Analysis and tuning of PID controller gains for DC servo drives using Garpinger's trade-off plots*

Simon Hoher[1] and Jakob Rehrl[2]

***Abstract*— Although PID tuning for DC drives is widely studied, a structured, practical guide addressing robustness and setpoint tracking/disturbance rejection trade-offs is still lacking. This paper condenses established methods into a clear, step-by-step approach for optimal PID tuning using Garpinger's trade-off plots, aiming at practical use in industrial applications.**

***Index Terms*— PID control, Garpinger's trade-off plots, AMIGO and Garpinger method, servo drives**

I. Introduction

Precise PID tuning is essential for electric drive control in industrial settings, requiring fast disturbance rejection, setpoint tracking, and robustness. This paper presents the Approximate M-constrained Integral Gain Optimization (AMIGO) [7] and the Garpinger method [4] as structured tuning approaches addressing these needs. AMIGO ensures fast, robust control without overshoot; Garpinger adds flexibility by allowing gain adjustment without compromising performance. Though focused on DC motors due to their modeling simplicity, results apply to synchronous motors via PQ-transformation [6], which are standard in industry. Using Garpinger trade-off plots, we show that optimal gains can be selected directly, offering an efficient and practical tuning method suitable for broader adoption.

II. Related Work

A. PID Control: Basics and Challenges

PID control is widely used due to its simplicity and robustness [3]. The controller output is defined as

$$u(t) = K_\mathrm{P} \cdot e(t) + K_\mathrm{I} \cdot \int_0^t e(\tau)\,\mathrm{d}\tau + K_\mathrm{D} \cdot \frac{\mathrm{d}e(t)}{\mathrm{dt}},$$

where $u(t)$ is the controller's output, $e(t)$ the control error, K_P the proportional gain, K_I the integral gain and K_D the derivative gain.

Tuning the gains K_P, K_I, K_D is nontrivial, as it must ensure:

- Stability of the closed loop system,
- Fast response to setpoint changes and disturbances,
- Minimal overshoot,
- Robustness to model uncertainties.

In practice, cascaded control structures are often used to enhance performance (see Figure 1).

* The financial support of the Christian Doppler Research Society and the associated company partners of JRC ISIA is gratefully acknowledged

[1]Salzburg University of Applied Sciences, JR Center for Intelligent and Secure Industrial Automation (JRC ISIA) simon.hoher@fh-salzburg.ac.at

[2]Salzburg University of Applied Sciences, JR Center for Intelligent and Secure Industrial Automation (JRC ISIA) jakob.rehrl@fh-salzburg.ac.at

B. Cascaded Control

Cascaded control is widely used in motor control, especially for DC and synchronous motors [2]. It consists of nested loops (see Figure 1):

1. **Inner Control Loop**: The inner control loop regulates the motor's speed and quickly responds to load changes.
2. **Outer Control Loop**: The outer control loop handles position control. The primary goal of the outer loop is to maintain accurate position control and ensure closed-loop stability.

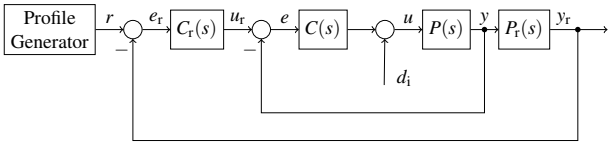

Fig. 1. Cascaded control with upstream profile generator

Typically, a trapezoidal velocity profile is used to provide the motor movement in three phases [3] (see Figure 2):

1. **Acceleration Phase**: The motor accelerates with a constant maximum acceleration a_max to the maximum speed v_max.
2. **Constant Speed Phase**: After reaching the maximum speed, the motor continues to move at constant speed v_max.
3. **Deceleration Phase**: The motor decelerates with the same maximum acceleration $-a_\mathrm{max}$ to precisely reach the final position.

