Journal of Industrial Information Integration 9 (2018) 45–52

Contents lists available at ScienceDirect

Journal of Industrial Information Integration

journal homepage: www.elsevier.com/locate/jii

Antlion optimizer tuned PID controller based on Bode ideal transfer function for automobile cruise control system

Rosy Pradhan[a], Santosh Kumar Majhi[b,*], Jatin Ku Pradhan[a], Bibhuti Bhusan Pati[a]

[a] Department of Electrical Engineering, VSS University of Technology, Burla, India
[b] Department of Computer Science & Engineering, VSS University of Technology, Burla, India

ARTICLE INFO

Keywords:
PID controller
Ant lion optimizer
Cruise control system
Time domain analysis
Frequency domain analysis
Robustness analysis
Bodes ideal model

ABSTRACT

Automobile cruise control reduces driver fatigue and ensures traffic safety by heavily relying on ICT and control engineering. A number of cruise control schemes are expected to be designed for proper safety management of vehicles in automobile design sector. Due to boosting road crowds, there is a new demand in developing automation of cruise control. The successfulness of automation of cruise control is heavily dependent on the efficient design of feedback control system. The use of evolutionary algorithms in designing feedback control of automobile cruise control system has reported in recent literature. It involves the communication of traffic data captured by sensor with actuator of the vehicle. The literature still lacks use of recently developed evolutionary algorithms for cruise control that could leverage road safety by tuning the control parameters. This paper presents the design of robust PID controller for an automobile cruise control system (ACCS). A linearized version of the cruise control system has been studies as per the dominant characteristics in closed loop system. The PID controller is designed as per Bode ideal transfer function to ensure robustness and formulated as an optimization problem. The gain parameters of the designed PID controller are tuned by ant lion optimizer (ALO) algorithm such that the proposed compensated system depicts the Bode ideal reference model. A comparison of this approach, a state space method, classical PID, fuzzy logic, genetic algorithm is presented. Numerical simulations shows that the ALO tuned PID controller based on bodes ideal model has better performance in terms of settling time, rise time, maximum overshot, peak time and steady state error. The performance analysis has performed by time domain analysis and frequency domain analysis. The robustness of the system is evaluated by considering sensitivity, complimentary sensitivity and controller sensitivity function. Further, the disturbance rejection behavior is studied for the proposed system. The analysis result reveals that the proposed ALO based PID controller with Bode ideal model for ACCS performs better than other recently published methods. All the simulations are carried out in Simulink/Matlab environment.

1. Introduction

Automobile cruise control system (ACCS) is designed to reduce the drivers fatigue in the long run drive. In addition, now-a-days traffic safety is given a priority and primary concern in the automotive research area. In that perspective, cruise control permits the driver to manage the vehicle speed and when ACCS is turn on, the speed of the automobile is upheld automatically without the application of the accelerator pedal. In turn the probability of potential crash is minimized and the safety is maintained. Thus the regulation of the velocity plays a crucial role in ACCS. In addition to that, the reaction time of the ACCS is considered very critical due to the velocity variation of the system which can be decreased by designing a suitable controller. Currently, a number of features are added due to the revolution of recent technologies e.g., all the control operations such as the speed control, the attainment of last run speed, deactivate the speed after brakes etc. by pressing the buttons [1]. Moreover these features are common in automotive vehicles. The automobile vendors are moving towards designing of automatic vehicles. ACCS is used to provide safety, traffic fluency and also reduces fuel consumption [2,23]. Moreover, cruise control system helps in avoiding collisions between vehicles, better traffic management, reducing travel time and lower consumption of fuel by maintaining desired speed [3,22]. These features of cruise control system can be achieved by the integration of information of the moving objects in front of the monitoring sensor and current speed of the vehicle. By taking this integral information the controller initiates necessary control action to throttle and brake systems of the vehicle to control the speed of the vehicle. Basically, this system integrates ICT,

* Corresponding author.
E-mail addresses: rosy.pradhan88@gmail.com (R. Pradhan), smajhi_cse@vssut.ac.in (S.K. Majhi), jkp10@iitbbs.ac.in (J.K. Pradhan), pati_bibhuti@rediffmail.com (B.B. Pati).

https://doi.org/10.1016/j.jii.2018.01.002
Received 23 February 2017; Received in revised form 3 January 2018; Accepted 3 January 2018
Available online 04 January 2018
2452-414X/

control engineering and automated systems. This kind of integration is highlighted in reference [24]. Moreover, the ACCS falls in the transportation sub-classification as per the classifications defined in [24]. In addition, considerable research attempts have been made on many real time systems such as the surveillance and diagnosis of a gas turbine system [26], manufacturing grid based virtual enterprise operation platform [25], with the approach of integrating information technologies and intelligent computing methods. In this work, the research efforts have been made to integrate the intelligent computing technology for cruise control system. In that perspective, various control methods applied to cruise control system is found in literature [4–9,17–21] including conventional PID, state space, fuzzy logic, genetic algorithm (GA). In these control methods the objective is to find the values of proportional (K_p), integral (K_i) and derivatives (K_d) gain by optimizing the unit feedback function. As ACCS system is a complex integration of information in cruise control system, the controller tuning process is a challenging job to achieve optimal performance of the system. Thus, researchers have applied intelligent techniques such as GA and fuzzy logic to obtain optimal result. Moreover, these algorithms look for global optima of the function by the collective measures of search agents and deciding parameters in the search space. Therefore, the exploration and exploitation is an important performance criterion of the optimization methods. In addition, one major factor of the optimization algorithms lies in the randomness; even if randomness results in different values in each run, still the avoidance of local minima is difficult. Moreover, some of the algorithms like PSO, DE, GA etc. suffer from premature convergence. The premature convergence has been resolved in some advanced variants along with the hybridization of the algorithms. Some recent algorithms such as GWO, ABC etc. are having different operations for exploitation and exploration. Antlion Optimizer (ALO) algorithm is such type of meta-heuristic algorithm proposed by Mirjalili (2015), where the exploration through global search and the exploitation through local search presented in one algorithm. ALO algorithm has high level of exploration which leads to explore the potential regions of the search space. High exploitation assists the ALO algorithm to rapidly converge towards the optimum and exploit it accurately. The ALO has better performance which has been verified by applying 19 benchmark functions. The important factor of choosing ALO is due to its effective search space using random walk and selection of search agents randomly. Moreover, exploitation of the search space is secured by adjustive limits of traps.

