ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014

Speed Control of Hybrid Electric Vehicle Using Optimization Algorithm

Avtar Singh[1], Er. Amrit Kaur[2]

Student of Department of Electronics and Communication Engineering, Punjabi University, Patiala, India[1]

Faculty of Department of Electronics and Communication Engineering, Punjabi University, Patiala, India[2]

Abstract: Hybrid electric vehicle (HEV) is getting more attention recently because it has no emission of toxic gases which leads to global warming. Now a days, Particle swarm Optimization Gravity Search Algorithm (PSOGSA) is widely used for optimization, because of its faster convergence rate to the optimal minimum value than other optimization techniques. The main aim of this paper is to control the speed of non-linear hybrid electric vehicle by controlling the throttle position so as to get driving safety, improved fuel economy, reduced manufacturing cost and pollution. To control the speed of hybrid electric vehicle tuning of PID controller is done using particle swarm optimization gravity search algorithm. The performance of the technique is determined by taking mean square error(MSE) as a fitness function. The comparative study is carried out to identify the suitable optimization technique for controlling the various parameters of PID for HEV. The results obtained by tuning of PID with PSOGSA technique compared with other techniques such as Zeigler-Nichols tuning method and standard PID tuning.

Keywords: Hybrid Electric Vehicle, Gravity search algorithm (GSA), PID controller, Particle Swarm optimization (PSO), Particle Swarm optimization Gravity search algorithm (PSOGSA), Zeigler-Nichols (ZN).

I. INTRODUCTION

A major source of green house effect is increasing number of vehicles. Scientists are finding ways to reduce it. So, in recent times hybrid electric vehicles are getting more attention because it has no emission which will lead to global warming. Moreover the sources used for generation of electric power battery of electric vehicles are non conventional source which are environment friendly. The HEVs uses both electric machine and IC engine for delivering the propulsion power [1],[2]

The oppugns on which research work is still going on to achieve high performance efficiency, ruggedness, compact sizes, and less costs in power converters and electric machines, as well as in associated electronics [3].
This capability is mainly due to:

1. The possibility of downscale of engine.
2. The enery recovery ability of rechargeable.
3. Storage system during breaking phases.
4. The power can be split into thermal and electrical paths, so it enables to have an additional degree of freedom.

Also, as summarized below HEVs have more benefits than conventional ICEVs [4],

1. The torque generation in electric vehicles is very much prompt and veracious, so
2. crash can be easily averted by decreasing the torque.
3. The output torque of these motors can be easily estimated.
4. A motor can be inclined to each wheel of electric vehicle.
5. Give high efficacy over broad torque and speed pasture.
6. High sureness and robustness for vehicular environment .

The pioneering authors working on the intelligent energy management in HEVs have proposed compendious category and anatomize of the state of art control design for the same. The performance of HEVs merely depends upon the applied automation system. Many controller such as LQR controller, state feedback controller (SFC), Observer based controller have been designed [5],[6]

Several algorithms are implemented for the purpose of optimization on the basis of premier knowledge of the future driving cycle. Initially fuzzy logic based controllers were designed and tested in different kinds of applications, but fuzzy controller have more adjustable parameters in the rule analysis, because of that manual tuning of these parameters could be feckless. Kim et al.[6] suggested a self-organizing fuzzy logic controller for wheeled mobile rotor using evolutionary algorithm. But both controllers were together used for certain problems as fuzzy controller alone was not able to give certain features of adaptive controller [6], [8],[9]

The speed of hybrid electric vehicle can be controlled by controlling the servo motor which in turn is controlling the throttle position to have smooth torque. Some other authors have purposed different techniques for electric vehicles [10].

In this paper, we propose (1) control the speed of hybrid electric vehicles with conventional PID controller, tuning of PID controller is done with classical techniques and metaheuristic algorithm, (2) Zeigler-Nichols Tuning Method, (3) MATLAB toolbox PID tuner (3)Particle Swarm Optimization Gravity search Algorithm (PSOGSA).

PSOGSA is used to beget a number of possible solutions for k_p, k_i, k_d. The performance index evaluates its value

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

and selects it based on the fitness. Such features of PSOGSA make it easy and fast convergent.

This paper is organized as follows: in section 2 the detail of the HEV system is described. Section 3 presents a summary of the stability analysis of the control system. In section 4 the control strategy used is summarized. Section 5 provides the inclusive overview of PSOGSA algorithm, experimental results are drawn in section 6, and hence conclusion is drawn in section 7.

II. MODEL DESCRIPTION OF VEHICLE

A schematic diagram of electronic throttle control using a DC servo is shown in Fig. 1. Simulink model of system is shown in fig. 2.

