

Longitudinal control for an autonomous vehicle using modified particle swarm optimization method

Ghaidaa Hadi Salih Elias ^{1*}

College of Computer Science and Information Technology, Kerbala University, Kerbala, Iraq

*Corresponding author E-mail: gadahadi6@gmail.com

Abstract

The control system is a very important part of the efficiency and safety of autonomous vehicles. This paper proposes the use of Particle Swarm Optimization (PSO) and Modified PSO (MPSO) algorithms to optimize Proportional Integral Derivative (PID) controller coefficients for a longitudinal dynamics vehicle system. The objective is to enhance system performance, measured by metrics such as maximum Overshoot (OS), Steady-State Error (SSE), Settling Time (t_s) (2%), and Rise Time (t_r). The MPSO algorithm, when combined with the PID controller, demonstrates a 2.5% improvement over the traditional PSO algorithm. The study contributes by showcasing the effectiveness of MPSO in fine-tuning PID controllers for superior control of longitudinal vehicle dynamics, as evidenced by the optimized response specifications.

Keywords: Modified Particle Swarm Optimization Algorithm; Longitudinal Dynamics Vehicle; PSO Algorithm; Cost Function.

1. Introduction

Recent years have seen a lot of research and development into autonomous vehicle technology because of all the benefits it provides over traditional automobiles [1]. It reduces driver errors, which cause car collisions, and it has the power to completely change how people move around by improving the convenience, accessibility, efficiency, and safety of car travel [2] [3]. One of the essential parts of the architecture of an autonomous vehicle is the control system. An autonomous vehicle's control system consists of lateral control and longitudinal control. This work's objective is to reduce inaccurate speed measurements through a longitudinal vehicle model control design. The longitudinal control problem is focused on controlling the actuators (throttle and brake) for the speed profile tracking given by the motion planning [4].

The vehicle's longitudinal controllers which are employed by the researches are the PID control [4] [5] [6] [7], adaptive Network fuzzy inference system [8], adaptive control [9], model predictive control [10], fuzzy control [11], and sliding mode control [12]. The PID controller is one of the most frequently used methods for regulating a dynamic vehicle's longitudinal motion. The task of determining the gains of the ideal PID controller has been shown to be exceedingly challenging in many studies. Many researchers found the traditional adjustment techniques based on mathematical operations or trial and error to be ineffective [13]. Recently, strategies for optimizing controller gains have been proposed to prevent this issue. Consequently, gains for controllers were enhanced by employing a variety of optimization techniques, including the PSO algorithm [1] [14], the genetic algorithm [5], the ant colony optimization [15], and the salp swarm algorithm [4].

In this paper, a MPSO based tuning technique is used to modify the PID controller's coefficients for tracking target velocities effectively. IAE is used as the performance index for selecting the optimal PID controller's gains. The obtained outcomes are compared with the results of the traditional PSO method to evaluate the performance robustness of the proposed algorithm. This study's remaining sections are ordered as follows: The model system's description and the design of the optimal PID controller are provided in Section 2. The simulation's outcomes and the discussion are displayed in Section 3. In Section 4, there is a conclusion.

2. Method

2.1. Mathematical model of longitudinal vehicle dynamics

An autonomous vehicle's longitudinal control part regulates its speed in accordance with the speed profile. The speed profile is derived by a speed planner or profiler based on changes to the outside environment, such as changes in the road's design, traffic signals, and weather [16]. The longitudinal vehicle dynamic Equations (1-4) are produced via applying Newton's second law according to the vehicle diagram as shown in Fig. 1 [2].

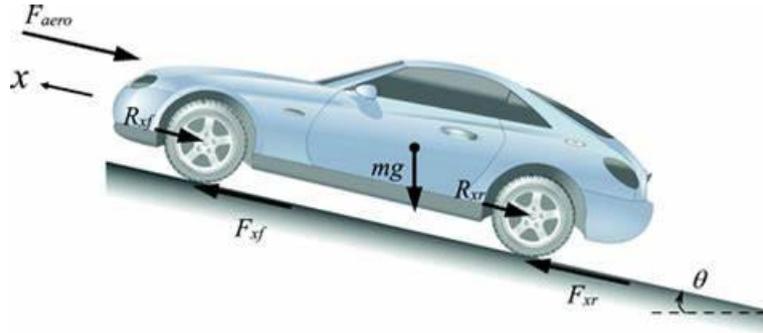


Fig. 1: Vehicle's Longitudinal Dynamics [17].

