ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

PARTICLE SWARM OPTIMIZATION TECHNIQUE FOR RULE BASE OPTIMIZATION OF FLC FOR LOW SPEED ACC VEHICLE

S. Paul Sathiyan[1], S. Suresh Kumar[2] and A. Immanuel Selvakumar[1]
[1]Department of Electrical and Electronics Engineering, Karunya University, Coimbatore, India
[2]Department of Electronics and Communication Engineering, Dr. NGP Institute of Technology, India
E-Mail: sathiyanme@yahoo.co.in

ABSTRACT

Adaptive Cruise Control is used in vehicles for giving support to the drivers and to circumvent rear end collision. Due to the presence of nonlinearity in the vehicle (system), it is difficult to design an exact mathematical model of the system. Heuristic based fuzzy logic controller with optimized rule base, which does not require accurate mathematical modelling is proposed here which uses the knowledge of the designer for developing the rule base. The performance metrics in terms of better control and response time of the controller depends upon how well the rule base is formulated. More the number of rules in the rule base, higher the search time and increase in the total response time of the system but better will be the control. On the other hand, if the rules are less, then the search time will reduce which in turn decreases the total time of response of the system but control will be poor. In order to obtain an optimized control, the rule base is optimized using Particle Swarm Optimization technique. The result had shown a better performance.

Keywords: adaptive cruise control (ACC), fuzzy rule base optimization, particle swarm optimization.

1. INTRODUCTION

According to Association for Safe International Road Travel, road traffic crashes rank as the 9th leading cause of death and account for 2.2% of all deaths globally. Rear end collision contributes to a considerable percentage of this death. Factors like driver's inattention or distraction, tailgating, panic stops, and reduced traction due to weather or worn pavement causes rear end collision. In order to assist the driver and to circumvent rear end collision, Adaptive Cruise Control (ACC) was developed. ACC works in Distance Control Mode (DCM) to avoid rear end collision whenever there is a vehicle travelling in front (lead vehicle) of the vehicle fitted with ACC (host vehicle) and in Velocity Control Mode (VCM) to travel at a constant set speed by the driver and relives the driver from constantly pressing the accelerator pedal whenever there is no lead vehicle in the front of the host vehicle [1-3]. During DCM, the comfort and safety of the occupants of the host vehicle should be considered for controller design. Comfort levels are fixed based on the level of jerks that the occupants' of the host vehicle may experience. Safety level is fixed based on the speed at which the host vehicle travels while following a lead vehicle. Percentage influence of these factors upon the controller depends upon the speed at which the vehicle travels and the severity of the error between the expected value (speed, distance) and the actual value (speed, distance). The performance of the controller for ACC vehicle proposed in [4] shows that the host vehicle speed oscillates as the speed during DCM is reduced below 30km / hr. At congested traffic, such ACC system becomes less useful. At lower speed and longer time of driving, the driver's legs were strained. Also most of the ACC system which works above 30km/hr is not useful in traffic jams or urban driving, situation. This research article focuses on implementing a controller which could control the vehicle below and around 30km/hr (8.3m/sec). For a highly non linear application like ACC, during uncertainties, the conventional mathematical modeling based controllers like PID, SMC etc., may not perform the way it should while it handles multiple parameters [2]. Due to the heuristic nature associated with simplicity and effectiveness during both linear and non linear situation, Fuzzy Logic Controller (FLC) is used in this paper. The performance of the FLC depends upon how well the rule base is formed [5]. The hitch with the FLC-based system is that, the performance of the controller highly relies upon the number of membership values and the rules. The rule increases exponentially with the increase of the number of membership values that involve in the rules. This increase, leads to the rise in the computation time to determine the crisp output by the controller. The crisp output of the FLC does not depend upon the best rule rather it depends upon the entire rule which gets qualified [6]. Whenever Particle Swarm Optimisation (PSO) algorithm is used in real time optimization of fuzzy rule base, the time for reaching the optimized value, depends upon the population size i.e., number of rules in the fuzzy rule base [6-7]. Therefore is not advisable for the time critical ACC application. Offline tuning of fuzzy rule base will help in gaining the advantage of PSO's optimization and effective nonlinearity handling ability of FLC for ACC. The rest of the paper is organised as follows: Vehicle Modelling and FLC based ACC, PSO based fuzzy rule base optimisation, Simulation, Conclusion and References. In this work the performance of the proposed controller model is compared with the conventional FLC based ACC.