Owing to above advantage offered, the ALO is used to optimize the regulatory parameter as per the specification of the system. The ALO tuned PID controller has used for the first time for ACCS, which performs comparatively better than other reported methods in literature [13–16]. This work is focused to optimize the ALO-PID controller parameters to obtain the robust performance for the cruise control system using bodes ideal transfer function. Here, the open loop transfer function of the proposed system is tuned such that it achieves the performance of the Bode ideal transfer function.

The rest of the paper is organized as follows. Section 2 presents the description of the non linear cruise control model and its linearized version. Section 3 presents the brief overview of the ant lion optimizer optimization to make it self-content. In Section 4, result and discussion are presented considering application of ALO algorithm, effect of objective function, transient analysis, robustness analysis (bode analysis, complimentary sensitivity and controller sensitivity) and root locus analysis. Finally, this paper concludes briefly in Section 5.

2. Description of automobile cruise control system (ACCS)

The ACCS is used to regulate the vehicle's speed according to the driver's reference command. The schematic block diagram of cruise control system considered in this paper is shown in Fig. 1. Here the cruise control system generates the desired amount of throttle input u so as to maintain the constant velocity (V) i.e. to follow the reference velocity V_{ref}. The system dynamics is presented in Fig. 2. The nonlinear longitudinal dynamics of the vehicle may be written as [9]

$$F_d = M\frac{dV}{dt} + F_a + F_g \tag{1}$$

Where M is mass of vehicle and passenger, V is velocity of the vehicle, $F_d = C_a(V - V_w)^2$ is an aerodynamic drag, C_a is aerodynamic drag coefficient, V_w is wind gust speed, $F_g = Mg\sin\theta$ is climbing resistance or downgrade force, θ is road inclination, F_a is aerodynamics drag and g is the gravitational acceleration.

Furthermore, the actuator of cruise control system is modeled as first order lag system i.e. $\frac{C_1 e^{\tau s}}{Ts+1}$ with saturation block having saturation limit F_{dmin} and F_{dmax}, where C_1 is the actuator constant, T is the time of observation and τ is the reaction time of the driver. The aerodynamic drag can be written as

$$F_d = \frac{C_1 e^{\tau s}}{Ts + 1} \tag{2}$$

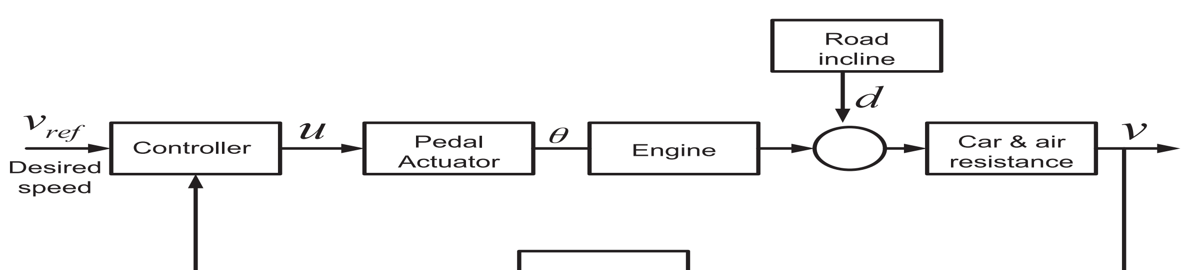


Fig. 1. Block diagram of cruise control system [9].

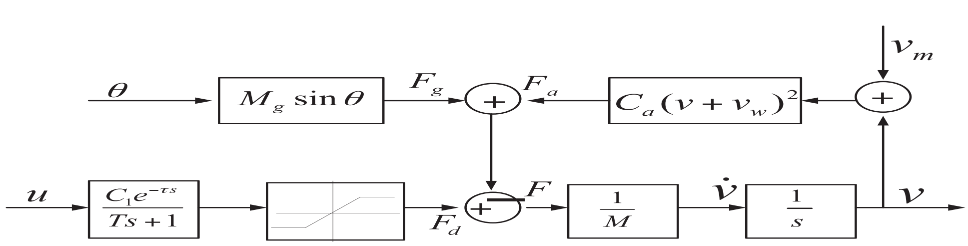


Fig. 2. Dynamic model of automobile [9].

2.1. Linearized model of cruise control system

For making the design of controller easy, one must have linearized model for the vehicle system to start with. By considering all initial conditions and disturbances i.e. $V_w = 0$ *and* $\theta = 0$, the nonlinear Eqs. (1) and (2) can be converted to

$$\dot{V} = \frac{1}{M}(F_d - C_a V^2) \tag{3}$$

$$\dot{F_d} = \frac{1}{T}(C_1 u(t-T) - F_d) \tag{4}$$

However, from the above two equations, it is observed that the non-linearity still exists due to the quadratic term in Eq. (3). The non-linearity is eliminated by using Taylor's series expansion. Let the non-linear Eqs. (3) and (4) be represented as

$$\dot{x} = f(x, u),\ \ y = h(x, u) \tag{5}$$

Where x is the state vector defined as $x = [V \ \ F_d]^T$, u is the control input vector, y is the output vector which is equal to the velocity. Let $x = x^0 + \Delta x,\ u = u^0 + \Delta u$, where (x^0, u^0) is the equilibrium or operating point and Δx and Δu is the small deviation around the operating point. The linearized system then becomes

$$\dot{\Delta} x = A\Delta x + B\Delta u(t-T) \tag{6}$$

$$y = C\Delta x \tag{7}$$

Where

$$\mathrm{A} = \left.\frac{\partial f}{\partial x}\right|_{x^0,u^0} = \begin{bmatrix} -\frac{2C_a V^0}{M} & \frac{1}{M} \\ 0 & \frac{-1}{T} \end{bmatrix},\ \mathrm{B} = \left.\frac{\partial f}{\partial u}\right|_{x^0,u^0} = \begin{bmatrix} 0 \\ \frac{C_1}{T} \end{bmatrix} \text{and } C = \left.\frac{\partial h}{\partial x}\right|_{x^0,u^0} = [1\ 0].$$