$$m \ddot{x} = -F_{xf} - F_{xr} + F_{aero} + R_{xf} + R_{xr} + mg \sin \theta \quad (1)$$

$$m \ddot{x} = -F_x + F_{aero} + R_x + mg \sin \theta \quad (2)$$

$$F_{aero} = 0.5 C_a \rho A (v)^2 \quad (3)$$

$$R_x = mg C_r v \quad (4)$$

Where m , \ddot{x} , F_{xf} , F_{xr} , F_{aero} , R_{xf} , R_{xr} , g , θ , C_a , ρ , A , v , C_r are the vehicle's mass (g), vehicle's acceleration (ms^{-2}), front tire force (N), back tire force (N), aerodynamic force (N), front rolling resistance force (N), rear rolling resistance force (N), acceleration of gravity (ms^{-2}), incline angle between the earth and vehicle (rad), drag coefficient, air density (kg/m^3), cross-sectional area (m^2), vehicle velocity (m/s), and the coefficient of the rolling resistance, respectively. The powertrain model demonstrates how velocity torque are transferred from the engine to the vehicle's wheels. The expression of powertrain model is specified by Equations (5-7) [2]:

$$\dot{w}_e J_e = T_e - T_l \quad (5)$$

$$w_w = w_e r_{eff} \quad (6)$$

$$T_e = T_{emax}(\text{Throttle angle}) + T_{emin}(1 - \text{Throttle angle}) \quad (7)$$

Where \dot{w}_e , J_e , T_e , T_l , w_w , r_{eff} , w_e , T_{emax} , and T_{emin} are the change in engine angular velocity (rad/s), inertia torque of an engine (Kg/m^2), engine torque (N-m), friction torque (N-m), wheel angular velocity (rad/s), the wheel's radius (m), engine angular velocity (rad/s), maximum engine torque (N-m), and minimum engine torque, in the order (N-m). The vehicle's tire is illustrated by the linear tire model. The Equations (8) and (9) are clarified by that model [2].

$$F_x = C_s S \quad (8)$$

Where C_s and S refer to the tire's longitudinal slip and stiffness. The PID-controller is one of the common kinds of feedback controllers applied to vehicle longitudinal control.

2.2. PID controller

The PID controller is one of the most common kinds of feedback controllers applied to vehicle longitudinal control. The PID controller involves three components: derivative, integral, and proportional. The throttle and brake values are components of the longitudinal autonomous vehicle control's velocity profile. A longitudinal controller is employed to minimize the difference between the actual and desired throttle and brake values so that the autonomous vehicle can follow the velocity profiles. Equations (10) and (11) can be used to depict the PID controller [2]:

$$\text{err}(t) = r(t) - v(t) \quad (10)$$

$$\text{th}(t) = k_p \text{err}(t) + k_d \frac{d\text{err}(t)}{dt} + k_i \int \text{err}(t). dt \quad (11)$$

Where err , r , th , k_p , k_d , and k_i the speed error, reference velocity, throttle control input, proportional gain, derivative gain, and integral gain, respectively. The brake and throttle positions are regulated by the control input. As well, the throttle values in that work are restricted to values between 0 and 1. A control system oscillates before reaching a steady state. Thus, the control response to a reference input unit step will have standard characteristics such as rise time, settling time, overshoot, and steady state error. The rise time is the time that the response needs to rise to 100% of its final value; the settling time is the time that the response needs to reach and stay within a range of an absolute specified percentage of 2% or 5% of the final response value; the overshoot means the upper response value computed from the steady state as in the Equation (12), and the steady state error is the difference between the target response and the input steady state response [18].

$$\text{OS \%} = \frac{\text{peak curve value} - \text{final curve value}}{\text{final curve value}} * 100 \quad (12)$$

2.3. Optimization algorithms

Two optimization algorithms are suggested for tuning the PID controller's parameters. The first method is the PSO method (PID-PSO), where researchers Kennedy and Eberhart explored the PSO strategy in 1995, which is based on a swarm of fishes or birds behaving. With this swarm technique, a collection of particles is put together at random. Each particle expresses a particular solution, and each particle group's members are provided by a speed and position. The swarm of particles is employed to create a huge number of fitness function solutions at each iteration. For each PSO method generation, the performance index is evaluated to identify the personal and global best (pbest and gbest), respectively. The particle with best fitness value is provided by pbest. In contrast, the gbest stands the most effective solution across all generations. At each iteration, the pbest and gbest are reorganized, and they are utilized to determine the new particular's speed and position in accordance with Equations (13 to 15) [2] [19].

$$\omega = \omega_{\max} - t(\omega_{\max} - \omega_{\min}) / T_{\max} \tag{13}$$

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (p_i^t - x_i^t) + (g_i^t - x_i^t) c_2 r_2 \tag{14}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}, i = 1, 2, \dots n \tag{15}$$

