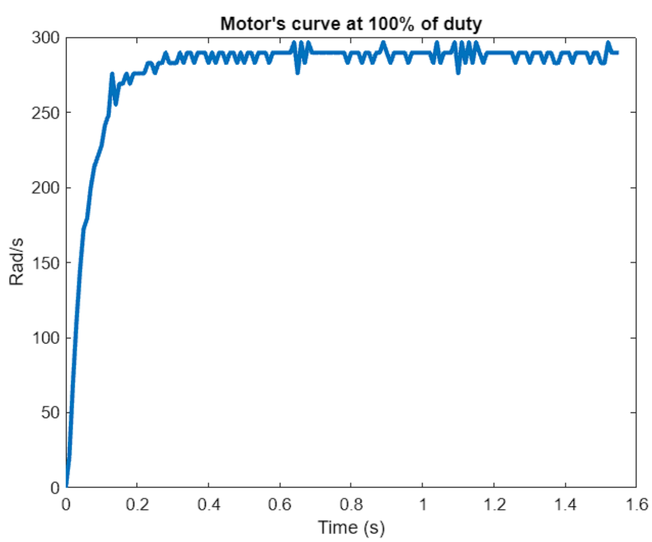


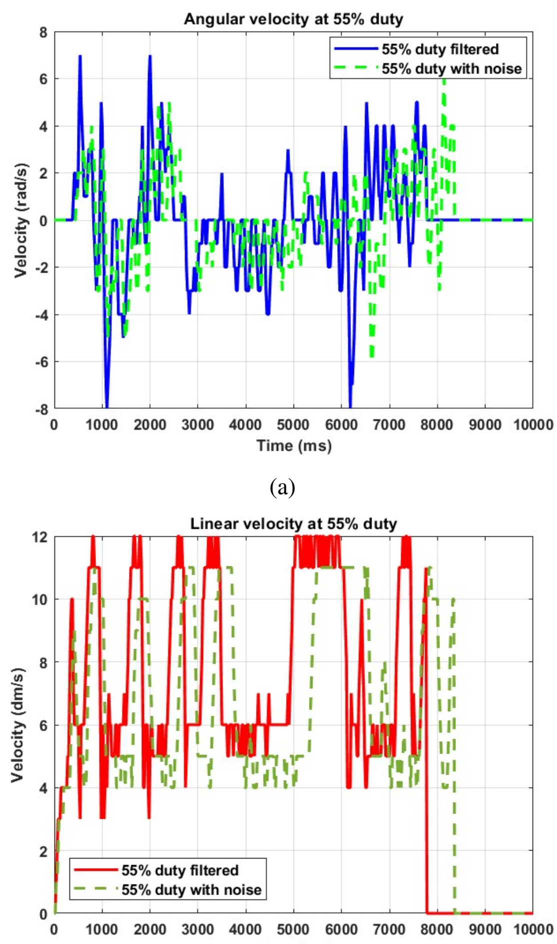
Fig. 4: Motor's curve at 100% of duty cycle.

According to Figure 4, it is possible to know the limit of the motor response time. Besides, it is also possible to perform motor tuning. However, it is not possible to do the robot tuning because the system plan is still missing. Then, tuning is done through graphical analysis.

So far, the analysis has been performed through graphical comparisons in the system response. A telemetry of the robot's behavior along the path is performed, then the behavior is examined graphically. The derivative is increased by observing the oscillations and the delay for the system to stabilize. If the system cannot reach an approximate value of the reference, which would be insufficient speed to make a curve, the proportional controller is increased. The graph used can be seen in Figure 5. In this image, it can be seen that the robot oscillates a lot both in linear and angular speed, straight and curved paths. The constants used are presented in Table I. It is worth mentioning that this tune was the first tune found that the robot did the track entirely with 55% of the duty cycle. It would be necessary to increase the system derivative to reduce these oscillations.

TABLE I: PID Values for Angular and Linear Control.

Type	P	I	D
Angular	13	0	0.09
Linear	50	0	0.000



(a)

(b)

Fig. 5: Angular and linear velocities with 55% duty cycle.

III. Development

A. Robot's hardware

For the proper functioning of the robot, it is necessary to use the appropriate components to meet the purpose for which it will be designed. As seen in Figure 6, it is possible to observe the components used in this robot. Next, all the components used will be described in detail.