The transfer function $G(s)$ can be directly obtained from the above state space matrices (6) and (7) as

$$G(s) = \frac{\Delta V(s)}{\Delta U(s)} = \frac{\frac{C_1 e^{-\tau s}}{MT}}{\left(s + \frac{2C_a v}{M}\right)\left(s + \frac{1}{T}\right)} \tag{8}$$

Now, with the operating point $v =$ 30 km/hr and from Table 1, the value of system matrices and the plant transfer function G (s) are

$$\mathrm{A} = \begin{bmatrix} -0.0476 & 0.00067 \\ 0 & -1 \end{bmatrix};\ \ \mathrm{B} = \begin{bmatrix} 0 \\ 743 \end{bmatrix};\ \ C = [1\ 0]$$

$$\mathrm{G(s)} = \frac{\Delta \mathrm{V(s)}}{\Delta \mathrm{U(s)}} = \frac{2.4767}{(s + 0.0476)(s+1)(s+5)}. \tag{9}$$

3. Tuning of PID using Bode reference model

In this paper model reference control is used to update the parameters of the proposed controller. Thus, the selection of the reference model affects the performance of the designed closed loop system. In order to design a robust PID control i.e., insensitive to gain changes, here, Bode ideal transfer function is used as a reference model [10–12]. The open loop transfer function of the Bode ideal transfer function is given by Eq. (10)

$$L(s) = \left(\frac{w_c}{s}\right)^\gamma,\ \ \gamma \in \mathscr{R} \tag{10}$$

Where w_c is the gain crossover frequency and γ is the slope of the magnitude curve. It is well known that, the Bode plot of $L(s)$ for $\gamma = 1$ will be a straight line with constant slope of -20db/dec, which will ensure good disturbance rejection response, however, it suffers from poor noise rejection capability. Thus to ensure both good disturbance and noise rejection property range of γ is chosen as $1 < \gamma < 2$, i.e., with variation of γ, the roll of rate of Bode ideal transfer function can vary between -20db/dec to -40db/dec which ensure robust design. Further, the obtained phase margin for the Bode reference model becomes $PM = \pi(1 - \frac{\gamma}{2})$ rad. The closed loop transfer function of the reference model becomes

$$T(s) = \frac{L(s)}{1 + L(s)} = \frac{1}{\left(\frac{s}{w_c}\right)^\gamma + 1} \tag{11}$$

3.1. Selection of γ value and the Bode ideal reference model

As discussed in the above section, the selection of γ value affects the noise rejection capability of the system. Therefore, the choice of γ plays a crucial role in design of robust PID controls. Here, one choice of γ is briefly presented below.

Let, the controller is designed to achieve $PM = 45°$ for a gain cross over frequency $w_c =$ 1 rad/sec (say). Then the value of γ will be 1.5. Thus the loop transfer function of the reference system becomes

$$L(s) = \left(\frac{w_c}{s}\right)^\gamma = \left(\frac{1}{s}\right)^{1.5}$$

4. Overview of Ant Lion Optimizer

Ant Lion Optimization technique is the one of the nature inspired optimization algorithm for solving the uni-dimensional as well as multidimensional optimization problem. This algorithm was proposed by Mirjalili (2015) [13]. This algorithm is encouraged by the hunting activities of the grey antlions in nature and basically their favorite preys are ants shown in Fig. 3. A cone shaped pit is created by the antlions and they hides its larvae under beneath the bottom of cone shaped pit to trap the ant. The edge of cone is made sharp such that the ant can easily falls into the bottom. Once the preys trapped into the cone, then the antlion throws the sand towards the edge of the cone which makes the prey incapable to escape from the trap. After that the antlion consumes the prey and prepares another pit to trap next prey. The above described hunting nature of antlion is explained in five stages such as the random walk of ants, building traps, entrapment of ants in traps, catching preys, and re-building traps. As ALO mimics the hunting activity of antlion, so this optimization governed by certain conditions [13]

1. Ants move in random walks about the search space.
2. The traps of the antlions are influenced by random walks.
3. Antlions can construct their pit according to their fitness value. The fitness value determines the pit size.
4. Antlion with higher fitness value i.e. larger pit have maximum chance to catch the ants.
5. In each iteration, an ant is caught by a fittest antlion.
6. In order to trigger the sliding behavior of ants towards antlions, the assortment of random walk of ant is adjusted in decreasing order.
7. If the fitness value of ant is more than that of an antlion, which means that it is captured by the antlion.

Table 1
Model parameters considered [9].

