INTERNATIONAL JOURNAL OF AUTOMOTIVE SCIENCE AND TECHNOLOGY

2024, VOL. 8, NO:1, 110-117 www.ijastech.org e-ISSN: 2587-0963

Trajectory Tracking Control Using Evolutionary Approaches for Autonomous Driving

Yiğit Çağatay KUYU[1]

[1.]*R&D Department, Karsan Automotive, Bursa, Türkiye*

Research Article

Abstract

Capitalizing on the strides in artificial intelligence and the escalating demand for safer and more efficient traffic systems, the investigation unveils a trio of evolutionary algorithms - namely Grey Wolf Optimizer (GWO), Multi-Verse Optimizer (MVO) and Salp Swarm Algorithm (SSA) - in the context of hyperparameter calibration for the Proportional-Integral-Derivative (PID) controller. Revered for their classical simplicity and widespread industrial use, PID controllers are pivotal in feedback control systems, ensuring desired system performance through meticulous parameter adjustments. This research introduces a novel application of GWO, MVO, and SSA in the realm of PID control, aiming to optimize the controllers' parameters. To exemplify the utility of the proposed algorithms, two distinct trajectory scenarios are employed as target trajectories. Rigorous numerical evaluations, accompanied by graphical analyses, showcase the prowess of these algorithms in steering the trajectory tracking process. By pioneering the application of these optimizers in the PID controller domain, this investigation not only demonstrates their superior performance over traditional methods but also contributes to the broader field of control engineering by suggesting a more efficient approach to traffic system optimization. This exploration also paves the way for further research into leveraging advanced optimization techniques to elevate the safety and efficiency of traffic systems.

Keywords: Autonomous driving; Evolutionary algorithms; PID; Trajectory tracking control

History
Received 01.09.2023
Revised 25.12.2023
Accepted 12.02.2024

Contact
* Corresponding author
Yiğit Çağatay Kuyu
yigit.kuyu@karsan.com.tr
Address:R&D Department, Karsan Company, Bursa, Türkiye
Tel:+902244842170

To cite this paper: Kuyu, Y.Ç. Trajectory Tracking Control Using Evolutionary Approaches for Autonomous Driving. International Journal of Automotive Science and Technology. 2024; 8 (1): 110-117. http://dx.doi.org/10.29228/ijastech..1354082

1. Introduction

Capitalizing on the rapid advancements in artificial intelligence technology and a surging demand for enhanced traffic safety and operational efficiency, scholarly focus has increasingly converged on the domain of autonomous vehicles (AVs) over recent decades. Broadly dissected, the architecture of autonomous vehicles encompasses three fundamental modules: perception, planning, and control. The planning module orchestrates the generation of a sequential array of waypoints that chart the vehicle's course toward its intended destination. Subsequently, these waypoints are relayed to the control system, entrusted with the task of executing precise trajectory tracking. The primary objective of trajectory tracking is to effectually and swiftly follow a designated target trajectory, all within the operational parameters and limitations [1, 2]. Within the realm of autonomous driving trajectory tracking, this strategy bifurcates into a tripartite categorization of motion tasks: point-to-point motion, path following, and trajectory tracking [3]. This classification serves to disentangle complexities into discrete task units, thereby facilitating the derivation of apt controllers tailored to the respective tasks. Commencing with point-to-point motion, the fundamental objective entails navigating from an initial starting position to a predetermined destination point, irrespective of the specific course or trajectory. Conversely, the pursuit of trajectory tracking hinges upon the meticulous pursuit and alignment with a pre-established geometric path within stipulated temporal confines. In parallel, the conception of path following materialized as an intermediary to enable the vehicle's attainment of the specified geometric trajectory while steadfastly adhering to its contours.