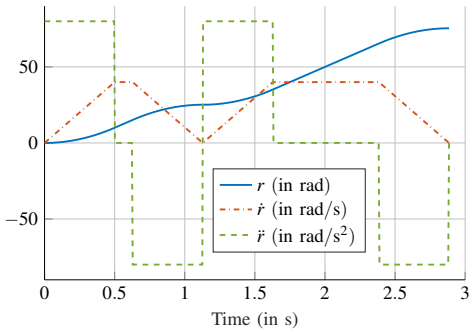

Fig. 2. Calculated trajectory with a trapezoidal velocity profile for given waypoints 4 and 12 revolutions ($v_\mathrm{max} = 40$ rad/s, $a_\mathrm{max} = 80$ rad/s^2)

C. Garpinger's trade off diagrams

The design of PID controllers often involves balancing competing objectives, such as minimizing control error and

https://doi.org/10.34749/3061-0710.2025.3

ensuring robustness against disturbances and model uncertainties. The Garpinger trade-off plots provide a powerful visualization of these trade-offs by representing performance and robustness criteria explicitly [4]. These plots help to identify optimal PID parameters by highlighting the compromise between error minimization and robustness.
Three key criteria are typically used in these analyses:

Performance Criteria: IE and IAE

The *IE* criterion measures the integral of the control error e, i.e., $IE = \int_0^\infty e(t)\,\mathrm{dt}$. This metric captures the overall magnitude of the error but does not emphasize short-term or large deviations.

The *IAE* criterion improves upon the *IE* by emphasizing absolute deviations, which are often more relevant in practical systems,

$$IAE = \int_0^\infty |e(t)|\,\mathrm{dt}. \tag{1}$$

The *IAE* is widely used as a performance metric because it penalizes persistent deviations more effectively than the *IE*. A lower *IAE* indicates better performance in terms of setpoint tracking and disturbance rejection. If $|IE|$ and *IAE* yield identical values, no overshoot occurs in the system.

The computation of the $I(A)E$ values is typically done for two experiments: i) a step response from d_i to y (disturbance rejection), and ii) a step response from u_r to y (setpoint tracking) in Figure 1.

Robustness Criterion: Maximum Sensitivity M_st

The robustness of a control system is commonly evaluated using the maximum sensitivity criterion, defined as:

$$M_\mathrm{st} = \max_\omega \left(|S(\mathrm{j}\omega)|, |T(\mathrm{j}\omega)|\right) \tag{2}$$

$S(\mathrm{j}\omega)$ is the sensitivity function, representing the system's response to disturbances and model uncertainties at different frequencies. $T(\mathrm{j}\omega)$ is the complementary sensitivity function and represents the closed-loop frequency response for setpoint tracking, describing how the output y responds to changes in the setpoint r. A lower M_st corresponds to a more robust system that tolerates model variations better. A higher M_st suggests the system is less robust, as uncertainties are amplified more significantly.

The Trade-Offs in Garpinger Plots

The Garpinger trade-off plots visualize the interplay between performance (*IAE*) and robustness (M_st) (see Figure 3).
Unstable K_P-K_I parameterizations are colored grey (by the term unstable, a closed-loop system that is not internally stable [9] is ment). The M_st is plotted as red line, and the *IAE* as blue line. The *IE* value can be calculated by reading the controller gain on the ordinate axis: $IE = -1/K_\mathrm{I}$. Where the $|IE|$ value coincides with the horizontal line of the *IAE* value, $|IE|$ and *IAE* have the same value. The *IAE* value does not change along the blue lines. Each point corresponds to a specific set of PI controller gains.
The optimal line (green in Figure 3), or Pareto front, is the set of points where no further improvement in one criterion can be achieved without degrading the other and is plotted as green line. Designers can choose parameters along this (green) line depending on the specific application requirements.