Here v , t , n , c_1 , c_2 , ω , r_1 , r_2 , p_i , g_i , T_{\max} , ω_{\min} , ω_{\max} , and x_i , are the velocity of the particle, iterations number, particles' numbers, cognitive learning factor, social learning factor, inertia weight, random value between (0 and 1), random value between (0 and 1), The particle with best fitness value, the most effective particle among group members, maximum iteration, the inertia weight's minimum value, the inertia weight's maximum value, and position of the particle. The generation of pbest and gbest is repeated until reaching the maximum limit of the iteration. The PSO method's primary drawback is the likelihood that every particle will reach a local minima in the searching space. As a result, they are unable to escape the trap on their own. The PSO method's restrictions are listed in Table I [2].

Table 1: The PSO Methods Restrictions

Name of the parameter	Value
Inertia weight's minimum value	0.4
Inertia weight's maximum value	0.9
Inertia weight's maximum value	0.9
Cognitive learning factor	2
Social learning factor	2
Upper value of the velocity	6
Size of the particle swarm	20
Maximum limit of the iteration	100
Lower gains limits	[0.00001,0.00001,0.00001]

The second method is the PID Related to the MPSO Method (PID-MPSO). A modified particle swarm optimization is suggested in this research in order to prevent the early convergence phenomenon of PSO and enhance its performance on difficult tasks. One of the PSO parameters, inertia weight (ω), can provide particles with the capacity to dynamically adjust to various situations and achieve a balance between exploration and exploitation [20]. It thus has a significant impact on PSO. The linear inertia weight approach is typically used, while the non-linear inertia weight method has better simulation and fitting capabilities. A well-known chaotic mapping called logistic can produce random values between 0 and 1. Equation (17) serves as its definition. In this study, the inertia weight is subjected to logistic chaos, leading to the construction of a non-linear inertia weight that is specified by Equation (18) [20].

$$r(1 + t) = -4r(t)(r(t) - 1), r(0) = \text{random} \tag{17}$$

Where $r(0) \neq \{0,0.25,0.5,0.75,1\}$

$$\omega = r(t)\omega_{\min} - t(\omega_{\max} - \omega_{\min}) / T_{\max} \tag{18}$$

Where $r(t)$ is a random number produced by the logistic chaotic, $\omega_{\min} = 0.4$, $\omega_{\max} = 0.9$.

Additionally, by modifying ω which has wave properties, the motion of particles will become more random, making it possible to prevent them from easily falling into a local optimum. The PSO method's restrictions are listed in Table 2 [20].

Table 2: The MPSO Methods Restrictions

Name of the parameter	Value
Inertia weight's minimum value	0.4
Inertia weight's maximum value	0.9
Cognitive learning factor	2
Social learning factor	2
Upper value of the velocity	6
Size of the particle swarm	20
Maximum limit of the iteration	100
Lower gains limits	[0.00001,0.00001,0.00001]
Upper gains limits	[10,10,10]

Figs. 2 and 3 show how the proposed block diagram and flowchart for the PID-MPSO controller were built, respectively. The majority of optimization problems are resolved differently depending on how the objective function is represented. The IAE serves as the specific objective function in the proposed techniques, and it is employed to reduce the speed error of the PID controller as indicated in Equation (19) [2].

$$IAE = \int_0^{\infty} |e(t)| \cdot dt \tag{19}$$

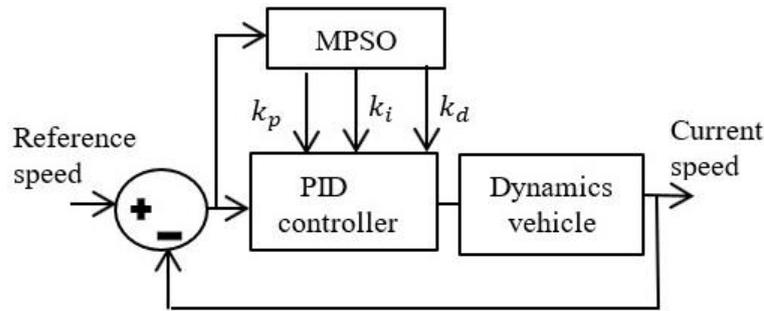


Fig. 2: Block Diagram for the PID-MPSO Controller.