2. VEHICLE MODELLING AND FLC BASED ACC

The Vehicle model was designed considering the basic parameters like aerodynamic drag, road inclination and friction between the road and the tyre. The modelling is done for carrying out the simulation using Matlab. Figure-1 shows the basic drive train system over which the modelling is carried out.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

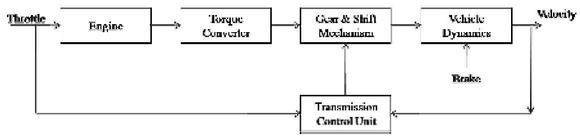

Figure-1. Basic drive train system.

The moment of inertia of the vehicle is given by:

$$m\frac{dv}{dt} = F - mgsin(\theta) + mgC_r sin(v) + \frac{1}{2}\rho C_d A v^2 \quad (1)$$

The above equations are modelled in Matlab Simulink for the purpose of simulation with the following values

T_m = 190Nm, ω_m = 4000rpm β = 0.4, C_r = 0.01, ρ = air density (1.3k/m^3), C_d = 0.32, A = 2.4m^2

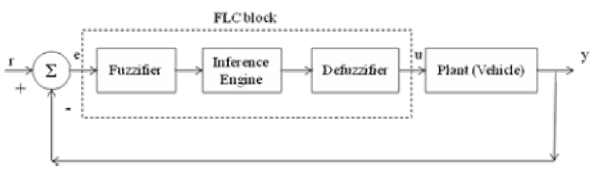

Figure-2. Block diagram of FLC.

The basic block diagram of FLC is shown in Figure-3. It consists of fuzzification, inference engine and defuzzification. The output is given by the formula

$$Y_{1}out = ((_{1}i\ w_{1}i\ a_{1}i)/((_{1}i\ w_{1}i\ ) \quad (2)$$

Where w_i represents the value of the weight of each rule i and a_i is the crisp value of each rule i condition, understanding weight as the degree in which the crisp current values of the inputs satisfy the set of rule condition. The error inputs were fuzzified. Each input error variable has 7 membership values spread over the error range from -30 to 30 for velocity error and -15 to 15 for distance error. The output variable ranges from -1 to 1 with 7 membership values. Triangular membership function and centre of gravity method is preferred for defuzzification process. Effectiveness of FLC depends upon the following parameters

a. How good the rule base is defined
b. Number of linguistic variables used
c. Type of the membership functions
d. Range of the fuzzy membership functions.

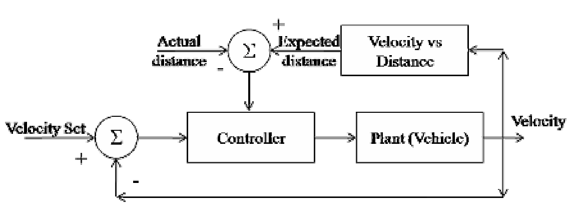

Figure-3. Block diagram of ACC.

The above figure shows the block diagram of an ACC system. The input to the controller during VCM is the difference between the set velocity and the actual velocity. The input to the controller during DCM, are the velocity error and the distance error.

$$X_{error} = X_{actual} - X_{desired} \quad (3)$$

$$X_{desired} = X_{safe} + THW\ x\ V_{host} \quad (4)$$

$$V_{error} = V_{lead} - V_{host} \quad (5)$$

$$V_{lead} = V_{host} + X_{error}\ x\ 3.6 \quad (6)$$

Where
X_{error} = Distance error
V_{error} = Velocity Error
V_{host} = Velocity of the host vehicle in (m/sec)
V_{lead} = Velocity of the lead vehicle in (m/sec)
X_{safe} = minimum distance of safety in meters
X_{actual} = Actual distance between the vehicles in meters
THW = Time headway