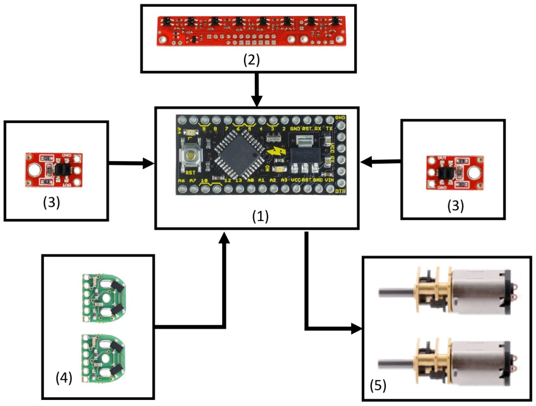


Fig. 6: Electric diagram of the line follower embedded system. (1) Microcontroller Arduino Pro mini. (2) Array of 8 IR sensors. (3) Lateral Sensors. (4) Encoders for micro metal gear motors. (5) Micro metal gear motors. From: [3]

1) Motor driver: In specific applications in electronics, when it is necessary to control the direction of rotation of electric motors, in this case, a DC motor, it is necessary to invert the direction in which it is being fed so that we have two directions of current flow. However, this action requires a control logic to be done correctly. In the case of a line-following robot, a microcontroller from [14] was used. In this case study, a TB6612FNG H bridge is used.

From this circuit, it is possible to power the motors through pins AO1 and AO2 as well as BO1 and BO2, and using any microcontroller to control the direction of rotation through pins AIN1 and AIN2 as well as pins BIN1 and BIN2. Another important resource of this circuit is adjusting the voltage value delivered to the load (i.e., DC motors). This makes it possible to determine different speeds and torques for the robot through the motors according to the need. This voltage control uses two distinct PWM signals generated by the microcontroller. They are inserted in the PWMA and PWMB pins, and with them, the voltage value that will be delivered in the output pins destined for the motors is precisely controlled.

2) Infrared sensors: The robot's orientation is controlled through a sequence of processes involving the microcontroller, facilitated by utilizing infrared sensors, specifically the integrated circuit QRE1113, to gather crucial information. Notably, multiple sensors are positioned at the robot's front to enhance positioning accuracy on the track, with the implementation employing seven sensors. The corresponding circuitry is outlined in the depicted figure. The infrared sensor interfaces with a pull-up resistor, establishing a voltage divider circuit. With a voltage applied at VIN (for instance, VIN = 5V), its output, denoted as OUT, takes on intermediary values between 0V and 5V based on the surface's reflectance. When encountering maximum reflectance (white), the OUT value registers 0V; conversely, in the absence of reflectance (black), the OUT value corresponds to 5V.

As illustrated in Figure 3, lateral sensors are responsible for reading markers around the track. The sensor on the robot's right reads stop markers, while the one on the left reads markers referring to the beginning and end of curves.

3) Encoders: Two encoder sensors are used. These include two dual-channel Hall Effect Sensor Boards models: *(i)* TLE4946-2K; and *(ii)* two 6-pole magnetic disks that can be used to add quadrature coding to motors. The magnetic disks emit several pulses with each motor shaft rotation. Knowing the diameter of the wheel, it is possible to determine the distance traveled by the robot precisely.

4) Bluetooth module: To collect data from the system's behavior, the HC-05 module was selected, which allows a wireless Bluetooth connection with devices, simplifying data storage.

5) Microcontroller: The microcontrollers consist of an integrated circuit, where in its encapsulation, there is a processor, memories, and peripherals such as an AD converter and Timer, in addition to a set of inputs and outputs (I/O) that make it possible to execute tasks, within their limits, in a compact form through a single chip [15]. This research used the microcontroller ATmega328p on the Arduino Pro mini board.

6) DC Motors: The motors chosen to generate movement for the robot were DC motors with a 10:1 reduction box with 3000 RPM, which allows the necessary torque for the movements. Its power supply was designed for 6V and is fed through the H bridge model TB6612FNG channels.