PID controllers, renowned for their elementary configuration, theoretical underpinning, and extensive industrial adoption, constitute a quintessential exemplar of classical control methodologies. Comprising three distinct terms of action, the PID controller's operational premise is typically predicated upon the discrepancy between the present state and the desired response. These terms, namely Proportional (P), Integral (I), and Derivative (D), exert respective influences on the modulation of the triggering error. The annals of research in this domain have yielded a plethora of investigations. An innovative physics-grounded path planning framework tailored for the autonomous traversal of tracked vehicles across rugged terrains has been propounded. The proposed methodology melds a hybrid planner and simulator, orchestrating an intricate interplay of these facets by orchestrating closed-loop robot motion simulation through a low-level controller embedded within a realistic terrain model within a physics engine [4]. In a parallel endeavor, an inventive steering control strategy has been introduced, catering to dual-motor coupled drive systems, where torque control forms the foundation, and speed closed-loop PID control governs the dual-side drive motors. Efficacy in terms of desired steering angular velocity response and vehicle maneuverability enhancement was substantiated through simulation results [5]. Similarly, a speed tracking controller, predicated upon the PID control paradigm, was formulated to determine the requisite vehicular traction force for attaining the desired velocity [6]. A trajectory planning and tracking paradigm, synergistically integrating artificial potential and PID feedback, was engendered to chart target trajectories. Empirical and simulation outcomes were proffered, exemplifying heightened tracking accuracy and steering smoothness vis-à-vis conventional Model Predictive Control (MPC) approaches [7]. Amidst a pantheon of adept control methodologies, PID control perseveres as an enduring focal point. Noteworthy attributes of PID control encompass its lucid architecture, commendable control efficacy, robust design, and facile implementation [8].

Over the course of the past four decades, evolutionary algorithms have been extensively employed across a diverse spectrum of domains within the natural and social sciences, encompassing disciplines such as physics, mathematics, engineering, and economics [9, 10]. In broad contours, these algorithms initially generate candidate solutions in a stochastic manner, which serve as proxies for potential problem resolutions. Subsequently, these candidate solutions undergo evaluation via the objective function to ascertain their fitness values. Proceeding onward, iterative procedures within the algorithms recalibrate these candidates, progressively converging towards the global minimum point. This iterative process persists until a stipulated termination criterion, encompassing parameters like the number of iterations, tolerance threshold, or number of function evaluations, is satisfied. The endeavor of discerning the optimal configuration of hyperparameters for a model or algorithm, commonly referred to as hyperparameter optimization, resonates as an approach capable of mitigating the exigencies of labor-intensive human intervention while circumventing the perils intrinsic to manual searches [11, 12]. It merits noting that the PID methodology necessitates complementary algorithms for the discernment and meticulous refinement of its hyperparameters. The quest for well-suited, optimized values of these hyperparameters proves to be a formidable undertaking, given the intricate interplay of factors such as the intricate nature of vehicle dynamics, the vagaries arising from external disturbances' uncertainty, and the vehicle's non-holonomic constraint [13].

In the ambit of this scholarly investigation, triads of distinct evolutionary algorithms (EAs), namely the grey wolf optimizer [14], multi-verse optimizer [15], and salp swarm algorithm [16]. These algorithms are strategically employed to meticulously calibrate the hyperparameters of the PID controller, a cornerstone of feedback control systems for autonomous vehicles. This novel application of evolutionary algorithms to PID controller tuning is a significant contribution to the field for several reasons.

First and foremost, our study marks an inaugural endeavor in this category, breaking new ground by deploying these advanced optimizers to discern the optimal hyperparameters for the PID controller. This innovation establishes a precedent that pushes the boundaries of existing knowledge, offering a fresh perspective on trajectory tracking challenges in the context of autonomous vehicles.

Secondly, our research goes beyond theoretical exploration. We provide a comprehensive evaluation of the algorithms' performance, combining quantitative analysis with numerical data and qualitative insights through graphical representation. This thorough assessment not only validates the practical effectiveness of these algorithms but also contributes to a scholarly discourse that can inspire further research in this promising domain.

In summary, our study's primary contributions lie in its pioneering approach to PID controller hyperparameter calibration, the comprehensive performance evaluation of the employed algorithms, and the potential for further exploration and advancements in autonomous vehicle trajectory tracking. These contributions collectively enhance our understanding of feedback control systems and pave the way for improved autonomous vehicle navigation and safety.

The main contents of this paper are as follows. In Sections 2-3, the vehicle model and PID controller are presented. In Section 4, the working principles of the algorithms used are explained. In Section 5, the results obtained are given and discussed. Finally, Section 6 is devoted to state the conclusion of the paper.