The Velocity error is the difference between the expected velocity at which the vehicle is suppose to travel for the distance maintained between the lead vehicle. The distance error is the difference between the expected distance which the vehicle has to maintain for the velocity at which the vehicle is travelling and the actual distance the vehicle is maintaining between the lead vehicle. For validating the proposed control technique, a vehicle model proposed in [8] is adopted. The cruise controller regulates the velocity v, of the vehicle as per the drivers' requirement by adjusting the throttle valve angle u and the brake torque.

3. PSO BASED FUZZY RULE BASE OPTIMISATION

Excessive number of fuzzy rules and the number of fuzzy linguistic variables will lead to increase in the search time and hence the computational time [9]. Using online tuning of fuzzy rule base is not advisable for time critical application like ACC, due to time taken by the algorithm to reach the global optimisation point for a given error input, hence offline tuning is preferred.

PSO algorithm finds the optimal solution using population of particles where the information is shared between the members. PSO uses the fitness concept, but less-fit particles do not die, i.e., No survival of the fittest. PSO uses the number of agents that constitute a swarm moving around in the search space looking for the best solution. Each particle in the search space adjusts its pattern of flying (velocity, position) dynamically by considering the experience of its own and that of the other particles in the group [10].

$$v_i^{k+1} = w^* v_i^k + c_1^* rand_1^* (pbest_i - x_i) + c_2^* rand_2^* (gbest_i - x_i) \quad (7)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (8)$$

VOL. 9, NO. 6, JUNE 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

where
w is inertia weight
c_1,c_2 is the accelerating constants
v_i^k and x_i^k are the velocity and the position of particle i in k^{th} iteration, *xi*
rand1, rand2 are random numbers between 0 and 1.

Equation (7) has three component which helps in obtaining the new velocity for the particle.

$w^*v_i^k$ - Inertial component
$c_1{}^*rand_1{}^*(pbest_i - x_i)$ - Personal influencing Component
$c_2{}^*rand_2{}^*(gbest_i - x_i)$ - Social influencing Component

Inertia component makes the particle move in the same direction with the same velocity. Personal influencing component improves the individual, makes the particle return to its previous position, better than the current and its conservative. Social influencing component makes the particle follow the best neighbours' direction.

3.1. PSO Algorithm for tuning fuzzy rule base

Step-1: Initialize related parameters, including the size of swarm m, the inertia weight w, the acceleration constants c_1 and c_2, the maximum velocity V_{max}, the stop criterion and the *m* particles of a population, which include the random location X_1 in the problem space and the velocity for the d^{th} dimension.

Step-2: Evaluate the desired optimization fitness function for each particle.

Step-3: Compare the evaluated fitness value of each particle with its P_{best}. If current value is better than *best P* then set the current location as the *best P* location. Furthermore, if current value is better than *best g*, then reset G_{best} to the current index in particle array.

Step-4: Change the velocity and location of the particle according to the Eqs. (7) and (8), respectively.

Step-5: Loop to step-2 until a stopping criterion is met. The criterion usually is a sufficiently good fitness value or a predefined maximum number of generations Gmax.

$$X_{error}(k) \rightarrow 0, \;\; V_{error}(k) \rightarrow 0 \tag{9}$$

The prime goal of the ACC is to follow the preceding vehicle without collision (minimum distance error and relative velocity) (9). Hence the individual having pbest and gbest are computed [11].

The 42 rules mentioned in Table-1 were considered from the rule based proposed in [2]. These rules are the members of the PSO and each member gives different solution in the solution space. The iteration count is 200.

Table-1. Fuzzy rule base table.