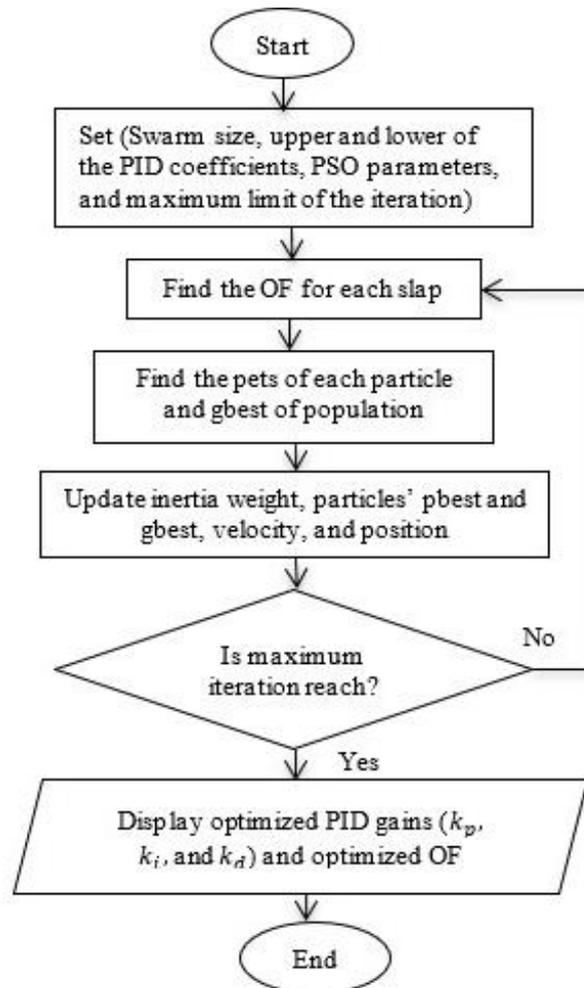


Fig. 3: Flowchart of the PID-MPSO.

In order to know the best between two optimal controllers, the performance enhancement percentage is selected in Equation (20):

$$\text{Enhancement \%} = \left(1 - \frac{\text{best IAE value}}{\text{worst IAE value}}\right) * 100 \quad (20)$$

3. Results and discussion

In this part, the simulation outcomes of the suggested controllers are run by the Python simulation program. A desired speed is fixed to 30 m/s and it is applied to the longitudinal control system of the dynamic vehicle. Then, to solve the speed tracing issue, PID controller's coefficients were adjusted using the PSO and MPSO algorithms. According to the Table III, the MPSO and PSO algorithms' convergence curve is depicted visually in Fig. 3. Due to its local minimum problem, the PSO method converged in this figure more slowly than MPSO. In other words, MPSO was more accurate than the PSO algorithm at identifying optimal gains. That demonstrates the strength of the suggested algorithm.

Table 3: Fitness Values of the PID-PSO and PSO-MPSO Controllers for Each Iteration Number

Iteration number	IAE for the PID-PSO	Iteration number
1	4.49958318619103	3.7483528623037996
2	1.879121084842843	1.9539154133752570
3	1.822493037856832	1.8422200463958256
4	1.811592133661932	1.8169098770274560