2. The Vehicle Model

This study utilizes the Kinematic Bicycle Model [17] to simulate the actions of the self-driving vehicle. This model encapsulates a streamlined rendition of bicycle kinematics, tailored to depict the vehicular motion when traversing at low velocities. In the pursuit of analytical expediency, it is postulated that the vehicle maintains a constant, subdued velocity across an even terrain, devoid of wheel slippage, thereby affording the supposition of rigid wheels [18]. Traditionally, the automotive rear axle center is commonly acknowledged as the point of reference, and the essence of the control objective is to ensure the alignment of this center point with the intended trajectory. Based on the in-depth kinematic scrutiny, the mathematical representation of the vehicle's motion is formulated as presented in prior studies [19, 20]:

$$\dot{\mathrm{x}} = \mathrm{v} \cos \delta \tag{1}$$

$$\dot{y} = v \cos \delta \tag{2}$$

$$\dot{\delta} = v \tan \theta / L \tag{3}$$

In this context, where (x, y) corresponds to the positional displacement within the global reference frame coordinates, δ symbolizes the orientation angle of the vehicle, v stands for the longitudinal velocity at the midpoint of the rear axis, L denotes the wheelbase, and θ represents the steering angle of the front wheel. The nonlinear equation below represents the formulation of the vehicle's kinematics model:

$$X(\dot{t}) = f(X(t), U(t)) \tag{4}$$

where state variable $X(t) = [x(t)y(t)\delta(t)]^T$ and the control amount $U(t) = [v(t)\theta(t)]^T$.

3. PID Controller

PID controller represents a ubiquitous feedback control methodology extensively applied within the realm of engineering for the regulation of dynamic systems. Its fundamental objective resides in the precise modulation of control input, with the overarching aim of attaining and preserving a desired output state. This attainment is predicated upon the judicious management of the discrepancy existing between the intended reference value, known as the setpoint, and the present actual process variable.

The PID controller derives its operational essence from a triad of hyperparameters, whereby each parameter corresponds to an individual constituent of the acronym P-I-D. These three components play distinct roles in shaping the controller's response [21]:

Proportional (P) Term: The proportional term, denoted as K_p, is primarily responsible for instantaneously responding to the current error. It produces an output value that is proportional to the current error value. In other words, it determines how much the control action should respond to the present error.

Integral (I) Term: The integral term, represented as K_i, considers past errors and accumulates them over time. It is employed to eliminate any residual steady-state error that might exist after the proportional and derivative terms have been applied. The integral term acts to counteract sustained error accumulation.

Derivative (D) Term: The derivative term, designated as K_d, focuses on the rate of change of the error. It predicts future error trends and counteracts the present rate of error change. By doing so, it helps prevent overshoots and oscillations in the system's response. The main equations of the PID controller can be seen in Table 1 [22]. The combined expression of the PID controller, rendered as:

$$Y(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d d(e(t))/dt \tag{5}$$

In the context of the PID controller framework, the tracking error signal, denoted as $e(t)$, emerges as a pivotal component encapsulating the disparity between the reference trajectory $r(t)$ and the actual output $y(t)$ of the controlled system. The integral of the absolute value of the error (IAE) is addressed through the integral term in the PID controller. Central to the unctionality of the PID control scheme are three distinct controller parameters, namely the proportional gain K_p integral gain K_i and derivative gain K_d. These parameters collectively contribute to shaping the controller's responsiveness and effectiveness in minimizing the tracking error, thereby ensuring the alignment of the controlled output with the desired reference trajectory. The effect of the PID parameters to system response can be seen in Table 2 [23].

4. Evolutionary Algorithms Employed

4.1. Grey Wolf Optimizer

The Grey wolf optimizer (GWO) was presented in 2014 and took inspiration from how gray wolves hunt by using the social hierarchy between them [14]. Social hierarchy within this context can be elucidated based on a classification of dominance levels, encompassing four distinct strata: alpha (α), beta (β), delta (δ) and omega (ω). Positioned at the pinnacle of this hierarchy, the alpha wolf assumes a preeminent role in orchestrating search endeavors. Sequentially following, the beta wolf occupies the second rank, while the delta wolf stands at the third tier. Concluding this hierarchy, the omega wolf resides at the fourth echelon, predominantly shadowing the trajectories traced by its counterparts in order to navigate toward regions of the search space that exhibit promise and potential. Encircling prey materializes in the context of hunting prey and involves a strategic maneuver to encircle the target. The underlying mathematical model for this tactical approach is presented below:

$$D = \left|C.X_p(t) - X(t)\right| \tag{6}$$

$$X(t+1) = X_p - A.D \tag{7}$$

where t indicates the current iteration, $A = 2a.r_1 - a$, $C = 2.r_2$ and r_1, r_2 are random vectors in [0,1]. Within the GWO framework, the Hunting phase is orchestrated by the alpha wolf, typically leading the hunting endeavors, while the beta and delta wolves might intermittently participate in the hunt. Consequently, the top three solutions, represented by alpha, beta, and delta, contribute to updating the positions of the grey wolves. To mathematically emulate this hunting behavior, the subsequent equations are introduced:

$$D_\alpha = |C_1 * X_\alpha - X| \tag{8}$$

$$D_\beta = \left|C_2 * X_\beta - X\right| \tag{9}$$

$$D_\delta = |C_3 * X_\delta - X| \tag{10}$$

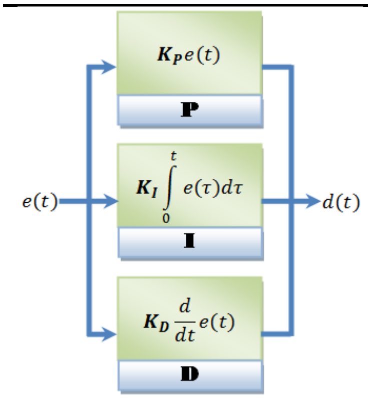
$$X_1 = X_\alpha - A_1 D_\alpha \tag{11}$$

$$X_2 = X_\beta - A_2 \ D_\beta \tag{12}$$

$$X_3 = X_\delta - A_3 D_\delta \tag{13}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \tag{14}$$

Table 1. The main equations of PID controller



Continuous-time	$d(t) = K_P e(t) + K_I \int_0^t e(\tau)d\tau + K_D \frac{d}{dt} e(t)$ $d(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau)d\tau + T_d \frac{d}{dt} e(t) \right)$
Discrete-time	$d[n] = K_c \left(e[n] + \frac{T_s}{T_i} \sum_{i=0}^{n} e[i] + \frac{T_d}{T_s}(e[n] - e[n-1]) \right)$

Table 2. Parameter effect of the PID controller

	K_P	K_I	K_D
Rise time	-	-	~
Overshoot	+	+	-
Settling time	~	+	-
Steady-state error	-	-	~

4.2. Multi-Verse Optimizer

The underpinning principles of the multi-verse optimizer draw inspiration from three fundamental cosmological concepts: the white hole, the black hole, and the wormhole [15]. Within this algorithmic framework, mathematical representations of these concepts are harnessed to facilitate exploration, exploitation, and local search processes, respectively. Specifically, the notions of white holes and black holes serve as agents for global search strategies, whereas the concept of wormholes augments local search endeavors. This paradigm conceives each solution as a distinct universe, with individual variables comprising objects within these universes. Moreover, every solution is endowed with an inflation rate, proportionate to its corresponding fitness function evaluation. Within the MVO algorithm, the ensuing interactions between the three aforementioned universe types are governed by a set of rules:

• A higher inflation rate correlates with an increased likelihood of manifesting white holes.

• Conversely, an elevated inflation rate corresponds to a diminished likelihood of encountering black holes.

• Universes characterized by higher inflation rates tend to transmit objects via white holes.

• In contrast, universes with lower inflation rates exhibit a propensity to receive more objects through black holes.

• Irrespective of the inflation rate, objects in all universes exhibit the potential for stochastic displacement towards the optimal universe through wormholes.

$$U = [x_1 x_2 \dots x_n] \ ,$$

$$x_i = [x_{i,1} x_{i,2} \dots x_{i,D}] \ , i = 1,2, \dots M \tag{15}$$

In Eq.15, U stands for a population, M is the number of the universe and D is the dimension of the problem.

$$x_i^j = \begin{cases} x_k^j & r1 < NI(Ui) \\ x_i^j & r1 \geq NI(Ui) \end{cases} \tag{16}$$

In Eq.16, x_i^j denotes the jth parameter of ith universe, Ui refers to the ith universe, $NI(Ui)$ is normalized inflation rate of the ith universe, r1 is a random number in [0, 1], and x_k^j indicates the jth parameter of kth universe selected by a roulette wheel selection mechanism. To instigate localized modifications within each universe and enhance the likelihood of augmenting inflation rates via wormholes, a premise is adopted wherein wormhole tunnels consistently connect a given universe with the most optimal universe identified thus far.

4.3. Salp Swarm Algorithm

Introduced in 2017, the salp swarm algorithm is rooted in the population-based paradigm and derives inspiration from the collective behavior of salp swarms in their natural habitat [16]. In the hierarchical structure of the salp chain, the vanguard salp assumes the role of the leader while the subsequent members trail in its wake. To render this phenomenon into a mathematical model, the populace is initially partitioned into two distinct factions: the leader and the followers. The leader embodies the foremost salp within the chain, while the remaining constituents are categorized as followers. Eponymously designated, the leader exerts directional influence over the swarm's trajectory, steering their course, while the followers, as their nomenclature suggests, adhere to a sequenced emulation of their predecessors. The below equation is given to update the position of the leader. To update the position of the followers, the following equations are used.

$$x_j^i = \frac{1}{2}at^2 + v_0 t \tag{17}$$

where $i \geq 2$, x_j^i shows the position of ith follower salp in jth dimension, t is time and v_o is the initial speed.