IV. Robot Improvements

A. Aerodynamics improvement

The current robot has 158 grams, while the previous one has 188 grams. In terms of aerodynamics, it is possible to observe that the new robot has more excellent stability because its design has a kind of wing that helps against the effect of air resistance. The previous robot performed the test track in 12 seconds for comparison purposes. In comparison, the current robot does the complete track, in its worst time in 9 seconds, and in its best time, in approximately 7.4 seconds. The new projected robot is given in Figure 7 and the previous version in Figure 8.

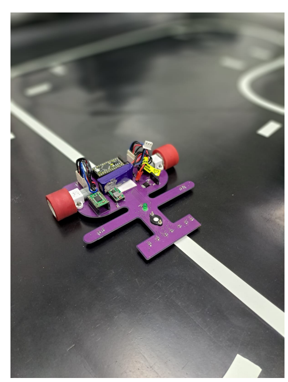


Fig. 7: Line follower robot called Micros.

B. Reading sensors with minimum noise

Although better designed, the electronic sensors are subject to noise. To mitigate this issue, several ways exist to treat sensor reading noise, either through low-pass circuits or digital filters [16]. In this study, the implementation of digital filters was used to observe the system's behavior. In the application

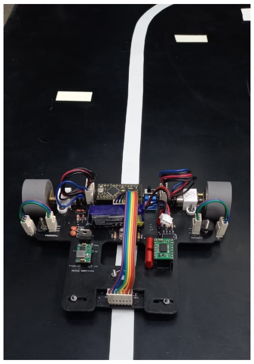


Fig. 8: The authors developed the previous line follower robot, called Millis [3]

of a line follower robot, the hardware has two sensors: *(i)* infrared sensors to capture the line; and *(ii)* encoder sensors to measure the speed of the motors. Initially, a low-pass digital filter was implemented to reduce the inherent noise of the microcontroller reading. The filter used was a moving average of 20 samples, with the sampling rate set at 2 ms. The moving average can be seen in Equation 7.

$$MA_k = \frac{1}{k} \cdot \sum_{i=n-k+1}^{n} pi \tag{7}$$

, where k is the number of samples and pi, is the sample at the time i.

A moving average consists of making an average of a set of samples and always shifting data to the right. After the average, the first sample is discarded, and the twenty-first becomes the last sample. However, what should be highlighted in this, and any filter, is that every filter needs several samples, so the filtered signal will always be delayed from the previous signal. In the case of this robot, the number of 20 samples is enough because, in addition to filtering the signal well, the rate for collecting all samples is 40 ms. This is enough for the system to react. Figures 9 and 10 compare the reading speeds of systems with and without filters.

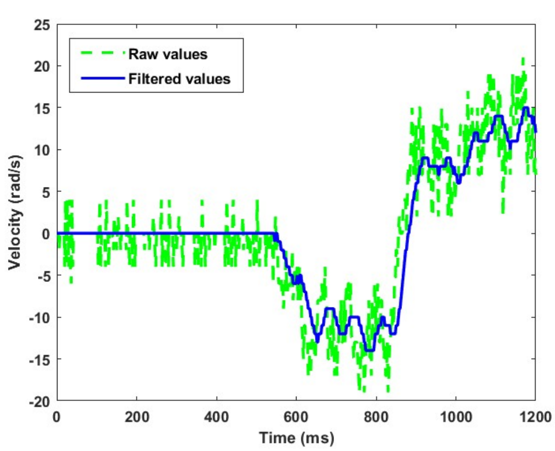


Fig. 9: Comparison between raw values and filtered values for angular velocity.

It is possible to observe the action of the filter concerning significant variations in the sensor reading and the filtered

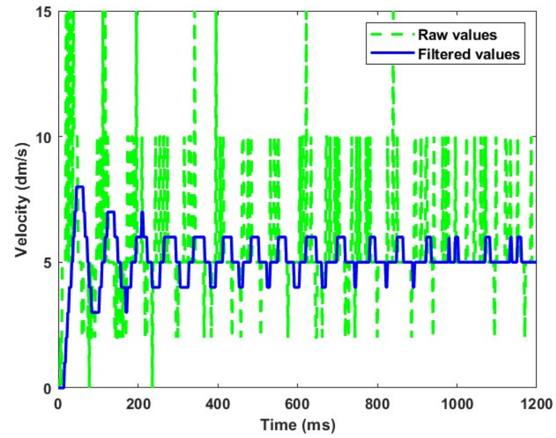


Fig. 10: Comparison between raw values and filtered values for linear velocity.

signal. The filtered signal maintains an average of the signal, ignoring peak values. It is worth mentioning that there is a slight delay in relation to the original signal due to the dependence on a specific number of samples. If more samples are needed, there will be a considerable signal delay.