5	1.791395639085765	1.8087362815788761
6	1.791395639085765	1.7992517962794996
7	1.791395639085765	1.7943420607308913
8	1.790997232664367	1.7915147490246646
9	1.790997232664367	1.7904879922062853
10	1.790234922669186	1.7888988777400283
11	1.790234922669186	1.7808726762748919
12	1.790234922669186	1.7802882732826740
13	1.790234922669186	1.7769478956017595
14	1.790234922669186	1.7743311725178492
15	1.790234922669186	1.7707029034343424
16	1.790234922669186	1.7617385018161453
17	1.790207263400226	1.7519847874465070
18	1.790207263400226	1.7470059675523133
19	1.790207263400226	1.7470059675523133
20	1.790207263400226	1.7463450982601420
21	1.790207263400226	1.7456741413737664
22	1.790207263400226	1.7455876881620458
23	1.790207263400226	1.7455698156593342
24	1.790207263400226	1.7455698156593342
25	1.790207263400226	1.7455564416282332
26	1.790207263400226	1.7455564416282332
27	1.790207263400226	1.7455484323776844
28	1.790207263400226	1.7455468894831003
29	1.790207263400226	1.7455468894831003
30	1.790207263400226	1.7455468894831003
31	1.790207263400226	1.7455467704101923
32	1.790207263400226	1.7455467704101923
33	1.790207263400226	1.7455467704101923
34	1.790207263400226	1.7455467704101923
35	1.790207263400226	1.7455467245225227
36	1.790207263400226	1.7455466482578117
37	1.790207263400226	1.7455466482578117
38	1.790207263400226	1.7455466086777860
39	1.790207263400226	1.7455466086777860
40	1.790207263400226	1.7455466086777860
41	1.790207263400226	1.7455466086777860
42	1.790207231720718	1.7455465829511543
43	1.790207231720718	1.7455465807834347
44	1.790207231720718	1.7455465801114978
45	1.790206960950600	1.7455465780009851
46	1.790206960950600	1.7455465780009851
47	1.790206960950600	1.7455465777547580
48	1.790206887261685	1.7455465777547580
49	1.790206887261685	1.7455465777547580
50	1.790206887261685	1.7455465777547580
51	1.790206887261685	1.7455465777547580
52	1.790206887261685	1.7455465777471095
53	1.790206887261685	1.7455465777471095
54	1.790206887261685	1.7455465776763137
55	1.790206887261685	1.7455465776335326
56	1.790206887261685	1.7455465775675827
57	1.790206887261685	1.7455465773239840
58	1.790206887261685	1.7455465773239840
59	1.790206887261685	1.7455465772794168
60	1.790206865286967	1.7455465772794168
61	1.790206865286967	1.7455465771303962
62	1.790206865286967	1.7455465771100982
63	1.790206865286967	1.7455465770550964
64	1.790206865286967	1.7455465770179070
65	1.790206865286967	1.7455465770134855
66	1.790206865286967	1.7455465770134855
67	1.790206865286967	1.7455465770121732
68	1.790206865286967	1.7455465770120742
69	1.790206863913207	1.7455465770094216
70	1.790206863913207	1.7455465770074907
71	1.790206863505661	1.7455465770053153
72	1.790206863505661	1.7455465770048080
73	1.790206863505661	1.7455465770047494
74	1.790206863505661	1.7455465770045488
75	1.790206863505661	1.7455465770042915
76	1.790206863505661	1.7455465770036702
77	1.790206863505661	1.7455465770031386
78	1.790206863505661	1.7455465770005012
79	1.790206863505661	1.7455465769984007
80	1.790206863505661	1.7455465769943481
81	1.790206863505661	1.7455465769905410
82	1.790206863505661	1.7455465769877354
83	1.790206863505661	1.7455465769812166
84	1.790206863505661	1.7455465769812166

85	1.790206863505661	1.7455465769812166
86	1.790206863505661	1.7455465769797642
87	1.790206863505661	1.7455465769797642
88	1.790206863505661	1.745546576979652
89	1.790206863505661	1.7455465769792207
90	1.790206863505661	1.7455465769788052
91	1.790206863505661	1.7455465769788052
92	1.790206863505661	1.7455465769786294
93	1.790206863505661	1.7455465769786294
94	1.790206863505661	1.7455465769786092
95	1.790206863505661	1.7455465769786092
96	1.790206863505661	1.7455465769785328
97	1.790206863505661	1.7455465769785328
98	1.790206863505661	1.7455465769785328
99	1.790206863505661	1.7455465769785328
100	1.790206863505661	1.7455465769785328

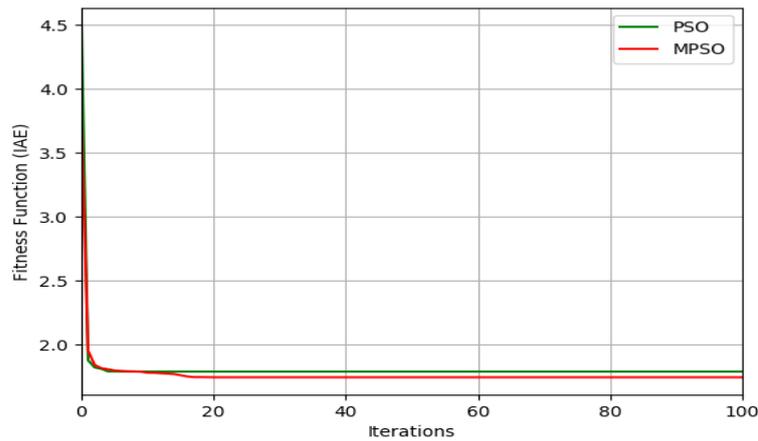


Fig. 3: Convergence Curve of the PID-PSO and the PID-MPSO

Table 4: Fitness Value and Optimal Gains of the PID Controller

Optimization Method	k_p	k_i	k_d
PSO	5.5788	1.79×10^{-5}	10
MPSO	5.2909	1.79×10^{-3}	10

The results provided in Table 2 involve the optimal controller’s coefficients and the minimal performance index value that the PSO and MPSO algorithms have to offer. The data listed in this table demonstrated that the MPSO strategy had a better fitness value in comparison to the PSO method. The performance enhancement percentage of the PID- MPSO technique than the PID-PSO technique is given by 2.5%. Fig. 4 displays the system response under the PID controller’s tuned parameters based on PSO and MPSO algorithms for comparison purposes. However, it was shown that the combined PID with the MPSO method provided a quicker and more dependable response compared to the PID-PSO technique.