Symbol	Value	Units
C_1	743	–
C_a	1.19	$N/(m/sec)^2$
M	1500	Kg
τ	0.2	sec
T	1	Sec
F_{dmax}	3500	N
F_{dmin}	−3500	N
g	9.8	m/sec^2

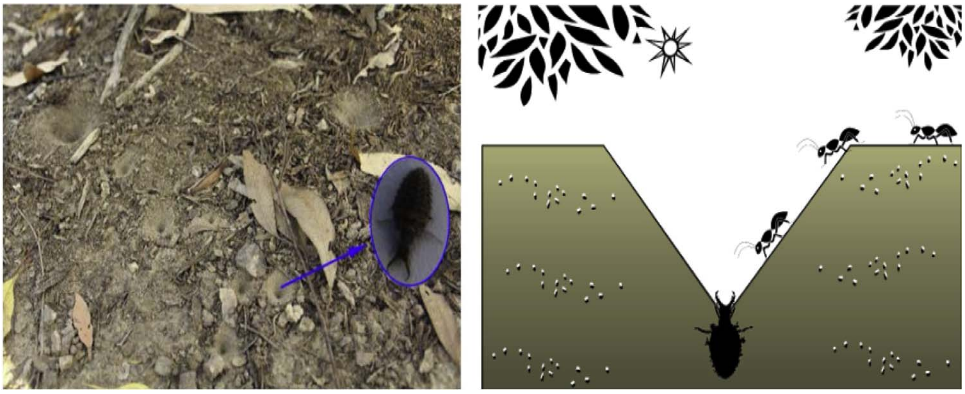


Fig. 3. Hunting nature of antlion [13].

8. After each hunt step, the position is changed by antlion and they build a new improved pit for trapping another prey.

The five major steps for the optimization method and their mathematical descriptions are mentioned below [10]

1. Random walks of ants

To model the hunting behavior of the antlion, the antlion and ant should have interaction with each other. For this the ants are required to move search space for food and shelter and antlions are hunted them by using their traps. Because of the stochastic movement of ants for searching of foods, a random walk is chosen for modeling of ant movements as follows,

$$X(t) = [0, cusum(2r(t_1) - 1), \cdots cusum(2r(t_n) - 1) \tag{12}$$

Where cusum calculates the cumulative sum, n indicates the maximum numbers of iteration; t is the steps of random walk and $r(t)$ is the stochastics function defined as below

$$r(t) = \begin{cases} 1 \; if \;\; rand > 0.5 \\ 0 \; if \;\; rand \leq 0.5 \end{cases} \tag{13}$$

Where rand is a random number generated with uniform distribution in the interval 0 to 1.

The mathematical representation of normalized random walk of ant is given by following equation

$$X_i(t) = \frac{(X_i^t - a_i) \times (d_i^t - C_i^t)}{(b_i - a_i)} + C_i^t \; [10] \tag{14}$$

Where a_i is the minimum of random walk of ith variable, b_i indicates the maximum of random walk of the ith variable, C_i^t is the minimum of the ith variable at tth iteration and similarly d_i^t is the maximum of the ith variable at tth iteration.

2. Building trap

The mathematical modeling of antlion's hunting capability is influenced by a roulette wheel. The ALO algorithm utilizes the roulette wheel for searching the fittest antlion during the optimization process. This process filters the best antlion with higher probability for catching the prey.

3. Entrapments of ants in traps

As discussed above, the random walk of ants are influenced by the traps of the antlions. Therefore the mathematical relationship for this assumption is expressed by the following equation

$$C_i^t = Antlion_j^t + C^t \tag{15}$$

$$d_i^t = Antlion_j^t + d^t \tag{16}$$

Where vector C and d are the hypersphere of randomly walked ant and around the selected antlion respectively. $C^t$and d^t indicates the minimum and maximum value of all the variable at tth iteration respectively. Similarly C_i^t and d_i^t are minimum and maximum of the ith variable at tth iteration respectively.

4. Sliding ants towards the antlion

Once the antlion realized that the ant is in the trap, it throw the sand toward the edge of the pit through the its mouth which make the ant trappes inside the pit. This behavior of antlion is model by the following equation

$$C^t = \frac{C^t}{I} \tag{17}$$

$$d^t = \frac{d^t}{I} \tag{18}$$

Where I indicates the ratio described in Eq. (15), C^t and d^t are the minimum and maximum values of all the variables at tth iteration respectively.

$$I = 10^w \frac{t}{T} \tag{19}$$

Where t is the current iteration indicates total number of iteration and w is the constant depend on the current iteration. The above two mathematical equations represent the sliding process of ant into the pit.

5. Catching the prey and rebuild the trap

The final stage of hunt occurred when the ant is caught by the antlion. This behavior of antlion is described by the Eq. (20) where the fitness value of ant is more than the fitness value of antlion. In this situation ant is consumed by the antlion. Then the antlion is updated with its position or build a new trap to catch a new prey.

$$Antlion_j^t = Ant_i^t \;\; if \; f(Ant_i^t) > f(Antlion_j^t) \tag{20}$$

Where t is the current iteration, $Antlion_j^t$ indicates the position of the selected jth antlion at tth iteration and Ant_i^t shows the position of ith ant at tth iteration. Function $f()$ indicates the fitness value of ant and antlion.

6. Elitism

Elitism is the important characteristics which allow the algorithm to obtain best solution at every stage of the optimization process. In ALO algorithm, the best antlion is obtained and saved as a elite in each iteration. Since the elite one is influenced the movement of all the ants during the iterations. Therefore, it is considered that every ant moved randomly around the selected antlion and elite simultaneously by the given equations

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{21}$$

Where R_A^t is the random walk around the selected antlion at tth iteration, R_E^t indicates the random walk around the elite tth iteration and Ant_i^t shows the position of ith ant at tth iteration. The steps for the algorithm are given in Table 2.

The ALO has shown high performance in solving the classical optimization problem. ALO converges rapidly towards the optimum with the help of exploitation. It has algorithm has a high level of exploration which assists it to explore the promising regions of the search space.

Table 2
Main steps of ALO algorithm.

Input: Number of search agents, maximum iteration
Output: Best Score and Best Positions
Initialize the ant and antlion population randomly
Calculation of fitness of ant and lion and determination of elite (E)
While (t < maximum iteration)
For each ant(search agent)
  Select an antlion based on Roulette wheel
  Update the minimum value and maximum value of t^{th} iteration
  Random walk creation and normalization based on Min-Max normalization
  Update the position of ants
End for
Calculate the fitness of all search agents
Replace an fitter ant lion
Update elite if antlion is better than elite
t = t + 1
End While
Return E

Furthermore, the local optima avoidance of this algorithm is satisfactory since it is able to avoid all of the local optima and approach the global optima. The algorithm provides better competitive results in comparison with PSO, DE and other meta-heuristic algorithms [13–16].