		Distance error						
		NL	**NM**	**NS**	**Z**	**PS**	**PM**	**PL**
Relative speed	**NM**	NVL	NL	NM	NM	NS	NS	Z
	NS	NL	NM	NS	NS	NS	Z	Z
	Z	NM	NS	Z	NS	Z	PS	PS
	PS	NM	Z	Z	Z	Z	PM	PVL
	PM	NS	Z	Z	PS	PM	PL	PVL
	PL	NS	Z	Z	PS	PL	PVL	PVL

4. SIMULATION

The simulation is performed with the vehicle model described in section 2. The performance of conventional FLC and FLC with optimized rule base for vehicle following scenarios are considered with different inter-vehicle distance, different rate of acceleration and deceleration. The optimisation of rule base is performed considering the following driving patterns shown below (Figure-4(a-d): Inter-vehicle distances) with different inter-vehicle distance and different rate of acceleration and deceleration. The inter-vehicle distance is varied from 3 meters to 9 meters. The minimum safer distance is kept as 3 meters.

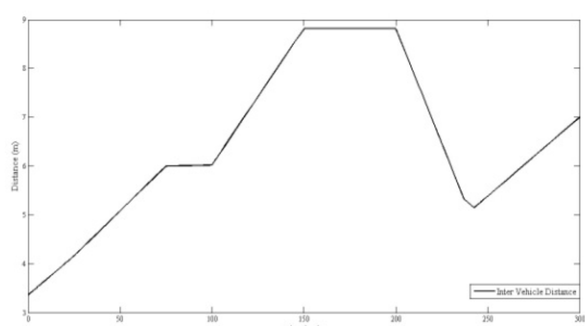

Figure-4(a). Inter-vehicle distance.

VOL. 9, NO. 6, JUNE 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

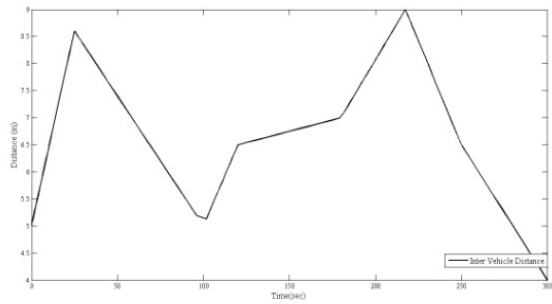

Figure-4(b). Inter-vehicle distance.

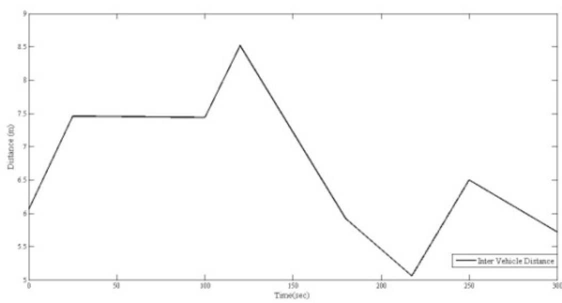

Figure-4(c). Inter-vehicle distance.

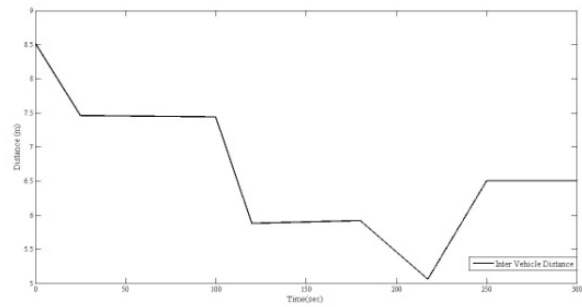

Figure-4(d). Inter-vehicle distance.

As there were oscillations observed in the velocity of the host vehicle recorded in [4] during DCM at speed below 30km/hr, the simulation is performed for speed below 30km/hr (8.3m/s). Relation between velocity of the vehicle and the distance to be maintained in shown in Table-2.

Table-2. Typical stopping distance.