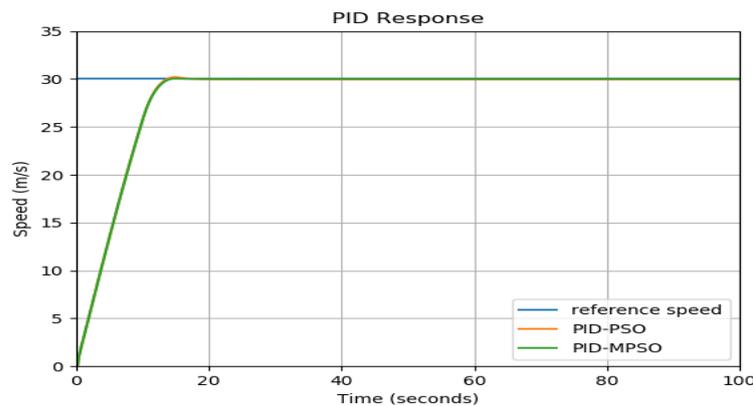


Fig. 4: Speed Response of the Optimized PID Controller.

Table 5: Speed Response Specifications for the Optimized PID Controller

Optimal Controller	OS (%)	SSE (cm)	t_s (s)	t_r (s)
PID-PSO	0.8546	0.0557	0.05547	9.4979
PID-MPSO	0.128	0.0009	0.00084	9.5148

The improved PID response parameters are contrasted in Table 5 using OS, SSE, t_s (2%), and t_r standards. The table’s outcomes showed that the PID-MPSO has lower values of the OS, SSE, and t_s than the PID-PSO while still maintaining a lower rise time value. To test the response of the PID-MPSO for minimizing external disturbances effect, two experiments are performed: the first one is setting the

coefficient of the wind to 5 m/s and 15 m/s. The second one is setting the incline angle to $\pi/10$ rad and $\pi/8$ rad. The simulation results of these tests are displayed in Figs. 5, 6, 7, and 8, respectively.

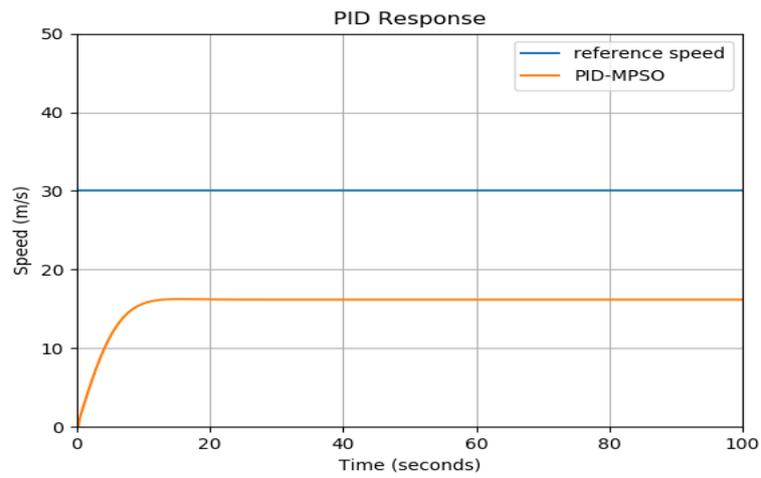


Fig. 5: Speed Tracking Result Under Drag Coefficient at Value of 5 Rad/S.

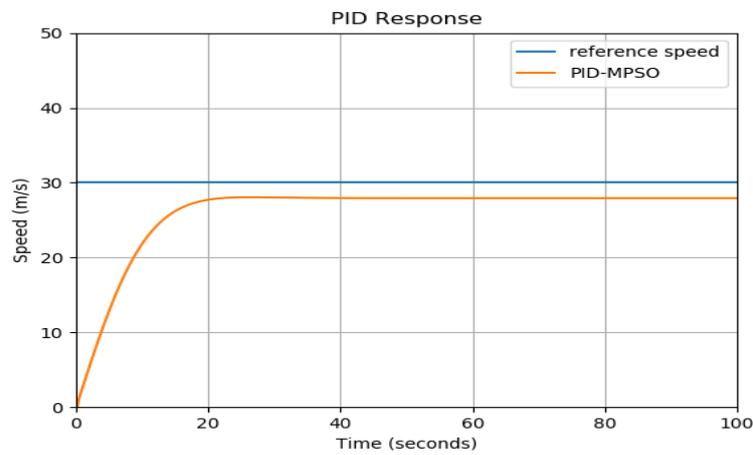


Fig. 6: Speed Tracking Result Under Drag Coefficient at Value of 15 Rad/S.