5. Implementation and analysis

5.1. Problem formulation

An ACCS system with PID controller based on Bode's ideal transfer function is shown in Fig. 4. The gains of the PID controller are tuned by ALO algorithm. The self tuning PID is considered as controller for the ACCS system because of its proven advantages. The classical PID, state space, GA, Fuzzy Logic and ALO algorithms were used for the tuning of PID controller parameters. During the evolutionary algorithms, the upper and lower bounds of the gains are chosen as $\mathbf{3 < K_p < 4, 0.1 < K_i < 0.25, 3 < K_d < 4}$ [8]. The number of iterations and search agent for ALO algorithm are 150 and 10 respectively. For the design of controller, there is various performance indices present in the literature. The most commonly used indices are Integral Absolute Error (IAE), Integral Square Error (ISE), Integral of Time Multiplied Absolute Error (ITAE) and Integral of Time Multiplied Square Error (ITSE). The objective functions are represented by the below Eqs. (22)–(25). Optimum values of the controller can be calculated by minimizing the following optimizing functions. The objective function is chosen for minimizing the time response characteristics due to the dependency of error on time.

(a) Integral Absolute Error (IAE)

$$J_1 = \text{IAE} = \int_0^\infty |e(t)|dt \tag{22}$$

(b) Integral Square Error (ISE)

$$J_2 = \text{ISE} = \int_0^\infty |e^2(t)|dt \tag{23}$$

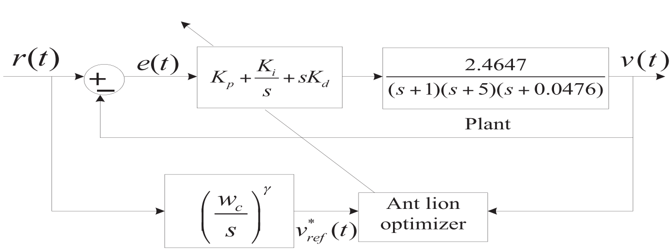


Fig. 4. The block diagram of an ACCS system ALO tune PID controller.

(c) Integral of Time multiplied Absolute Error (ITAE)

$$J_3 = \text{ITAE} = \int_0^\infty t|e(t)|dt \tag{24}$$

(d) Integral of time multiplied Square Error (ITSE)

$$J_4 = ITSE = \int_0^\infty t|e^2(t)|dt \tag{25}$$

The optimization problem can be formulated as

Minimize J
Subjected to

$$\begin{aligned} K_{pmin} < K_p < K_{pmax} \\ K_{imin} < K_i < K_{imax} \\ K_{dmin} < K_d < K_{dmax} \end{aligned} \tag{26}$$

Where, J is the objective function (J_1, J_2, J_3 *and* J_4)

In this paper, the ISE is chosen as a performance criterion for the ACCS. To get the error $e(t)$, it is necessary to select the difference between reference model and actual model. As small is the error as close to the controller parameters.

$$J(K_p, K_i, K_d) = ISE = \int_0^\infty |v_{ref}(t)^* - v(t)|dt \tag{27}$$

Where $v_{ref}(t)^*$ is the output response of the reference model and $v(t)$ is the output response of actual model. Therefore, the optimal PID controller parameter can be calculated by minimizing objective function.

From the various performance indices, the optimal gains of the controller obtained are presented next section.

5.2. Proposed ALO-based PID parameter design

In order to design an optimal PID gains, here ISE objective function is used. Optimizing the ISE function through ALO algorithm based on Bode ideal transfer function, the PID controller parameters are found to be $K_p = 3.858$, $K_i = 0.2089$ *and* $K_d = 3.8581$. Therefore, the closed loop transfer function of the ACC system with the optimal PID parameter is given by

$$\frac{v(s)}{r(s)} = \frac{9.509s^2 + 9.509s + 0.5149}{s^4 + 6.048s^3 + 14.79s^2 + 9.747s + 0.5149} \tag{28}$$

5.3. Comparison with existing result

A comparative analysis of proposed ALO-PID algorithm with other recent published methods such as state space, fuzzy logic, genetic algorithm (GA) and conventional PID techniques [8] are shown in Table 3. The comparative analysis are based on the transient performance, e.g., rise time (T_r), settling time (T_s), peak time (T_p), maximum overshoot (% M_p) and steady state performance i.e. steady state error (E_{ss}). It is clear from Table 4 that with proposed ALO optimized based on Bode ideal model for PID controller shows better result as compared to GA, Fuzzy Logic, state space, and conventional PID methods in terms

Table 3
Various tuning methods with performance parameter values.

Control Methods in Cruise Control System	T_r (sec)	T_s(sec)	$\%M_p$	T_p(sec)	E_{ss}
PID [8]	1.7	5.5	10.2	3.54	0.01
State Space [8]	1.38	3	10	2.97	0.01
Fuzzy Logic [8]	2.21	3.37	1.91	3.16	0.01
GA [8]	0.945	2.15	1.14	1.46	0
ALO-BODE-ISE	0.8094	1.732	0.612	1.26	0

Table 4
Root locus analysis of ALO-PID ACCS.