Speed M/H	**Velocity m/s**	**Thinking distance** (Feet) / (Meter)	**Braking distance** (Feet) / (Meter)	**Overall stopping distance** (Feet)
20	8.94	20 / 6	20 / 6	40
30	13.4	30 / 9	45 / 14	75
40	17.9	40 / 12	80 / 24	120
50	22.4	50 / 15	125 / 38	175
60	26.8	60 / 18	180 / 55	240
70	31.3	70 / 21	245 / 75	315

Thinking distance is taken as the speed in Miles per hour at which the vehicle is travelling. Stopping distances for wet road is twice the overall stopping distance and for snowy and icy roads the stopping distance is 10 times the overall stopping distance mentioned in Table-2. The conversion of distance in feet to meter is rounded to the nearby value.

The model was first simulated considering the four vehicle following scenarios separately. The optimised rule for all the four scenarios mentioned was found. After that the overall optimized rule base was found so that the host vehicle can follow the lead vehicle at different starting velocity. Figure with sub index 'a' of 5 to 8 shows the velocity relation between the host and the lead vehicle with FLC and Figure with sub index 'b' of 5 to 8 shows the velocity relation between the host and the lead vehicle with FLC with PSO optimised rule base. The later controller could precisely follow the lead vehicle than the former controller.

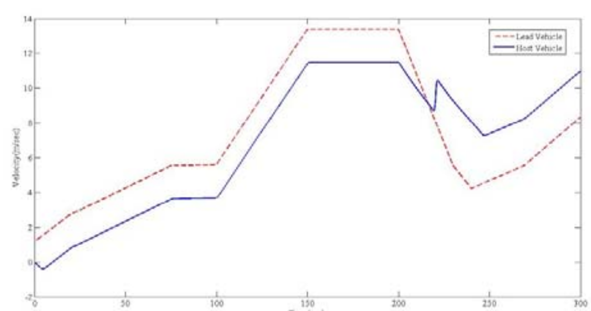

Figure-5(a). Velocity time curve (FLC).

VOL. 9, NO. 6, JUNE 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

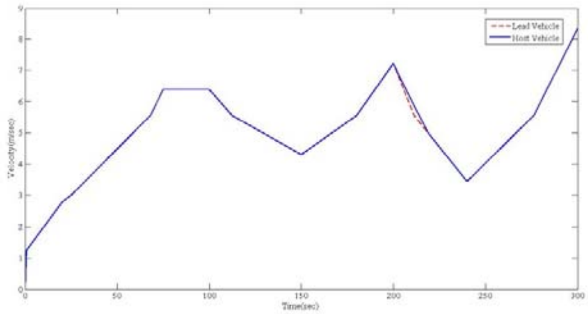

Figure-5(b). Velocity time curve
(FLC with PSO optimised rule base).

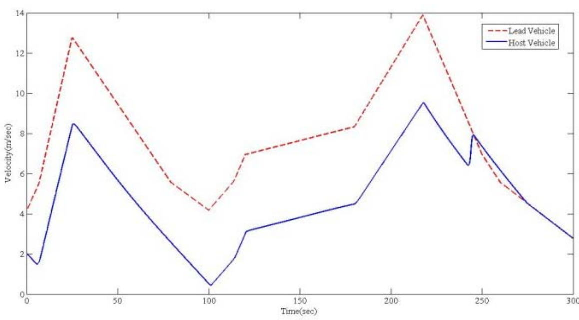

Figure-6(a). Velocity rime curve (FLC).

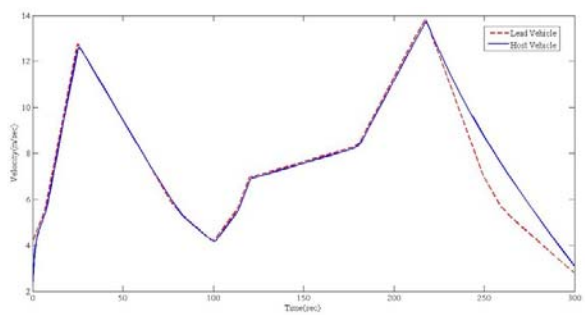

Figure-6(b). Velocity rime curve.
(FLC with PSO optimised rule base)