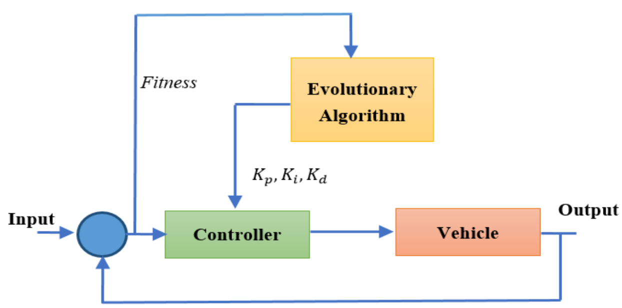


Fig. 1. The framework of EA based PID controller

The depicted framework, as illustrated in Fig. 1, is employed for the purpose of trajectory tracking control, wherein the assessment of fitness is accomplished by means of the root mean square error, as represented by Eq. (18) below.

$$Fitness = \sqrt{\frac{\sum_i^N CTE_i^2}{N}} \tag{18}$$

Here, CTE_i denotes the cross-track error between the vehicle and the corresponding segmented path at the ith instance, and N signifies the duration over which this evaluation is conducted.

5. Experimental Results

In order to verify the performance of the proposed algorithms in finding hyperparameters of PID controller, two example paths given in Fig.3 are used as target trajectories. We ran the experiments on a Desktop PC with Intel Xeon Gold 5220R processor, 256 GB RAM, Ubuntu 22.04. For the fairness of comparison, the algorithm-specific parameter values proposed by the original studies [14, 15, 16] were used in the experiments. The population size and maximum iteration are chosen 10 and 100, respectively for whole algorithms that run 10 times independently. The performance of the algorithms is evaluated numerically and graphically. The best hyperparameters found by each algorithm are also given.

Initially, the process begins with the initialization of the evolutionary algorithms (EAs), namely GWO, MVO, and SSA (See Section 4). This involves setting up the algorithm-specific parameters, such as population size, maximum iterations, and other relevant settings (See references 14, 15, 16). Once the algorithms are initialized, the vehicle's trajectory tracking process is executed. This involves the movement of the vehicle along a predefined reference trajectory using a control strategy based on the PID controller. The vehicle's position is calculated over time using a simplified kinematic bicycle model (See Section 2). While the vehicle follows the reference trajectory, the CTE is calculated for each waypoint along the trajectory (See Eq. 20). The CTE represents the deviation between the vehicle's actual position and the desired position on the reference trajectory. The calculated CTE errors are used as the objective function values. The evolutionary algorithms (GWO, MVO, and SSA) are then applied to optimize the parameters of the PID controller. The goal is to find PID parameters that minimize the CTE errors and improve the trajectory tracking performance. The steps can be represented below (Fig. 2):

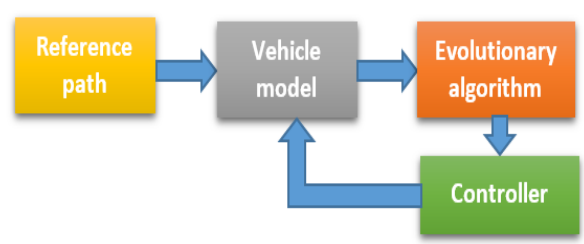
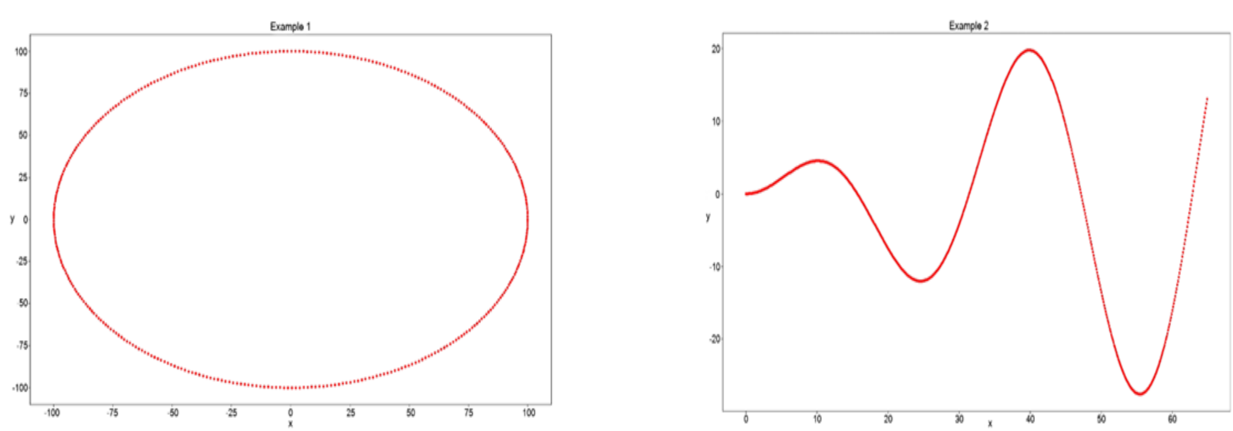