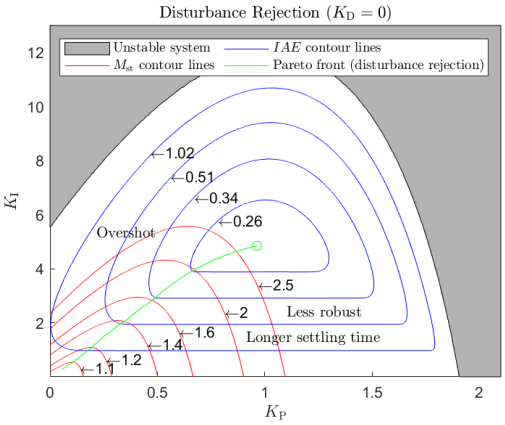

Fig. 3. Grapinger's trade off plot

Control gains to the left of the Pareto front lead to an overshoot (since $|IE| \neq IAE$), to the right of the Pareto front to a less robust system (since *IAE* is constant but M_st increases) and longer settling time (since *IAE* is constant but proportional controller K_P gain is increasing). By using these plots, it is possible to systematically select PID parameters that balance performance and robustness in a way that aligns with the specific needs of the control system. This approach not only improves system reliability but also provides a clear methodology for achieving optimal tuning.
The plots show how reducing the $|IE|$ or *IAE* (better performance) often comes at the expense of increased M_st (reduced robustness). Lowering the *IAE* typically requires higher PID gains, which may improve disturbance rejection or setpoint tracking but also makes the system more sensitive to noise and model uncertainties. Reducing M_st enhances robustness but may result in slower responses and larger errors. In the Garpinger's trade-off plots, an optimal line emerges, representing the best compromise between performance and robustness (see green line in Figure 3). *Remark:* This optimal line yields different sets of controller parameters when considering either disturbance rejection or setpoint tracking trade-off plots. In the remainder of the paper, the type of plot that is used is always mentioned. In some of the Garpinger plots, two lines describing the optimal controller parameters are shown. The green one is related to the disturbance rejection, whereas the magenta one is related to setpoint tracking.

D. AMIGO and Garpinger Method for PID Controller Tuning

The AMIGO method is a modern approach for tuning PID controllers, designed to overcome the limitations of traditional tuning methods such as Ziegler-Nichols method [7].

https://doi.org/10.34749/3061-0710.2025.3

The method incorporates advanced design criteria to optimize disturbance rejection and minimize overshoot, while ensuring high robustness against model uncertainties and process variations.

The AMIGO method calculates the PID controller gains (K_P, K_I, K_D) based on the step response of the system.

1. **Process Gain** (K): The steady-state gain of the system, calculated as: $K = \Delta y/\Delta u$ where Δy is the change in the output and Δu is the change in the input of the plant.

2. **Time Delay** (L): The time it takes for the output to begin responding significantly after the input step.

3. **Time Constant** (T): The time required for the system to reach approximately 63 % of its steady-state response, minus the time delay.

Based on the above parameters, the AMIGO method calculates the PI gains as follows:

$$K_P = \frac{0.15}{K} + (0.35 - \frac{L \cdot T}{(L+T)^2}) \cdot \frac{T}{K \cdot L}, \tag{3}$$

$$T_I = 0.35L + \frac{13L \cdot T^2}{T^2 + 12L \cdot T + 7L^2}, \tag{4}$$

$$K_I = \frac{K_P}{T_I}. \tag{5}$$

If the D term is also to be considered, then there are also analogous formulas that interpret the controller gains somewhat more conservatively [1].

In certain cases, process constraints require adjustments to the controller gain. The Garpinger method addresses this by calculating the optimal integral gain (K_I) as a function of the proportional gain (K_P) based on parameter fitting derived from Garpinger's trade-off plots [4]

$$K_I = \frac{K_P + 0.1K \cdot K_P^2}{0.3L + 0.7T}, \tag{6}$$

and is valid for $M_{st} < 1.6$.