The dynamics of the vehicle is given as [5],[8]

$$m\frac{dv}{dx} = F_e(\theta) - \alpha v^2 - F_g \tag{1}$$

$$\tau_f \frac{dF_e(\theta)}{dx} = -F_e(\theta) + F_{e1}(\theta) \tag{2}$$

$$F_{e1}(\theta) = F_1 + \gamma\sqrt{\theta} \tag{3}$$

F_e = Engine force, a function of throttle position

F_g = Gravity induced force, a function of the road grade.

θ =Throttle Position, v=vehicle speed

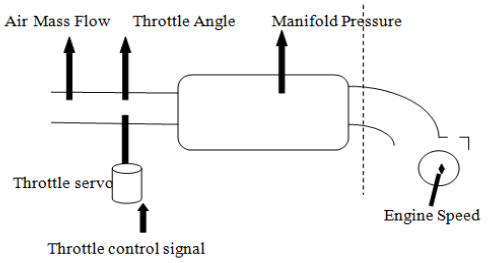

Fig. 1 Schematic diagram of electronic throttle control

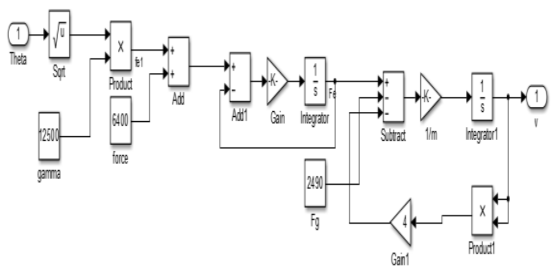

Fig.2 Simulink model of vehicle

A. Assumptions

1. Gravity induced force (F_g) is 30% of weight of vehicle.
2. Engine time constant (τ_f) is taken as 0.2 that is in the range from 0.1 to1.0 sec.

   The values of remaining parameters are given in Table 1 [5]

Table 1
Numerical Values of Parameter

Constants	Notation	Value(SI unit)
Mass (m)	m	1000 Kg
Aerodynamic Drag Coefficient	α	4 N/(m/s)2
Engine Force Coefficient	γ	12500 N
Engine Idle Force	F_1	6400 N

By using given system dynamics and hence state space equation transfer function is obtained [5]

Transfer function

$$\frac{V(s)}{\theta(s)} = \frac{829000}{s^2+5s}$$

III. STABILITY ANALYSIS

The state space variable is given as [4]

$$A=\begin{bmatrix}0 & 0.001\\ 0 & 5\end{bmatrix},\ B=\begin{bmatrix}0\\ 829000\end{bmatrix},\ C=[1\ \ 0],\ D=[0]$$

The chararacterstic equation is given as

|λI-A|=0

When we get the Eigen values as λ_1 =0, λ_2 =-5. Then given system is controllable and observable. As the order of matrix M and N is equal to rank of the matrix =1;

$$M=829000\begin{bmatrix}0 & 0.001\\ 0 & -5\end{bmatrix},\ N=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$

IV. CONTROL STARTEGY

4.1 PID Controller

Proportional – integral- derivative control is a feedback control system having three parameters k_p, k_i and k_d. All these parameter have their own advantage. So that PID controller add all the advantages of the parameter such faster response time due to P-only control, decreased/servo offset due to I- only control and prediction of disturbance to the system by determining the change in error due to D-only control. The transfer function of PID is given as [17].

$$C(S)= K_p + \frac{K_i}{s} + k_d s = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \tag{4}$$

Where k_p =proportional Gain, k_i =Integral Gain, k_d =Derivative Gain, Ti=Reset Time= k_p/k_d, Td=Rate Time or derivative time= k_d/k_p.

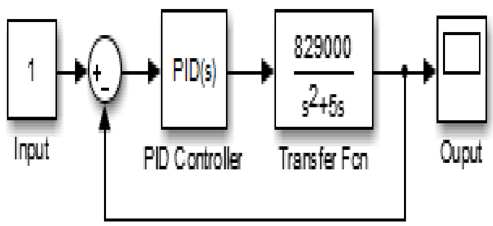

Fig. 3. Simulink Model Of plant with PID controller

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014

4.2 Performance Estimation of PID controller

In general, during the design of PID controller various methods are used in control system that are integrated absolute error (IAE), or integral of squared error (ISE), or integral of time weighted squared error (ITSE). Here we use mean square error (MSE) as a performance estimator. The formula of MSE is given as :

$$\text{MSE}=\frac{1}{n}\sum_{i=1}^{n}(r(t)-y(t))^2 \quad (5)$$

V. AN OVERVIEW OF PSOGSA

In this section first the brief detail of standard PSO algorithm is given, then brief detail of standard GSA algorithm is given and then an overview of PSOGSA algorithm is presented.