Fig. 2. The process of the evolutionary optimizatio



(a) Example 1 (b) Example 2

Fig. 3. Example trajectories

Table 3. Comparative results of the algorithms for example 1 (Error data is in units.)

	GWO	SSA	MVO
Best	11.818	11.818	11.818
Mean	11.818	11.818	11.818
Std	6.223e-6	3.652e-5	1.239e-5
Run time (sec)	23.11	23.07	23.29

Table 4. Best found parameters of the algorithms for example 1

	GWO	SSA	MVO
K_p	0.020	0.020	0.020
K_i	1.132e-10	2.021e-9	1.820e-9
K_d	6.082e-4	6.085e-4	6.082e-4

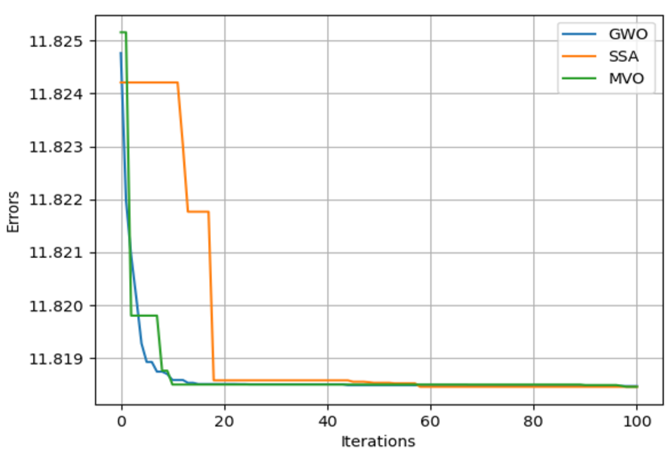


Fig. 4. Convergence graphs of the algorithms for example 1

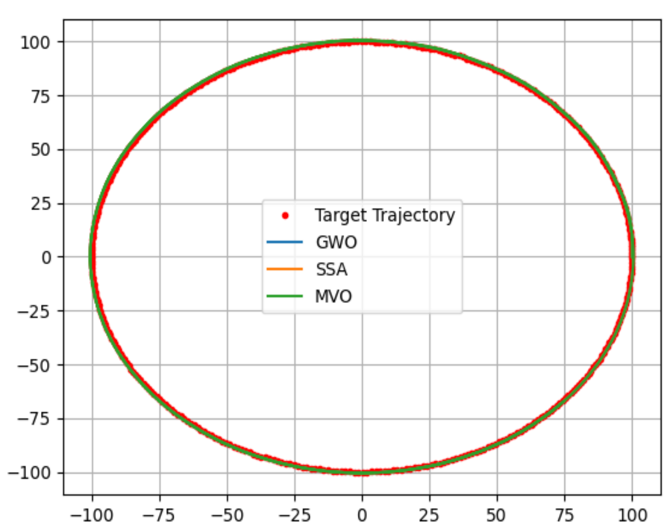


Fig. 5. Trajectory results for example 1

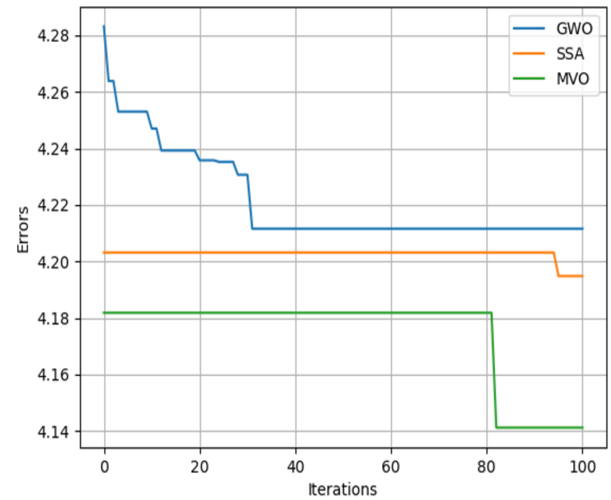


Fig. 6. Convergence graphs of the algorithms for example 2

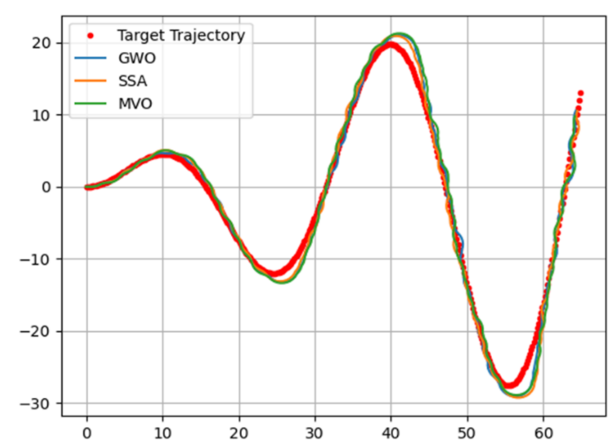


Fig.7. Trajectory results for example 2

Table 5. Comparative results of the algorithms for example 2 (Error data is in units.)

	GWO	SSA	MVO
Best	4.194	4.194	4.141
Mean	4.211	4.207	4.158
Std	0.012	0.008	0.021
Run time (sec)	22.99	22.67	23.08

Table 6. Best found parameters of the algorithms for example 2

	GWO	SSA	MVO
K_p	1.153	1.115	1.262
K_i	4.001e-4	0.001	1.102e-4
K_d	0.151	0.152	0.225

6. Conclusion

In the context of autonomous driving trajectory tracking, the amalgamation of evolutionary algorithms for the refinement of PID controller hyperparameters has yielded substantial insights. In this study, we embarked on a journey to explore the synergy between evolutionary algorithms and the PID controller in the context of vehicular trajectory tracking for autonomous vehicles. By harnessing the power of three distinct evolutionary algorithms, namely GWO, MVO and SSA, we aimed to calibrate the hyperparameters of the PID