ALO-PID	
Closed loop poles	Damping Ratio
–2.5 + 1.95i	**0.789**
–2.5–1.95i	**0.789**
–100	1
–0.892	1

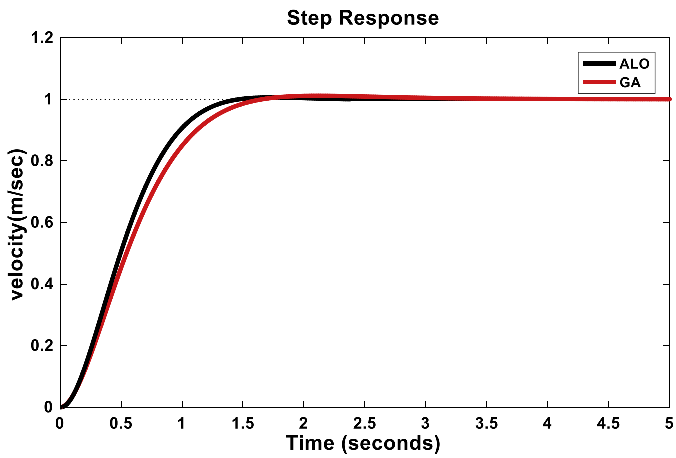


Fig. 5. The step response of the Cruise Control System for different algorithms.

of transient and steady state performance. The terminal velocity change is given in Fig. 5 for GA and ALO tuned PID controller.

5.4. Root locus analysis

Fig. 6 shows the root locus curve for cruise control system tuned by proposed ALO algorithm. The values for closed loop and damping ratio of ACC system tuned by ALO are given in Table 4. From Table 4, it has been observed that the closed loop poles reside to the left half of the s-plane. Therefore we can say the closed loop system is stable. It clear from the results depicted in the Table 4 that the conjugate poles obtained from ALO algorithm are more left on the s-plane and highest damping ratio.

5.5. Bode analysis

The frequency response analysis by using Bode plot for proposed ALO tuned PID for the ACC system is shown in Fig. 7. In Table 5, the bandwidth, delay margin, phase margin and peak gain are presented for

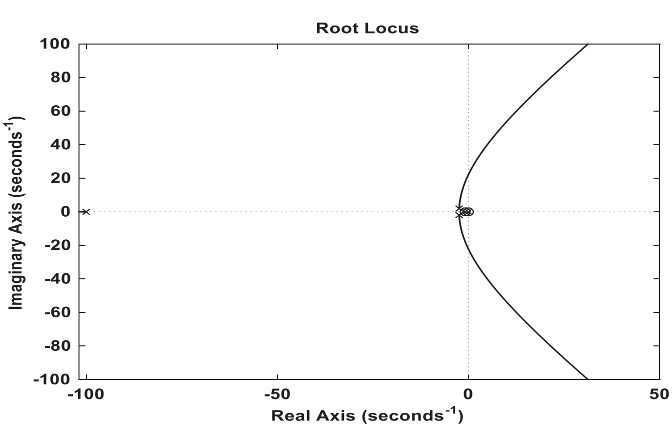


Fig. 6. Root locus curve of the ALO-PID cruise control system.

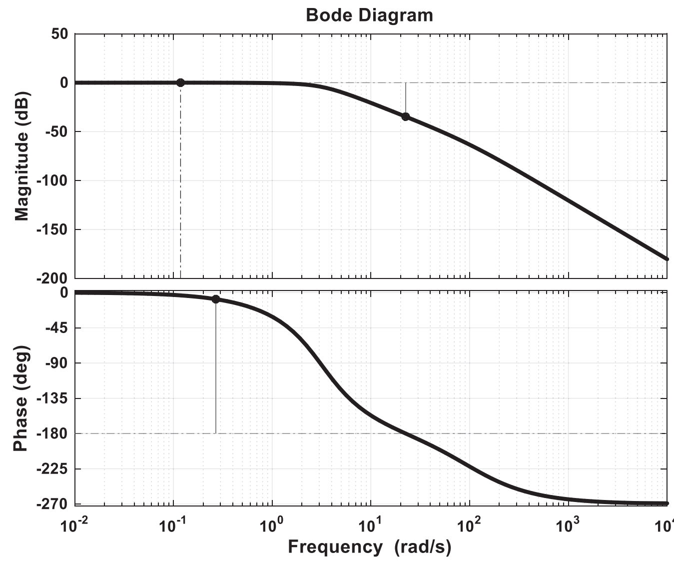


Fig. 7. Bode plot of the ALO-PID ACCS.

Table 5
Bode analysis.

Algorithms	Peak gain (dB)	Phase margin (deg)	Gain margin (s)	Bandwidth
BODE-ALO	**0.0301**	**172**	**34.7**	**2.59**
GA[6]	NA	71.5	NA	NA

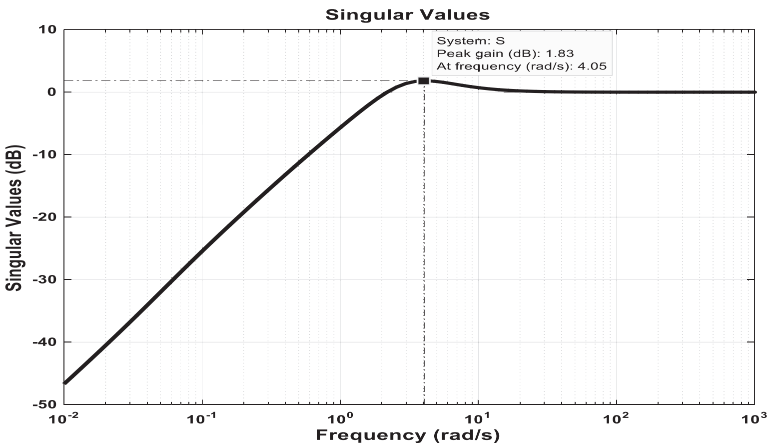
the ALO algorithm. In addition, the values have been compared with GA [8]. From Bode plot, the minimum peak gain, maximum phase margin, delay margin and bandwidth are obtained. Therefore we conclude that, the ALO algorithm results the best frequency response (Fig. 8).