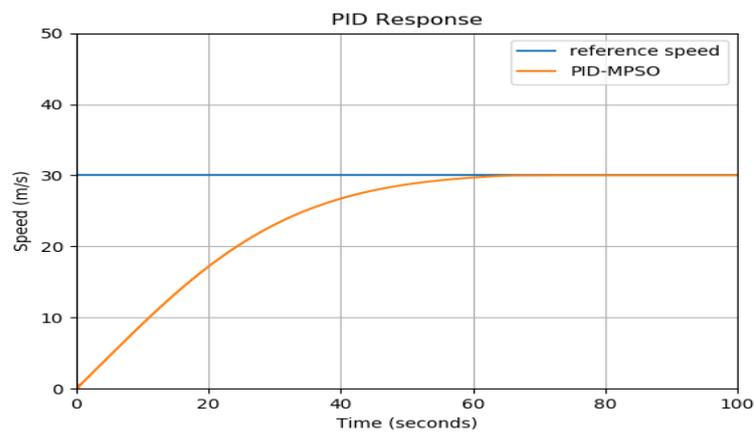


Fig. 7: Speed Tracking Result Under Incline Angle at Value of 18 Deg.

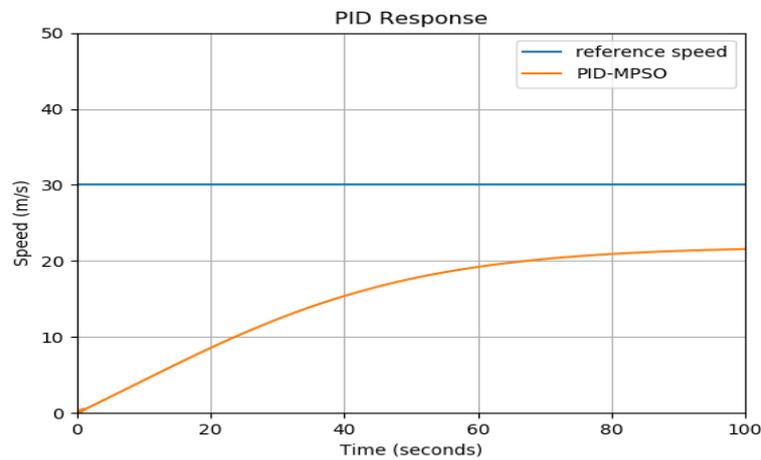


Fig. 8: Speed Tracking Result Under Incline Angle at Value of 22.5 Deg.

Figs. 5 and 6 show the aerodynamic force effect: when the drag coefficient increases, the aerodynamic force increases, which reduces the vehicle's speed. The tracking outcomes displayed by these Figs are worse in comparison with Fig. 4 due to the drag coefficient effect. Besides, Figs. 7 and 8 clearly show that the incline angle effect increases the incline angle, reducing the vehicle's speed. At a small value of incline angle, such as 18 deg, the PID-PSO attenuates the external disturbance effect, but at a large value of incline angle, such as 22.5 deg, the PID-PSO is not able to reduce the external disturbance effect.

4. Conclusion

In conclusion, this study introduced a modified particle swarm optimization (MPSO) algorithm for tuning the PID controller parameters in the longitudinal dynamics control of an autonomous vehicle. Comparative analysis with the traditional PSO method demonstrated that the PID controller optimized with MPSO outperforms its PSO-based counterpart in tracking desired signals. The observed improvement, quantified by 2.5%, underscores the efficacy of the MPSO technique. Furthermore, the study highlighted the robustness of the proposed algorithm and its potential implications for enhancing the performance and safety of autonomous vehicle systems. Future works of that research include adding a feedforward controller to the PID controller or designing an adaptive PID controller for solving the problem of the delay response to errors and reducing external disturbance effects such as drag coefficient increases and increases the incline angle.