III. Research Question and Approach

This research explores and compares the performance and robustness of two modern tuning methods for PID controllers in motor control: the AMIGO method and the Garpinger method. Both methods are evaluated in cascaded control systems for speed and position control, using Garpinger's Trade-Off Plots to balance performance metrics (e.g., fast disturbance rejection, precise position tracking) with robustness criteria (e.g., maximum sensitivity).

The evaluation was conducted on an Arduino-based DC motor system, replicating realistic operating conditions and noise to test controller robustness. The controller parameters were validated by observation of the response times of the speed and position.

IV. Case Study

This section presents the design of a velocity control loop based on feedback control (IV-A to IV-C) and the implementation of a position control loop (IV-D).

A. System Identification and Model Extraction

To capture the behavior of the real system, an open-loop step response was performed using a DC motor from the Makeblock mBot Ranger kit. The motor is driven by a PWM signal ranging from -255 to +255 (12 V max). The measured outputs are angular velocity y (in rpm) and angular position y_r (in radians), obtained via the onboard encoder. The controller operates at a cycle time of 5 ms.

The step response (Figure 4) reveals significant noise in the speed signal, whereas the position response is relatively smooth.

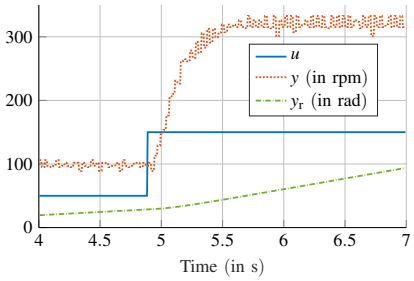

Fig. 4. Step response of DC motor

To address this, a first-order low-pass filter with a time constant $T = 0.05$ s was applied, chosen to be about four times faster than the system's natural cutoff. This reduces noise while preserving essential dynamics (see Figure 5).

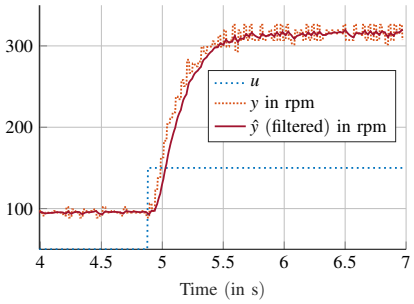

Fig. 5. Step response of DC motor with a filtered angular velocity with filter time constant $T = 0.05$s

The filtered system is modeled as

$$P(s) = \frac{\hat{Y}(s)}{U(s)} = \frac{K}{T \cdot s + 1} \cdot e^{-L \cdot s} = \frac{2.222}{0.198 \cdot s + 1} \cdot \mathrm{e}^{-0.087 \cdot s}. \tag{7}$$

This first-order lag plus time delay (FOLPD) model sufficiently captures the motor dynamics (see Figure 6) and serves as the basis for controller design.

The maximum velocity v_{max} and acceleration a_{max}, needed for trajectory generation, are estimated directly from steady-state K and maximum input u as

$$v_{max} \approx K \cdot u_{max} \cdot \frac{2\pi}{60} = 2.222 \cdot 255 \cdot \frac{2\pi}{60} \approx 50\,\mathrm{rad/s}, \tag{8}$$

$$a_{max} \approx 2 \cdot u_{max} \cdot \frac{K}{T} \cdot \frac{2\pi}{60} \approx 600\,\mathrm{rad/s^2}. \tag{9}$$

https://doi.org/10.34749/3061-0710.2025.3

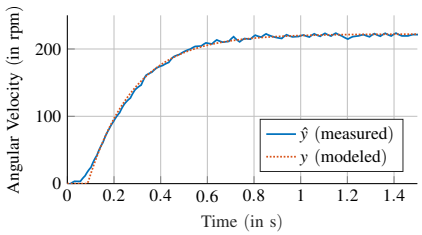

Fig. 6. Comparison of the real measurement data with the identified FTOD model

B. Garpinger's Trade-Off Plots for Performance and Robustness

To evaluate the controller design, trade-off plots based on the Garpinger method were created in MATLAB for both disturbance rejection (Figure 7) and and setpoint tracking (Figure 8). These plots visualize the trade-off between performance (measured by *IAE*) and robustness (measured by maximum sensitivity M_{st}).