5.6. Robustness analysis

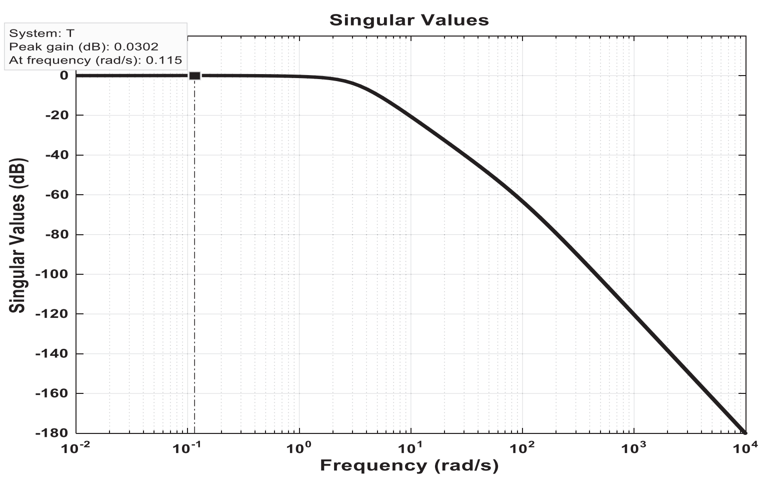
The robustness of a system is better captured by the singular value plots of the following transfer function: Sensitivity function $(s) = \frac{1}{1+L(s)}$, Complementary function $T(s) = \frac{L(s)}{1+L(s)}$, Controller sensitivity $K(s)S(s)$, where $L(s) = G(s)K(s)$ is the loop transfer function and $K(s)$ is the PID controller. For the better robustness, the peak of the above transfer functions i.e. $||S||_\infty$, $||T||_\infty$ should be as small as possible (less than 2 or 6 dB) and at the same time, the gain of the sensitivity function should be less in low frequency region and for complementary sensitivity function, the gain should be less at high frequency region. As shown in Fig. 9(a), (b), (c) the peak of the sensitivity is **1.23**, complementary sensitivity is **1.0035** and there is no peak (within the frequency range 10^1 to 10^3 rad/sec) in controller sensitivity plot. Therefore from the above parameter values, we can conclude that the close loop system is robust against any disturbances such as input output disturbances, parametric uncertainty etc.

5.7. Disturbance rejection

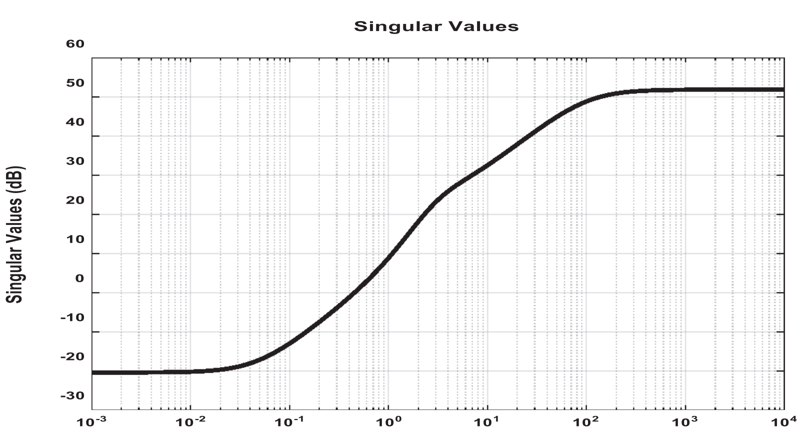
To see the disturbance rejection behavior of the proposed controller, an impulse signal of magnitude 20% of reference is added at the output. The disturbance is added at time $t = 3$ sec and it persists up to $t = 3.05$ sec. Fig. 9 shows the disturbance rejection response for the cruise control system. It is seen from Fig. 9 that the proposed controller brings back the system to its original position very quickly which ensure the disturbance rejection behavior of the proposed compensated system.



(a) Sensitivity plot for ACCS



(b) Complementary sensitivity plot for ACCS



(c) Controller sensitivity plot for ACCS

Fig. 8. Different sensitivity plots for cruise control system.

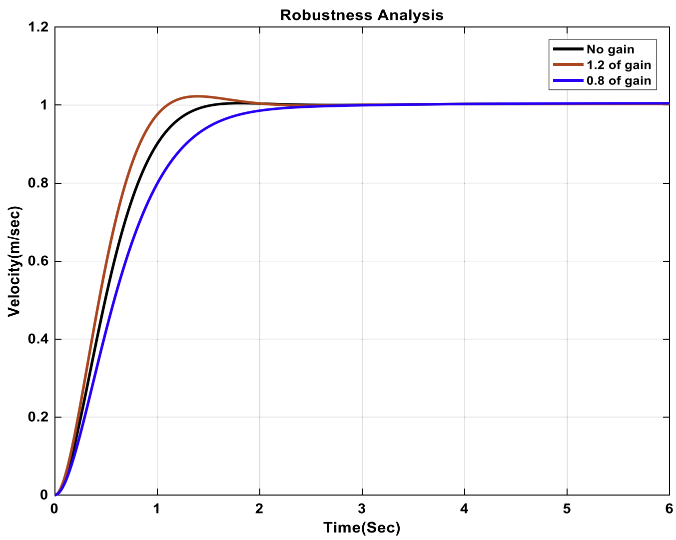


Fig. 9. Robustness analysis of ACCS.

6. Conclusion

The integration of intelligent technique in cruise control system leverages speed regulation for better speed control and traffic and safety management. The concept of integration of intelligent technique with cruise control has been recently introduced in the literature, which manages efficient control parameters tuning to achieve desired objective i.e., avoidance of collision, reducing driver fatigue, improving comfort by allowing positioning changes more safely. This paper introduces a complete design of intelligent cruise control system that integrates the monitoring sensor with the data from the intelligent feedback control system. In particular, this work uses a recent optimization method named Ant Lion Optimizer to regulate the performance indices for automobile cruise control system. This paper considers regulating the performance index **ISE** for automobile cruise control system. The ALO is used to evaluate the optimal tuning parameters for PID controlled cruise control system. From the simulation study, it is observed that the proposed controller exhibits promisingly better results than the existing methods.