5.1 *Standard Particle Swarm Optimization (PSO)*

In 1995, Kennedy and Eberhart first introduced the particle swarm optimization (PSO) algorithm.PSO is one of the optimization techniques and is found to be robust in solving non-linear, non-differentiable problems [12].

The PSO was motivated from social behavior of bird flocking. It uses a number of particles which fly around in the search space to find best solution. In the meantime, they all look at the best particle (best solution) in their paths [13]. In additional words, particles consider their own best solutions as well as the best solution that has been found so far. Every particle in PSO should consider the current position, the current velocity, the distance to pbest, and the distance to gbest to modify its position. Mathematical equation of PSO algorithm is given as [14], [15].

$$v_i^{(t+1)} = w * v_i^t + c_1 * rand * (\text{pbest}_i - x_i^t) + c_2 * rand * (\text{gbest} - x_i^t) \quad (6)$$

$$x_i^{(t+1)} = x_i^t + v_i^{(t+1)} \quad (7)$$

where *n*=number of particles, t=pointer of iteration, w=inertia weight factor, *w*=wmx-((wmx-wmi)/maxIterations)*iterationCount, *c1* and *c2*=acceleration constant, rand=random number between 0 to 1; x_i^t =current position of i^{th} particle. v_i^t =velocity of particle *i* at iteration t. pbest_i = pbest_i of agent i at iteration t.

The PSO starts randomly by placing the particles in problem space. In every iteration, velocity of the particles calculated using (6) and then current position updated using equation (7).The process of updating position of the particles continue until the desired results are not obtained.

5.2 Standard Gravity Search Algorithm (GSA)

GSA is an optimization method and GSA is inspired from Newton's theory which states that every particle in this universe attract each other with a force directly proportional to the product of their masses and inversely proportional to the square of the distance between them.GSA can be viewed as a group of particles having masses proportional to their value of fitness function. Around the generation, due to gravity of force all masses attract each other between them. Massive particle have greater force of attraction. Consequently, the massive particle which is possible close to the global optimum attracts the other particles inversely proportional to square of their distance. Mathematical representation of GSA is as follows. Suppose a system have N particles. The algorithm starts randomly by placing particles in a search space. During all epochs, gravitational force from particle *j* to particle *i* at a certain time is defined as follows [16].

$$F_{ij}^t = G(t) \times \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t)+\varepsilon} \times (x_j^d(t) - x_i^d(t)) \quad (8)$$

Where $M_{aj}(t)$= attractive gravitational mass related to particle *j*, $M_{pi}(t)$= passive gravitational mass related to particle *i*, $G(t)$ = Gravitational constant at time t, $R_{ij}(t)$ = Euclidian distance between two particles *i* and *j*, ε = small constant.

G(t) is calculated as

$$G(t) = G_0 * \exp(-\alpha * \text{cit}/\text{maxIt}) \quad (9)$$

Where α is descending coefficient, G_0 is a initial value. cit= current iteration, maxit=maximum number of iteration. In a problem of search space with dimension *d*, the total force that acts on particle *i* is calculated as follows [17].

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t) \quad (10)$$

Where $rand_j$ is random number between 0 to 1. By law of motion, acceleration of particle is proportional to force and is inversely proportional to their mass, so acceleration of particles can be calculated as follows:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (11)$$

Where t is a certain time and M_{ii} is a mass of particle *i*, the velocity and position of particle is calculated as follows:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (12)$$

$$x_i^d(t) = x_i^d(t) + v_i^d(t+1) \quad (13)$$

Where $rand_i$ is a random number between 0 to 1. In GSA , first of all each mass of particle is initialized with random values. Each mass is a particle solution. By following the initializations, velocity of particles is calculated using (12). By using gravitational constant, total force and acceleration is calculated by using (9),(10) and (11). The position of particle masses is calculated using (13). At the end GSA will stop when meeting of an end criteria

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 6, June 2014
ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

5.3 . Hybrid PSOGSA Algorithm

In hybrid PSOGSA, we hybridize the PSO with GSA algorithm by using low-level co-evolutionary heterogeneous hybrid. The hybridization is a low level because we combine the functionality of two algorithms. The basic idea of PSOGSA is to combine the capability of social thinking (gBest) in PSO with local search capability of GSA. In order to combine these two algorithms ,(14) is proposed as follows [16].