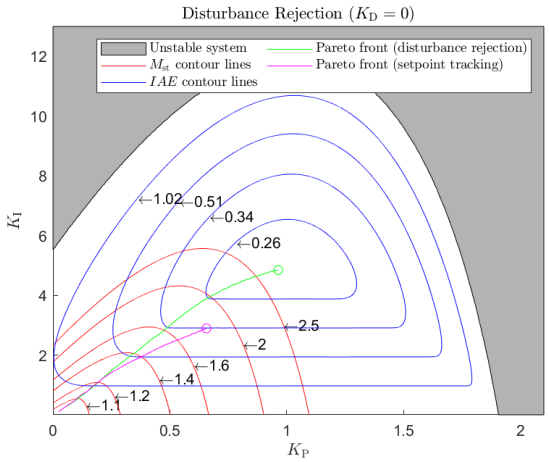

Fig. 7. Garpinger's trade-off plot for disturbance rejection

1. **Pareto Front Analysis**:
The green and magenta lines represent Pareto fronts for disturbance rejection and setpoint tracking, respectively. Each point on the front offers the best achievable performance for a given robustness level. For $M_{\text{st}} < 1.2$, the fronts diverge, revealing that both objectives cannot be optimized simultaneously.

2. **Choice of Robustness Level**:
A moderate robustness level of $M_{\text{st}} = 1.4$ is selected, as this value provides a reasonable balance between sensitivity to disturbances and robustness against uncertainties. For this M_{st} value, the controller gains can then be read off the Pareto Front for disturbance rejection (green line) at $K_{\text{P}} \approx 0.34$ and $K_{\text{I}} \approx 2.07$ (see Figure 7) and $K_{\text{P}} \approx 0.38$ and $K_{\text{I}} \approx 1.95$ for setpoint tracking (see Figure 8).

3. **Impact of Optimization Choice**:
Tuning for disturbance rejection leads to better rejection performance but results in overshoot during setpoint changes.

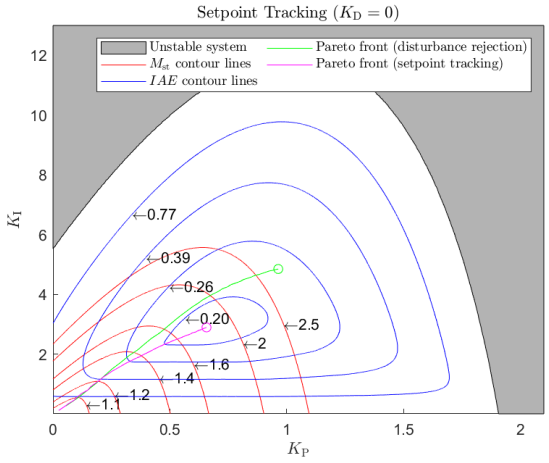

Fig. 8. Garpinger's trade-off plot for setpoint tracking

Conversely, tuning for setpoint tracking sacrifices disturbance suppression. This reflects the inherent conflict between these objectives in PID control.

4. **Significance of the Integral Term (K_{I})**:
The results consistently demonstrate the importance of including an integral term ($K_{\text{I}} > 0$) in the controller design. While the literature often suggests that a P-P cascaded control may be sufficient for many applications [8], the trade-off plots show that an integral term significantly enhances both disturbance rejection and setpoint tracking. By implementing an *I*-term, the controller achieves superior overall performance compared to purely proportional control strategies. In addition, the disturbance error would not be eliminated with a pure P control, as the plant does not have a pure *I* component.