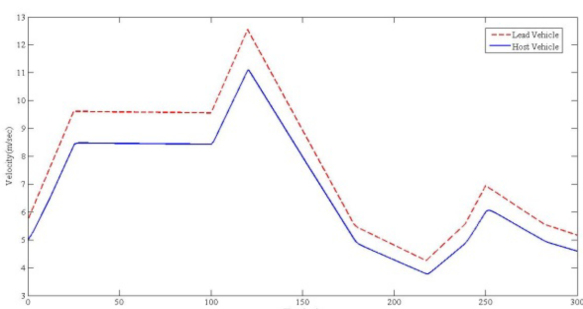

Figure-7(a). Velocity rime curve (FLC).

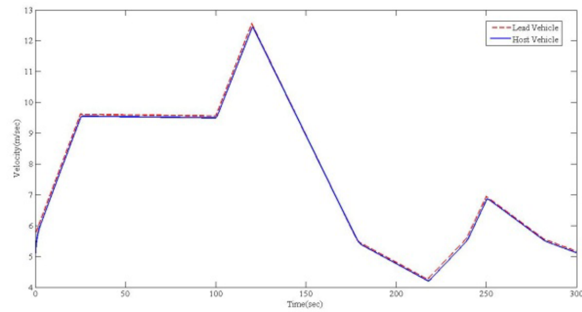

Figure-7(b). Velocity rime curve.
(FLC with PSO optimised rule base)

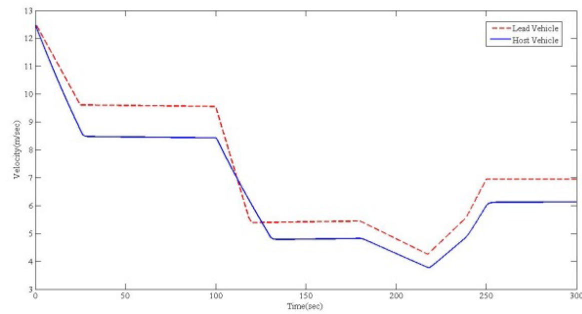

Figure-8(a). Velocity rime curve (FLC).

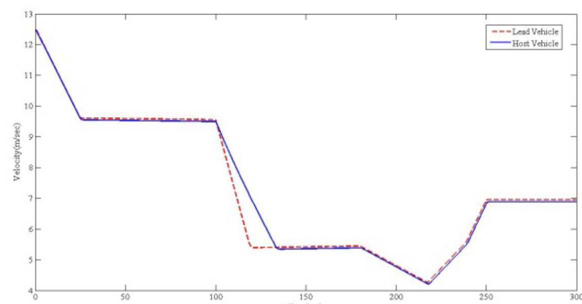

Figure-8(b). Velocity rime curve.
(FLC with PSO optimised rule base)

The following figures (Figures 9-12) show the jerk experienced by the host vehicle during DCM for the above four scenarios.

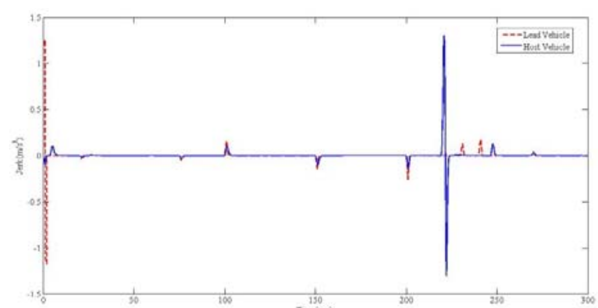

Figure-9(a). Jerk rime curve (FLC).