$$v_i(t) = w \times v_i(t) + c_1 \times rand \times a_i(t) + c_2 \times rand \times (gBest - x_i(t)) \quad (14)$$

In every iteration, position of particle is updated as follows:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (15)$$

In PSOGSA, first all the particles are randomly initialized. Every particle is counted as a candidate solution. After initialization, gravitational force, gravitational constant and resultant force among the particles are calculated using equation (8), (9) and (10) respectively. After that acceleration of the particle is calculated using equation (11). After each iteration, the best solution until now should be updated. After determining acceleration and with update best solution, the velocities of all particles can be determined using (14). And then position of particles is updated using (15). The whole procedure will continue until the desired results are not obtained. The flow chart of PSOGSA algorithm is represented in figure 4 [16].

Table 2

Parameters of PSOGSA

Parameters	Values
Population Size	20
Space Dimension	3
Maximum Iterations	20
G_0	1
c_1	2
c_2	2
Alpha(α)	20
wmx	0.9
wmi	0.4
Objective Function	MSE

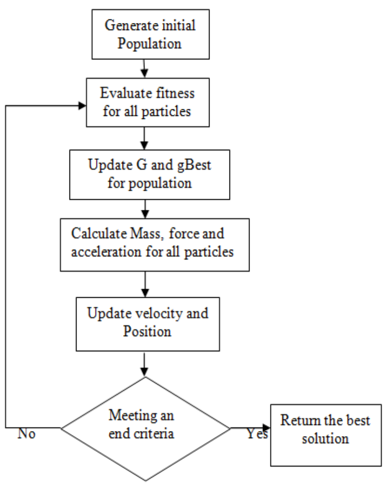

Fig. 4 Flow Chart of PSOGSA

VI. SIMULATION RESULTS AND DISCUSSION

In this section, results are shown of open loop system without any controller and closed loop system with PID controller. The responses obtained from ZN-PID, PID tuner MATLAB toolbox and PSOGSA-PID tuned PID are shown and compared. Fig. 5 shows the open loop step response which is unstable and not converging.

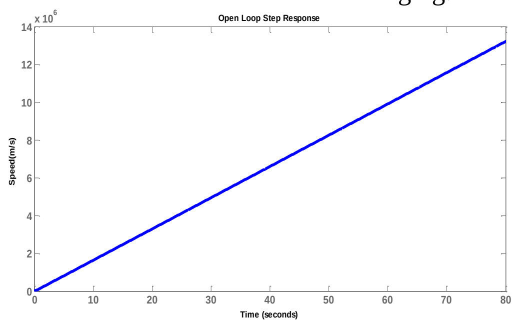

Fig. 5 Open Loop Step Response

Fig. 6(a) shows the comparison of closed loop response of PID controller tuning with ZN-Method and MATLAB toolbox PID tuner. Fig. 6(b) shows the Closed loop response tuning with PSOGSA-PID

PID controller is tuned by Zeigler-Nichols method and MATLAB R2013a toolbox PID tuner that provide maximum overshoot of 37.3% and 9.8% respectively.

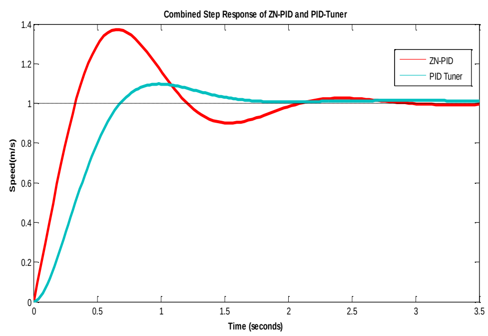

Fig. 6(a) Closed loop response tuning with ZN-Method and MATLAB toolbox PID Tuner

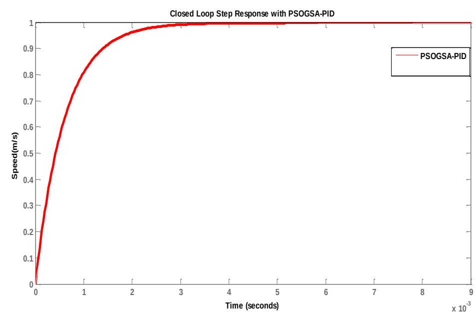

Fig. 6(b) Closed loop response tuning with PSOGSA-PID

Table 3.

Performance Index of Various Controllers for HEV

Controller	%Max overshoot	Settling time (sec.)	Rise Time (sec.)
ZN-PID	37.3	2.62	0.256
PID	9.8	1.61	0.461
PSOGSA-PID	**0**	**0.0024**	**0.0013**

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

The optimized values of PID controller that are obtained with PSOGSA-PID are k_p =0.002, k_i =0.002 and k_d =0.002 . So that the improved results are obtained and tuning techniques for PID are compared in Table 3.