By analyzing these plots, designers can select the most suitable controller gains for their specific application, balancing the trade-offs between disturbance rejection, setpoint tracking, and robustness. Furthermore, the findings clearly underscore the practical benefits of including an integral term in the control design.

C. AMIGO and Garpinger's tuning rules

The controller gains were initially analyzed using the Garpinger's trade-off plot for disturbance rejection. We now calculate the gains using the AMIGO rule-of-thumb method (see equations (3), (4) and (5)) and the parameters L, T and K from our model (see equation (7)), which yielded specific values for $K_{\text{P}} \approx 0.21$, $T_{\text{I}} \approx 0.18$, and $K_{\text{I}} \approx 1.17$.

These values were then compared to the trade-off plot for disturbance rejection (see Figure 9). The results showed that the gains obtained from the AMIGO method lie closely on the Pareto front for a robustness level of $M_{\text{st}} < 1.4$. This demonstrates that the AMIGO method provides optimal controller gains for a fixed robustness criterion of $M_{\text{st}} < 1.4$, ensuring a balance between disturbance rejection and robustness.

However, the AMIGO method has a notable limitation: it does not allow for independent adjustment of the pro-

https://doi.org/10.34749/3061-0710.2025.3

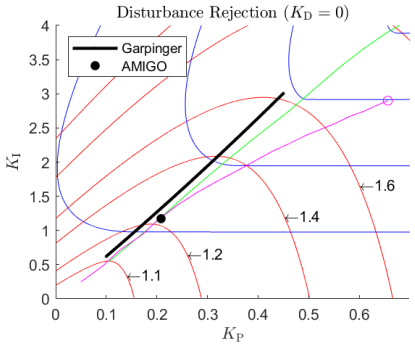

Fig. 9. AMIGO and Garpinger's rule-of-thumb highlited in Garpinger's trade-off plot for disturbance rejection

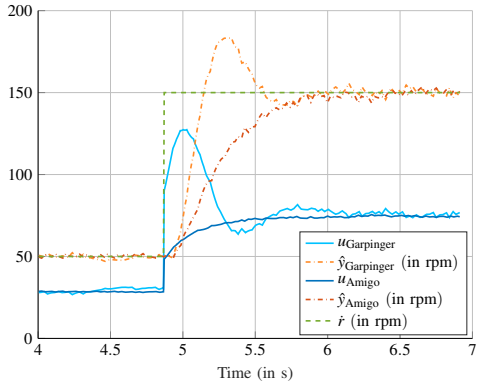

Fig. 10. Step response of motor with AMIGO and Garpinger's rule-of-thumb

portional gain K_P. To address this issue, the Garpinger method was applied. Different values below an M_{st} value of 1.6 were chosen manually for the proportional gain and the corresponding integral gain K_I is calculated using the Garpinger formula (6):

K_P	0.1	0.2	0.3	0.4	0.5
K_I	0.62	1.27	1.94	2.64	3.37

The new K_P-K_I pairs lie once again near the Pareto front of the Garpinger trade-off plot for disturbance rejection (see Figure 9). This result confirms that the Garpinger method not only accommodates adjustments to K_P but also ensures that the recalculated K_I maintains an optimal balance between performance and robustness for disturbance rejection. However, these pairs are located significantly to the left of the Pareto front for setpoint tracking (magenta line in Figure 9), particularly for higher K_P values. This indicates that controllers optimized for disturbance rejection are expected to exhibit considerable overshoot in response to setpoint changes. The results further emphasize the fundamental trade-off between disturbance rejection and setpoint tracking: the system can be optimized for one objective or the other, but not for both simultaneously. Consequently, the choice of controller parameters must carefully consider the specific performance priorities of the application, as optimizing for one criterion will inevitably compromise the other.

To validate the calculated controller parameters, the AMIGO and Garpinger methods (with $K_P = 0.4$ and $K_I \approx 2.64$) were tested on the real motor system. The controllers were implemented on the Arduino-based setup, and their performance was evaluated under practical conditions, focusing on setpoint tracking scenarios (see Figure 10).