C. Implementation of Derivative Filter

Note that the derivative portion of the PID control is subject to noise due to error variation, especially at high frequencies. Thus, an alternative is implementing a digital filter, as seen in Figure 11. There are several ways to implement a derivative filter, either by using a feedback loop with an integrator, as seen in the image, or by using a first-order low-pass filter in the derivative calculation. In the study carried out, the second method was used using Equation 5. The outcomes are shown in Figure 12. By using the derivative filter, it can be observed that the robot demonstrated fewer oscillations and a faster transient response. For this approach, a moving average of 20 exclusively was used to keep the corporate rate consistent with the moving average of the speed sensors. It can be seen in Figure 12(a) that the system without a digital filter took around 9.5 s to complete the track, while the system with a digital filter (Figure 12(b)) took around 8.8 s. Having a reduction of 7.36% of the lap time.

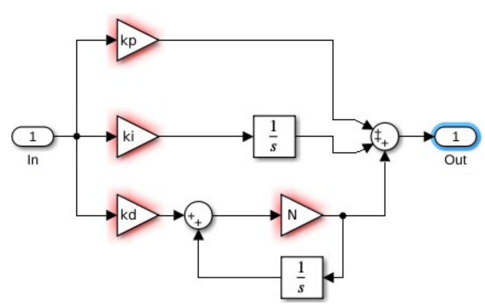
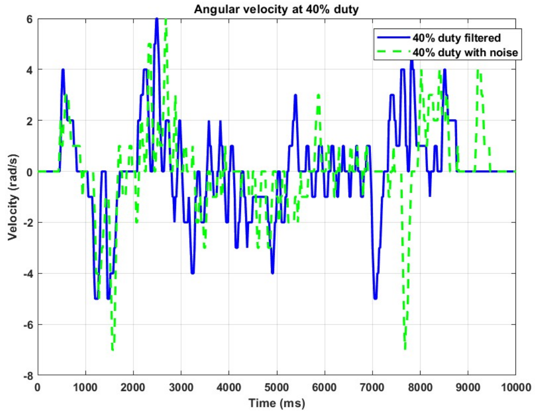


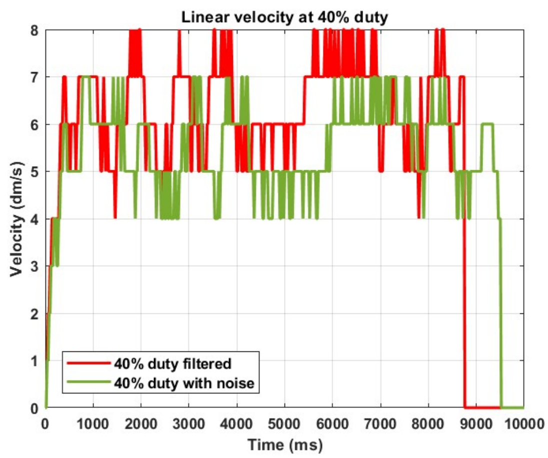
Fig. 11: PID block diagram with a derivative filter.

V. Results and Discussion

Note that the PD-type control brought better performance than the PID control for this type of robot and the competition in which it is applied. This approach made it possible to



(a)



(b)

Fig. 12: Angular and linear velocities with 40% duty cycle. Both graphics show results with digital filters and with noise.

achieve the 7.4 seconds range on an 8-meter-long track. From the graphical analysis, it was also possible to identify which constant should be changed. Despite not being the best tuning methodology, it yielded satisfactory results for the study. Figure 12 compares the robot performance on track at 40% duty. The difference between the reading of the sensors using a digital filter and without a filter is noticeable, as in Figures 10 and 9. It is worth mentioning that it is possible to make these readings more stable by increasing the number of samples. However, it is necessary to recalculate the system sampling rate. In addition, it is possible to observe that this practice made the robot maintain the linear control reference for a longer time and more excellent stability in the angular control, consequently making the robot finish the track in less time using digital filters in the code. All code and a video of the robot can be found at **GitHub**