controller to enhance its performance. Notably, this endeavor represents a pioneering effort, marking the first deployment of these evolutionary optimizers for PID tuning within the intricate domain of trajectory tracking. Our research revealed profound contributions on two primary fronts. Firstly, it established a novel precedent in the field by introducing the GWO, MVO, and SSA algorithms as potent tools for discerning optimal PID hyperparameters. This novel application underscores the originality of our approach, offering fresh insights into the realm of autonomous vehicle control. Secondly, our study conducted a thorough and rigorous assessment of the algorithmic performance, combining quantitative analysis with qualitative graphical representations. This comprehensive evaluation showcased not only the practical effectiveness of the employed algorithms but also laid the foundation for potential future explorations and advancements in this burgeoning field.

Acknowledgment

The author thanks the KARSAN Automotive, R&D Department for providing high-performance computing resources.

Conflict of Interest Statement

The author declare that there is no conflict of interest in the study.

References

[1] Farag W. Complex Trajectory Tracking Using PID Control for Autonomous Driving. Int J Intell Transp Syst Res. 2020; 18(2): 356-366. https://doi.org/10.1007/s13177-019-00204-2

[2] Farag W, Saleh Z. Tuning of PID Track Followers for Autonomous Driving. Presented at International Conference on Innovation and Intelligence for Informatics Computing, and Technologies; 2018; Sakhier, Bahrain. 10.1109/3ICT.2018.8855773

[3] De Luca A, Oriol G, Samson C. Feedback Control of a Nonholonomic Car-like Robot. Robot Motion Planning and Control. 2005; 171-253.