References

[1] R.C. Zhao, P.K. Wong, Z.C. Xie, J. Zhao, Real-time weighted multi-objective model predictive controller for adaptive cruise control systems, Int. J. Automot. Technol. 18 (2) (2017) 279–292.
[2] M.H. Lee, H.G. Park, S.H. Lee, K.S. Yoon, K.S. Lee, An adaptive cruise control system for autonomous vehicles, Int. J. Precis. Eng. Manuf. 14 (3) (2013) 373–380.
[3] D. Corona, B. De Schutter, Adaptive cruise control for a SMART car: a comparison benchmark for MPC-PWA control methods, IEEE Trans. Control Syst. Technol. 16 (2) (2008) 365–372.
[4] N. Vedam, I. Diaz-Rodriguez, S.P. Bhattacharya, A novel approach to the design of controllers in an automotive cruise-control system, Proceeding of the 40th Annual Conference of the IEEE on Industrial Electronics Society (IECON '14), USA, 29 Oct–01 Nov, 2014, pp. 2927–2932.
[5] R. Muller, G. Nocker, Intelligent cruise control with fuzzy logic, Proceedings of the Intelligent Vehicles `92 Symposium, Detroit, MI, 1992, pp. 173–178.
[6] A. Morand, X. Moreau, P. Melchior, M. Moze, F. Guillemard, CRONE Cruise Control System, IEEE Trans. Veh. Technol. 65 (1) (2016) 15–17.
[7] H. Suzuki, T. Nakatsuji, Effect of adaptive cruise control on traffic throughput: numerical example on actual freeway corridor, Society Automot. Eng. Japan Inc. 24 (4) (2003) 403–410.
[8] M.K. Rout, D. Sain, S.K. Swain, S.K. Mishra, PID controller design for cruise control system using genetic algorithm, Proceeding of International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT'16), India, 2016, pp. 4170–4174.
[9] K. Osman, M.F. Rahmat, M.A. Ahmad, Modelling and controller design for a cruise control system, Proceeding of 5th International Colloquium on Signal Processing & Its Applications, Kuala Lumpur, 2009, pp. 254–258.
[10] R.S. Barbosa, J.A.T. Machado, I.M. Ferreira, Tuning of PID controller based on Bode's ideal transfer function, Nonlinear Dyn. 38 (2004) 305–311.
[11] Z. Dong-Li, T. Ying-Gan, G. Xin-Ping, Optimum design of fractional order PID controller for an AVR system using improved artificial bee colony algorithm, Acta Automot. Sin. 40 (2014) 973–979.
[12] K. Ranjbaran, M. Tabatabaei, Fractional order [PI], [PD] and [PI][PD] controller design using Bode's integrals, Int. J. Dyn. Control (2017) 1–13.
[13] S. Mirjalili, The Ant Lion Optimizer, Adv. Eng. Softw. 83 (2015) 80–98.
[14] E.S. Ali, S.A. Elazim, A.Y. Abdelaziz, Ant Lion Optimization Algorithm for optimal location and sizing of renewable distributed generations, Renewable Energy 101 (2017) 1311–1324.
[15] A.E. Hassanien, H. Hefny, P.W. Tsai, Antlion optimization based segmentation for MRI liver images. In genetic and evolutionary computing, Proceedings of the Tenth International Conference on Genetic and Evolutionary Computing, China, November 7-9, vol. 536, 2016, pp. 265–272.
[16] M. Raju, L.C. Saikia, N. Sinha, Automatic generation control of a multi-area system using ant lion optimizer algorithm based PID plus second order derivative controller, Int. J. Electr. Power Energy Syst. 80 (2016) 52–63.
[17] S. Miyata, T. Nakagami, Sei Kobayashi, T. Izumi, H. Naito, A. Yanou, H. Nakamura, S. Takehara, Improvement of adaptive cruise control performance, EURASIP J. Adv. Signal Process. (2010) 1–8.
[18] K. Osman, M.F. Rahmat, M.A. Ahmad, Modelling and controller design for a cruise control system, Proceedings of the 5th International Colloquium on Signal Processing & Its Applications, 2009.
[19] C. Qiu, A design of automobile cruise control system based on fuzzy PID, Proceedings of the International Conference on Information Science, Electronics and Electrical Engineering (ISEEE), 26-28 Sapporo, Japan, April. 2014, pp. 450–453.
[20] R. Rajamani, C. Zhu, Semi-Autonomous Adaptive Cruise Control Systems, IEEE Trans. Veh. Technol. 51 (5) (2002) 1186–1192.
[21] V. Milanes, S.E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, M. Nakamura, Cooperative adaptive cruise control in real traffic situations, IEEE Trans. Intell. Transp. Syst. 15 (1) (2014) 296–305.
[22] H. Fukuoka, Y. Shirai, K. Kihei, Driving support system adaptive to the driver state of surrounding vehicles: simulation study on a rear-end precrash safety system, Trans. Soc. Automot. Eng. Japan 40 (3) (2009) 933–938.
[23] Z. Situm, D. Pavkovic, B. Novakovic, Servo pneumatic position control using fuzzy PID gain scheduling, J. Dyn. Sys., Meas., Control, ASME 126 (2) (2004) 376–387.
[24] Y. Chen, Industrial information integration—a literature review 2006–2015, J. Ind. Inf. Integr. vol.2, (2016) 30–64.
[25] W. Tan, Y. Xu, W. Xu, L. Xu, X. Zhao, L. Wang, A methodology toward manufacturing grid-based virtual enterprise operation platform, Enterp. Inf. Syst. 4 (3) (2010) 283–309.
[26] C. Wang, L. Xu, W. Peng, Conceptual design of remote monitoring and fault diagnosis systems, Inf. Syst. 32 (7) (2007) 996–1004.