VI. Conclusions and Future Work

This work proposed the development of a competitive line follower robot, describing the hardware, the control, and the control analysis. In this research, it was possible to observe the essential use of digital or analog filters to read sensors, mitigating possible reading noise. This practice brought a noticeable difference in the stability of the system. Furthermore, tuning is fundamental when stabilizing the system in different situations where its reference reaches different or more oscillatory values. Besides, it is essential to highlight that using anti-wind-up strategies and band-limited derivatives can greatly improve the discrete implementation of PID. In this sense, the research intends to implement an anti-wind-up and the desired tuning strategy for future works. Besides, it is also planned to compare with the currently used methodology more dynamically, using displacement data and acceleration of the system. Besides, the authors also intend to implement a recognition path methodology so that the robot memorizes track data to optimize the route.

Acknowledgment

The authors would like to thank CEFET/RJ and the Brazilian research agencies CAPES, CNPq, and FAPERJ for supporting this work. Besides, the authors are grateful to the Foundation for Science and Technology (FCT, Portugal) for financial support through national funds FCT/MCTES (PIDDAC) to CeDRI (UIDB/05757/2020 and UIDP/05757/2020) and SusTEC (LA/P/0007/2021).

References

[1] S. Tayal, H. P. G. Rao, S. Bhardwaj, and H. Aggarwal, "Line follower robot: Design and hardware application," in 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), pp. 10–13, IEEE, 2020.

[2] H. Murcia, J. D. Valenciano, and Y. Tapiero, "Development of a line-follower robot for robotic competition purposes," in Workshop on Engineering Applications, pp. 464–474, Springer, 2018.

[3] J. S. J. C. C. Amorim, J. H. O. Fernandes, M. S. Machado, A. L. C. Canella, and M. F. Pinto, "Design of a control approach to assist the performance of a competitive line follower robot," in 2022 Latin American Robotics Symposium (LARS), 2022 Brazilian Symposium on Robotics (SBR), and 2022 Workshop on Robotics in Education (WRE), pp. 354–359, 2022.

[4] "Line follower robot regulation," 2016.

[5] E. Serrano Pérez and F. Juárez López, "An ultra-low cost line follower robot as educational tool for teaching programming and circuit's foundations," Computer Applications in Engineering Education, vol. 27, no. 2, pp. 288–302, 2019.

[6] D. V. Babu, D. Jenniffer, and R. Karthikeyan, "Line follower robot & obstacle detection using pid controller," in AIP Conference Proceedings, vol. 2271, p. 030005, AIP Publishing LLC, 2020.

[7] M. Nugraha, P. R. Ardianto, and D. Darlis, "Design and implementation of rfid line-follower robot system with color detection capability using fuzzy logic," in 2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), pp. 75–78, IEEE, 2015.

[8] M. Engin and D. Engin, "Path planning of line follower robot," in 2012 5th European DSP Education and Research Conference (EDERC), pp. 1–5, IEEE, 2012.

[9] M. Gomes, L. Bassora, O. Morandin, and K. C. T. Vivaldini, "Pid control applied on a line-follower agv using a rgb camera," in 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 194–198, IEEE, 2016.

[10] "Representation of kinematic models," 2022.

[11] S. F. Jibrail and R. Maharana, PID Control of Line Followers. PhD thesis, 2013.

[12] L. A. Silva, I. Barroso, A. Menezes, A. R. Zachi, M. F. Pinto, and A. G. Melo, "Development of a control systems training module for application on undergraduate engineering classes," in Congresso Brasileiro de Automática-CBA, vol. 2, 2020.

[13] K. Ogata, Engenharia de controle moderno. Pearson Prentice Hall, 2011.

[14] "Mundo Projetado ponte h - o que é e como funciona?." https://mundoprojetado.com.br/ponte-h-o-que-e-e-como-funciona. Accessed: 2022-07-15.

[15] "IEEE RAS UFCG o que é um microcontrolador?." https://edu.ieee.org/br-ufcgras/o-que-e-um-microcontrolador. Accessed: 2022-09-30.

[16] M. M. d. O. Carneiro, M. F. Pinto, A. G. Manhães, and U. d. F. da Graça, "Performance analysis of digital filters employed to plants electrical signals," in Simpósio Brasileiro de Automação Inteligente-SBAI, vol. 1, 2021.