[4] Sebastian B, Ben-Tzvi P. Physics Based Path Planning for Autonomous Tracked Vehicle in Challenging Terrain. J Intell Robot Syst. 2019; 95: 511-526. https://doi.org/10.1007/s10846-018-0851-3

[5] Hu J, Tao J, Zhao W, Han Y. Modeling and Simulation of Steering Control Strategy for Dual-motor Coupling Drive Tracked Vehicle. J Braz Soc Mech Sci Eng. 2019; 41: 1-11. https://doi.org/10.1007/s40430-019-1692-0

[6] Zhai L, Sun TM, Wang QN, Wang J. Lateral Stability Control of Dynamic Steering for Dual Motor Drive High Speed Tracked Vehicle. Int J Automot Technol. 2016; 17: 1079-1090. https://doi.org/10.1007/s12239-016-0105-y

[7] Chu D, Li H, Zhao C, Zhou T. Trajectory Tracking of Autonomous Vehicle Based on Model Predictive Control with PID Feedback. IEEE trans Intell Transp Syst. 2022; 24(2): 2239-2250. https://doi.org/10.1109/TITS.2022.3150365

[8] Tan KK, Qing-Guo W, Chieh HC. Advances in PID Control. Springer-Verlag; 1999.

[9] Kuyu YÇ, Vatansever F. Advanced Metaheuristic Algorithms on Solving Multimodal Functions: Experimental Analyses and Performance Evaluations. Arch Comput Methods Eng. 2021; 28: 4861–4873. https://doi.org/10.1007/s11831-021-09555-0

[10] Kuyu YÇ, Vatansever F. GOZDE: A Novel Metaheuristic Algorithm for Global Optimization. Future Gener Comput Syst. 2022; 136: 128-152. https://doi.org/10.1016/j.future.2022.05.022

[11] Yu T, Zhu H. Hyper-parameter optimization: A review of algorithms and applications. arXiv preprint 2020; arXiv: 2003.05689. https://doi.org/10.48550/arXiv.2003.05689

[12] Kuyu, YÇ, Onieva E, Lopez-Garcia P. A Hybrid Optimizer Based on Backtracking Search and Differential Evolution for Continuous Optimization. J Exp Theor Artif Intell. 2022; 34(3): 355-385. https://doi.org/10.1080/0952813X.2021.1872109

[13] P Zhao, J Chen, Y Song, X Tao, T Xu, T Mei. Design of a Control System for an Autonomous Vehicle Based on Adaptive-PID. Int J Adv Robot Syst. 2012; 9(2): 44. https://doi.org/10.5772/51314

[14] Mirjalili S, Mirjalili SM, Lewis A. Grey Wolf Optimizer. Adv Eng Softw. 2014; 69: 46-61. https://doi.org/10.1016/j.advengsoft.2013.12.007

[15] Mirjalili S, Mirjalili SM, Hatamlou A. Multi-verse Optimizer: A Nature-inspired Algorithm for Global Optimization. Neural Comput Appl. 2016; 27: 495-513. https://doi.org/10.1007/s00521-015-1870-7

[16] Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM. Salp Swarm Algorithm: A Bio-inspired Optimizer for Engineering Design Problems. Adv Eng Softw. 2017; 114: 163-191. https://doi.org/10.1016/j.advengsoft.2017.07.002

[17] Barzegar A, Doukhi O, Lee DJ. Design and Implementation of an Autonomous Electric Vehicle for Self-driving Control Under GNSS-denied Environments. Appl Sci. 2021; 11(8): 3688. https://doi.org/10.3390/app11083688

[18] Hanilçi F, Kuyu YÇ, Vatansever F. Optimizing Autonomous Path Tracking Using Population-Based Metaheuristic and Pure Pursuit Control. Presented at 14th International Conference on Electrical and Electronics Engineering. 2023; Bursa, Türkiye. https://doi.org/10.1109/ELECO60389.2023.10415995

[19] BaniHani S, De S. Development of a Genetic Algorithm-based Lookup Table Approach for Efficient Numerical Integration in the Method of Finite Spheres with Application to the Solution of Thin Beam and Plate Problems. Int J Numer Methods Eng. 2006; 67(12): 1700-1729. https://doi.org/10.1002/nme.1678

[20] Xu Z, Zhao W, Wang C. Local Path Planning and Tracking Control of Vehicle Collision Avoidance System. Trans Nanjing Univ Aeronaut. 2018; 35(4): 729–738. http://dx.doi.org/10.16356/j.1005-

1120.2018.04.729

[21]Johnson MA, Moradi MH. PID Control. London, UK: Springer-Verlag London Limited; 2005.

[22]Vatansever F, Hacıiskenderoğlu E. PID Tuning with Up-to-date Metaheuristic Algorithms. Uludağ University Journal of the Faculty of Engineering. 2022; 27(2): 573-584. https://doi.org/10.17482/uumfd.1090766

[23]Vatansever F, Sen D. Design of PID Controller Simulator Based on Genetic Algorithm. Uludağ University Journal of The Faculty of Engineering. 2013; 18(2): 7-18. https://doi.org/10.17482/uujfe.33406