VOL. 9, NO. 6, JUNE 2014 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

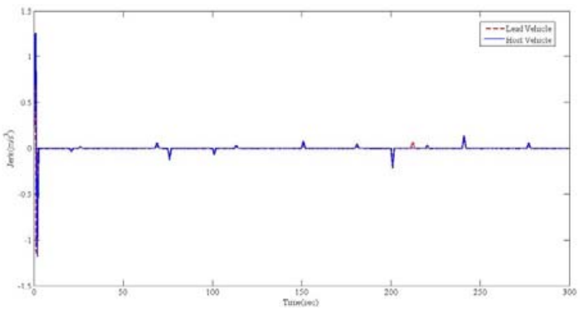

Figure-9(b). Jerk rime curve.
(FLC with PSO optimised rule base)

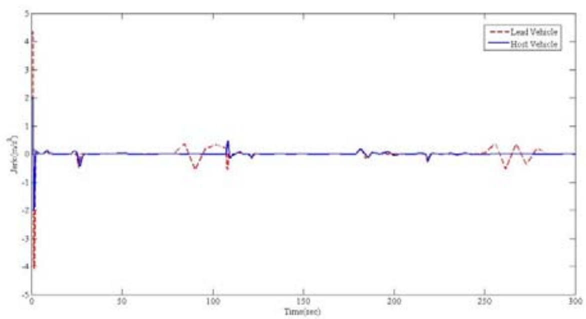

Figure-10(a). Jerk Time Curve (FLC).

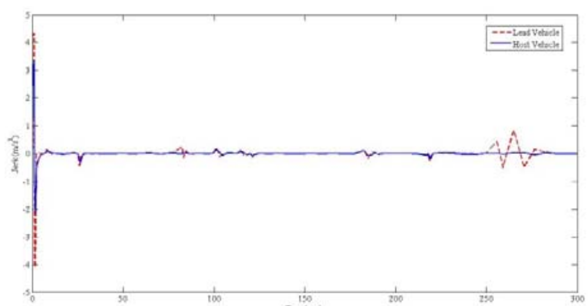

Figure-10(b). Jerk rime curve.
(FLC with PSO optimised rule base)

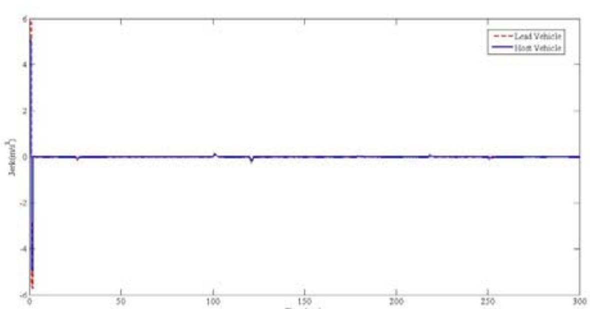

Figure-11(a). Jerk rime curve (FLC).

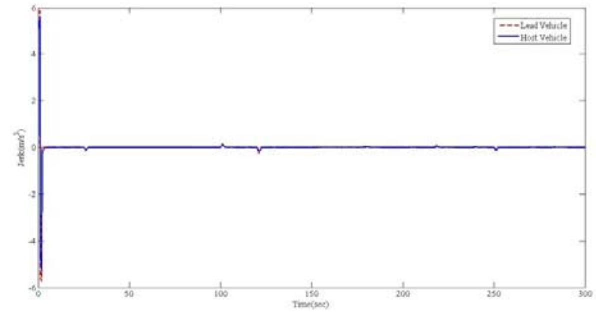

Figure-11(b). Jerk time curve.
(FLC with PSO optimised rule base)

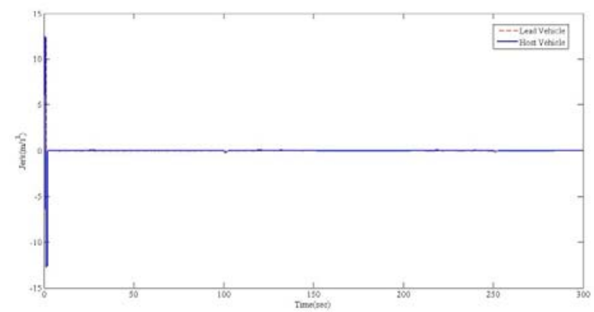

Figure-12(a). Jerk Time Curve (FLC).