VII. CONCLUSION

A different kind of PID controller tuning techniques is applied to control the speed of non-linear hybrid electric vehicle. From the results, shows that PSOGSA tuned PID controller gives excellent performance in all respects. Maximum overshoot and settling time is very much less to achieve the desired speed so that current and torque will also be optimized and also its battery operation. From above we may conclude that hybrid Particle swarm optimization Gravity search algorithm (PSOGSA) based controller gives optimal performance and may be applied to other non-linear systems for performance optimization. Vehicle drive train efficacy may be improved and fuel efficacy of electric vehicles may also be optimized.

ACKNOWLEDGEMENT

I express my sincere thanks to Er. Amrit Kaur, Assistant Professor Electronics and Communication Engineering Department, Punjabi University Patiala for the Technical Assistance.

REFERENCES

[1] Husain, 'Electric and Hybrid Vehicles', Design Fundamentals, 2nd edition, CRC Press, Taylor and Francis Group, New York, Chap 1, 2011
[2] K.T. Chau C.C. Chan, and C. Liu, "Overview of Permanent magnet brushless drives for electric and And hybrid electric vehicles," IEEE Trans. Ind. Electron ,Vol. 5, No. 6, pp. 2246-2257, jun. 2008.
[3] Ali Emadi , Young Joo Lee, Kaushak Rajashekare, "Power Electronics and Motor Drives in electric, Hybrid Electric, and lug-in hybrid Electric Vehicles", IEEE Transactions on Industrial Electronics, Vol. 5, No. 6, pp. 2237-2245, 2008.
[4] Yoichi Hori, "Motion Control Of Electric Vehicles and Prospects of Supercapacitors", IEEJ Transactions on Electrical and Electronic Engineering, Vol. 4, p. 231-239, 2009
[5] Anil Kumar Yadav, Prerna Gaur, S.K. Jha, J.R.P Gupta and A.P. Mittal, "Optimal Speed Control Of Hybrid Electric Vehicle", journal Of Power Electronics, Vol. 4, pp. 393-400, 2011.
[6] X. Liu, Q. Huang , Y. Chen, and J. Li, "Nonlinear Modelling And optimal Controlling Design for Radar Servo System", Third International Symposium on intelligent Information Technology Application. Pp. 327-330, 2009.
[7] S.H. Kim and C. Park, "A Self Organizing Fuzzy Controller for wheeled Mobile Robot using an Evolutionary Algorithm", IEEE Transactions on Industrial Electronics, Vol. 48, No. 2, pp. 467-474, 2001.
[8] K.M. Junaid and S. Wang, "Automatic Cruise Control Moddeling-A Lattice PWL Approximation Approach", Proceedings Of IEEE Intelligent Transactions system Conference, pp. 1370-1375, 2006
[9] Trebi –Ollennu and J.M. Dolan, "Adaptive Fuzzy Throttle Control for an All-Terrain Vehicle", Proceedings Of Robotics Institute, Paper 198, 1999
[10] S. Omatu, M. Yoshioka, amd T. Kasoka, "PID Control of Electric Vehicle",Third International Conference on Advanced Engineering Computing and Applications in sciences, pp. 157-162, 2009
[11] Zwe-Lee Gaing, "A Particle Swarm Optimization Approach for Optimization Design of PID Controller In AVR System", IEEE Transactions on Energy Conversion, Vol. 19, No.- 2, June 2004
[12] J. Kennedy and R. Eberhart, "particle Swarm Optimization", in proc. IEEE International Conference Neural Networks, Vol. 4, Perth, Australia, 1995,pp-1942-1948
[13] Y. Shi and R. Eberhart, "A Modified particle Swarm Optimizer", in proc. IEEE international Conference Evol. Comput., Anchorage, AK, May1998,pp.69-73.
[14] Y. Shi and C. Eberhart, "Emperial Study Of Particle Swarm Optimization", in proc. IEEE internation Conf. Evol. Comput., Washington,DC, July 1999,pp.1945-1950
[15] D.B. Fogel, Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, 2nd ed. New York: IEEE Press,2000
[16] Seyedali Mirjalili, Siti Zaiton Mohd Hashim, "A New Hybrid PSOGSA Algorithm for Function Optimization",IEEE International Conference on Computer and Information Application ICCIA,pp.374-377, 2010.
[17] K. Ogata, 'Modern Control Engineering', (India: Pearson Education,2002) 681-751