The results revealed that the motor followed the desired velocity setpoint accurately, with a significant velocity overshoot for the Garpinger method. This behavior is consistent with the predictions from the trade-off plot for setpoint tracking, which shows that controller parameters optimized for disturbance rejection (with a robustness level of $M_{st} \leq 1.4$) tend to exhibit reduced performance in setpoint tracking. Specifically, the controller gains derived from the AMIGO and Garpinger methods prioritize disturbance rejection, which can lead to overshoot during rapid setpoint changes.

D. Cascaded control

Experimental results highlight how the cascaded control structure performs in response to a trapezoidal velocity profile (see Figure 11). A significant deviation was observed between the reference and actual response of the system and the target values could not be accurately reached. Instead, load disturbances are optimally compensated for.

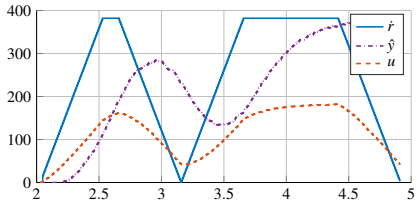

Fig. 11. Trapezoidal velocity profile response

In the final stage, a PID controller is implemented in the outer position loop. Unlike the inner loop, whose gains were determined via tuning rules, the outer loop gains are directly derived from the Garpinger trade-off plots to optimize tracking robustness.

The plant for the outer loop is defined as:

$$P_{out} = \frac{Y_r}{U_r} = \frac{PC}{1+PC} \cdot P_r, \tag{10}$$

where $P_r = 1/s$ represents the integrator that translates the velocity to a position signal.

A Garpinger trade-off plot is generated for setpoint tracking. Since the system has an integrating behavior, the plot visualizes the trade-off between the proportional gain (K_P) and the derivative gain (K_D), rather than the integral gain (K_I) used in the previous trade-off analyses. The disturbance still acts at the input of P and the noise filter was included in the calculation (compare equation (7)). The resulting Pareto front distinctly demonstrates the benefit of incorporating a

https://doi.org/10.34749/3061-0710.2025.3

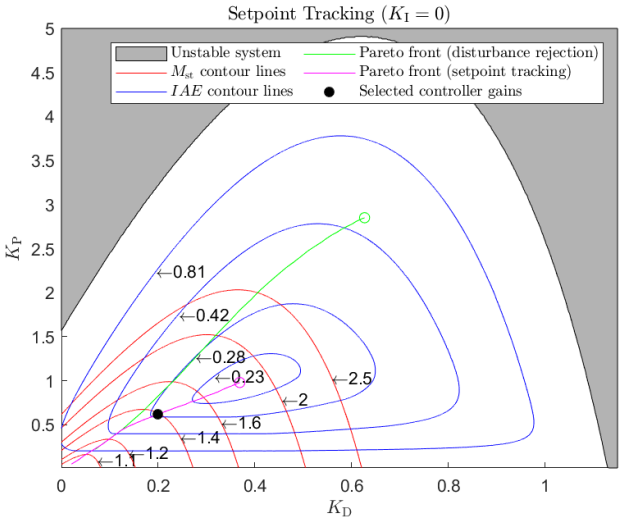

Fig. 12. Garpinger trade-off plot for setpoint tracking of the outer loop

derivative action to improve system performance (see Figure 12).

We again select an M_{st} value of 1.4 and now determine the controller gains from the trade-off plot with $K_P \approx 0.62$ and $K_D \approx 0.2$.

The complete control system is now validated using trapezoidal velocity trajectory tracking (see Figure 13). The system successfully follows the trapezoidal velocity profile, demonstrating that the controllers are properly tuned.