References

- [1] Y. Kebbati, N. Ait-Oufroukh, V. Vigneron, and D. Ichalal, "Coordinated PSO-PID based longitudinal control with LPV-MPC based lateral control for autonomous vehicles," in *European Control Conference (ECC)*, 2022, pp. 518–523, <https://doi.org/10.23919/ECC55457.2022.9838192>.
- [2] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Veh.*, vol. 1, no. 1, pp. 33–55, 2016, <https://doi.org/10.1109/TIV.2016.2578706>.
- [3] G. Hadi and A. A. A. Al-moadhen, "Optimization of Path Tracking of Self-Acting Mobile Robotic System," University of Kerbala, 2022.
- [4] G. H. S. Elias, A. Al-Moadhen, and H. Kamil, "Optimizing the PID controller to control the longitudinal motion of autonomous vehicles," in *AIP Conference Proceedings*, 2023, vol. 2591, no. 1, p. 040045, <https://doi.org/10.1063/5.0120396>.
- [5] S. Azam, F. Munir, and M. Jeon, "Dynamic Control System Design for Autonomous Car," in *International Conference on Vehicle Technology and Intelligent Transport Systems*, 2020, pp. 456–463, <https://doi.org/10.5220/0009392904560463>.
- [6] R. M. De Santis, "A novel PID configuration for speed and position control," *J. Dyn. Syst. Meas. Control*, vol. 116, no. 3, pp. 542–549, 1994, <https://doi.org/10.1115/1.2899250>.
- [7] S. Chen and H. Chen, "MPC-based path tracking with PID speed control for autonomous vehicles," in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 892, no. 1, p. 012034, <https://doi.org/10.1088/1757-899X/892/1/012034>.
- [8] M. Marcano, J. A. Matute, R. Lattarulo, E. Martí, and J. Pérez, "Low Speed Longitudinal Control Algorithms for Automated Vehicles in Simulation and Real Platforms," *Complexity*, p. 12, 2018, <https://doi.org/10.1155/2018/7615123>.
- [9] H. Kim, D. Kim, I. Shu, and K. Yi, "Time-varying parameter adaptive vehicle speed control," *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 581–588, 2016, <https://doi.org/10.1109/TVT.2015.2402756>.
- [10] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 3, pp. 556–566, 2011, <https://doi.org/10.1109/TCST.2010.2049203>.
- [11] J. E. Naranjo, C. González, R. García, and T. De Pedro, "ACC+Stop&Go maneuvers with throttle and brake fuzzy control," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 213–224, 2006, <https://doi.org/10.1109/TITS.2006.874723>.
- [12] Y. Li and L. He, "Counterbalancing Speed Control for Hydrostatic Drive Heavy Vehicle under Long Down-Slope," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 4, pp. 1533–1542, 2015, <https://doi.org/10.1109/TMECH.2014.2385700>.
- [13] F. A. Ma'ani and Y. Y. Nazaruddin, "Optimization of Longitudinal Control of an Autonomous Vehicle using Flower Pollination Algorithm based on Data-driven Approach," *Int. J. Sustain. Transp. Technol.*, vol. 3, no. 2, pp. 58–65, 2020, <https://doi.org/10.31427/IJSTT.2020.3.2.4>.
- [14] S. Allou and Y. Zennir, "A Comparative Study of PID-PSO and Fuzzy Controller for Path Tracking Control of Autonomous Ground Vehicles," in *15th International Conference on Informatics in Control, Automation and Robotics*, 2018, pp. 306–314, <https://doi.org/10.5220/0006910903060314>.
- [15] M. Karami, A. R. Tavakolpour-Saleh, and A. Norouzi, "Optimal Nonlinear PID Control of a Micro-Robot Equipped with Vibratory Actuator Using Ant Colony Algorithm: Simulation and Experiment," *J. Intell. Robot. Syst.*, vol. 99, no. 3–4, pp. 773–796, 2020, <https://doi.org/10.1007/s10846-020-01165-5>.
- [16] Y. Kebbati, N. Ait-oufroukh, V. Vigneron, D. Ichalal, and D. Gruyer, "Optimized self-adaptive PID speed control for autonomous vehicles To cite this version: HAL Id: hal-03442081 Optimized self-adaptive PID speed control for autonomous vehicles," in *International Conference on Automation and Computing (ICAC)*, 2021, pp. 1–6, <https://doi.org/10.23919/ICAC50006.2021.9594131>.
- [17] R. Rajamani, "Longitudinal Vehicle Dynamics," in *Vehicle Dynamics and Control*, 2nd Editio., Springer, Boston, MA, 2012, pp. 87–111, https://doi.org/10.1007/978-1-4614-1433-9_4.
- [18] K. Ogata, *Modern Control Engineering*, Fourth edi. New Jmey: Prentiffi Hall, 2002.

- [19] G. Hadi, S. Elias, A. Al-moadhen, and H. Kamil, "Lateral Control of an Autonomous Vehicle Based on Salp Swarm Algorithm," 2023, vol. 030043, no. March, <https://doi.org/10.1063/5.0120403>.
- [20] H. Liu, X. W. Zhang, and L. P. Tu, "A modified particle swarm optimization using adaptive strategy," *Expert Syst. Appl.*, vol. 152, p. 113353, 2020, <https://doi.org/10.1016/j.eswa.2020.113353>.