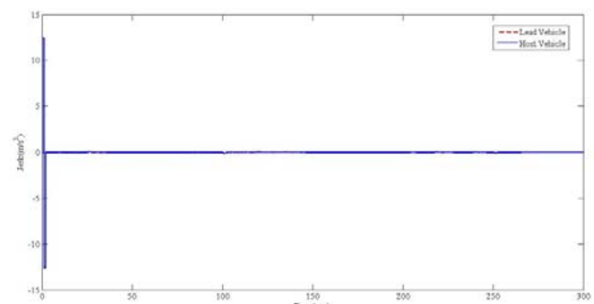

Figure-12(b). Jerk time curve (FLC with PSO optimised rule base).

5. CONCLUSIONS

The performance of the vehicle with the proposed methodology is validated with the conventional FLC based ACC. The proposed methodology shows an improved performance during vehicle following. The simulation is carried out with different starting velocity, rate of acceleration and deceleration. The rule base was optimised using PSO algorithm. The vehicle was able to follow the lead vehicle without much velocity oscillations. Also the jerk experienced by the host vehicle is reduced which ensures the comfort of the occupants in the host vehicle. Further the research is focused towards extending the working range of the system.

REFERENCES

[1] Rudwan A., Amir H., Kevin W. and Ali Z. 2008. Autonomous intelligent cruise control using a novel multiple-controller framework incorporating fuzzy-logic-based switching and tuning. In: Neurocomputing, Elsevier. 71(13-15): 2727-2741.

[2] Pananurak W., Thanok S. and Parnichkun M. 2009. Adaptive Cruise Control for an Intelligent Vehicle. In: Proceedings of the IEEE International Conference on Robotics and Biomimetics ROBIO '08, Feb 22-25, Bangkok. pp. 1794-1799.

[3] Paul S., Suresh S. and Immanuel A. 2011. Optimisation of ACC using Soft Computing Techniques. In: International Journal of Computer Science and Information Security. Pennsylvania, U.S.A. 9(2): 150-154.

[4] Naranjo J.E., Gonzalez C., Reviejo J., Garcia R. and de Pedro T. 2003. Adaptive Fuzzy Control for Inter-Vehicle Gap Keeping. In: IEEE Transactions on Intelligent Transportation Systems. 4(3): 132-142.

[5] Kwang S.C. and Jae S.C. 1995. Automatic Vehicle following using the Fuzzy Logic. In: Proceedings of the 6th International Conference on Vehicle Navigation and Information Systems VNIS '95, July 30-Aug 2, Seattle, WA. pp. 206-213.

[6] Paul S., Suresh S. and Immanuel A. 2013. Segmented Fuzzy Logic Controller for Vehicle Following With Optimised Rule Base. In: Journal of Theoretical and Applied Information Technology. 57(1): 7-15, Pakistan.

[7] Jassbi J., Khanmohammadi S. and Kharrati H. 2008. A New Hybrid Method for Determination of Fuzzy rules and Membership Functions. In: IEEE Congress on Evolutionary Computation CEC'08, June 1-6, Hong Kong. pp. 1649-1654.

[8] Osman K., Rahmat M.F. and Ahmad M.A. 2009. Modelling and controller design for a cruise control system. In: Proceedings of the 5th International Colloquium on Signal Processing and Its Applications CSPA '09, March 6-8, Kuala Lumpur. pp. 254-258.

[9] Streifel R.J., Marks R.J., Reed R., Choi J.J. and Healy M. 1999. Dynamic Fuzzy Control of Genetic Algorithm Parameter Coding. In: IEEE transactions on systems, man, and cybernetics-Part B: Cybernetics. 29(3): 426-433.

[10] Kennedy J. and Eberhart R. 1995. Particle Swarm Optimisation. In: Proceedings of IEEE International Conference on Neural Networks, Nov/Dec, Perth, WA. pp. 1942-1948.

[11] Bahareh A. and Hamed S. 2010. Bio-Inspired Algorithms for Fuzzy Rule-Based Systems. TMRF e-book Advanced Knowledge Based Systems: Model, Applications and Research, Iran.