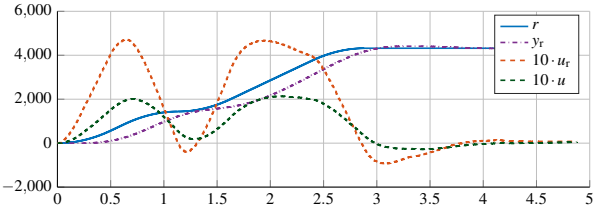

Fig. 13. Trapezoidal velocity profile response of the complete controlled system

The results confirm the importance of tuning the inner and outer loops differently to achieve optimal system performance:

- The **inner loop** should be primarily tuned for disturbance rejection, ensuring that speed fluctuations and external disturbances are suppressed effectively. The controller should have an I component for two reasons. Firstly, the trade-off plot shows that only then the smallest possible IAE value at a certain robustness requirement is met. Secondly, the I component in the inner loop is required to obtain zero steady-state control error.
- The **outer loop** should prioritize robustness and, if necessary, setpoint tracking, allowing the overall system to achieve smooth, accurate position control. However, since the setpoint tracking can also be achieved by feedforward control [5], particular attention should be paid to robustness.

By synthesizing controller gains directly from the Garpinger Trade-Off Plots, the system achieves a well-balanced trade-off between disturbance rejection, robustness, and setpoint tracking. The combination of optimized PID tuning, and low-pass filtering for noise reduction ensures that the cascaded control system performs with high precision in real-world applications.

V. Conclusion

The tuning of PID controllers for DC drives is well established, yet a structured step-by-step approach that systematically considers robustness, setpoint tracking, and disturbance rejection is still lacking. This paper addresses this gap by consolidating existing methods, particularly the AMIGO and Garpinger approaches, and systematically applying them to DC motor control. Garpinger's trade-off plots are utilized to facilitate the targeted selection of optimal controller parameters. The performance and robustness of the controllers are experimentally validated on an Arduino-based motor system, demonstrating enhanced setpoint tracking. Unlike the inner loop, which is tuned using rule-of-thumb methods, the outer loop of the cascaded control system is directly synthesized using the trade-off plots, ensuring an optimal balance between robustness and tracking performance. The results highlight the practical relevance of this methodology for industrial applications requiring high precision and reliability.

References

[1] K. J. Åström and T. Hägglund, *Advanced PID control*. ISA-The Instrumentation, Systems and Automation Society, 2006.

[2] N. Bacac, V. Slukic, M. Puškarić, B. Stih, E. Kamenar, and S. Zelenika, "Comparison of different dc motor positioning control algorithms," in *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2014, pp. 1654–1659.

[3] L. Desborough and R. Miller, "Increasing customer value of industrial control performance monitoring-honeywell's experience," in *AIChE symposium series*, no. 326. New York; American Institute of Chemical Engineers; 1998, 2002, pp. 169–189.

[4] O. Garpinger, T. Hägglund, and K. Åström, "Performance and robustness trade-offs in PID control," *Journal of Process Control*, vol. 24, 05 2014.

[5] P. Lino, J. Königsmarkova, and G. Maione, "Feedback-feedforward position and speed control of dc motors by fractional-order PI^v controllers," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2019, pp. 2584–2589.

[6] N. Mohamed, F. Aymen, B. H. Mouna, and S. Lassaad, "Modeling and simulation of vector control for a permanent magnet synchronous motor in electric vehicle," in *2021 4th International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, 2021, pp. 1–5.

[7] H. Panagopoulos, K. Åström, and T. Hägglund, "Design of PID controllers based on constrained optimisation," *Control Theory and Applications, IEE Proceedings* -, vol. 149, pp. 32 – 40, 02 2002.

[8] D.-L. Sabău, "Advanced control techniques for cnc machines," Ph.D. dissertation, Technical University of Cluj-Napoca, 2022.

[9] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control - Analysis and design*. John Wiley & Sons, 2001.

https://doi.org/10.34749/3061-0710.2025.3