Neural Adaptive Control Strategy for Hybrid Electric Vehicles with Parallel Powertrain

Yusuf Gurkaynak, *Student Member, IEEE*, Alireza Khaligh, *Senior Member, IEEE*, and Ali Emadi, *Senior Member, IEEE*

Electric Power and Power Electronics Center and Grainger Laboratories
Illinois Institute of Technology, Chicago, Illinois 60616, USA
Phone: +1-312-567-8940; Fax: +1-312-567-8976
URL: http://hybrid.iit.edu; E-mails: ygurkayn@iit.edu, khaligh@ece.iit.edu, emadi@iit.edu

***Abstract*—Many theoretical control strategies have been proposed for hybrid electric vehicles (HEVs) during the past decade. Some of these theoretical control strategies are not suitable for real-time applications mainly because of their sensitivity to vehicle parameter variations and different driving habits of the drivers. The computation times of such algorithms are also long because of their high accuracy demand. In this paper, the equivalent consumption minimization strategy (ECMS) is used and a faster solution algorithm is proposed to decrease the computation time while keeping the same accuracy. In addition, a neural adaptive network is proposed to decrease the sensitivity of the algorithm to drive cycle variations with drive cycle recognition.**

I. INTRODUCTION

Hybrid electric vehicles' (HEV) popularity is increasing since their first launch into the market. Each year manufactures announce new versions or improvements for their HEV brands. In the past decade, with the improvement of hybrid technology, many researchers have been focused on control strategies [1], but only few of the proposed strategies are implementable. An implementable control strategy should be stable (should sustain system variables within desirable bounds) and should perform its task (low fuel consumption, low emissions, high performance, etc.) properly within a given period of time, under all circumstances.

Most of the rule-based algorithms match with these requirements and, for that reason, they are preferred by the automotive companies. Despite this operating advantage, these algorithms are not best solutions for the given task. On the other hand, the operating performance of the model based control strategies like equivalent consumption minimization strategy (ECMS) are better than the rule based controllers since model based algorithms are optimized according to the model of the propulsion system components and also according to the drive cycle. However, they are sensitive to the parameter variations in vehicle as well as driver's behavior [2]. Reducing their sensitivity by adaptation algorithms will make them more reliable for real time applications [3]. Another problem is that their computing time is long. Using other solution methods can speed up the progress to make the algorithm more implementable.

In this paper, the main task is to propose a modified model based controller to make it more applicable while sustaining its operating performance by reducing the sensitivity and computation time. The ECMS methodology has been chosen. A neural adaptive network for drive cycle recognition and new improved steepest decent method is introduced for speed increase in computations.

II. POST PARALLEL HYBRID POWERTRAIN MODEL

In our study, we consider a typical large sport utility vehicle (SUV). The electric drive system is connected in parallel to the powertrain after the transmission through a torque coupler. The vehicle houses 6.5 L Diesel engine (120 kW) with a four speed gearbox, electric propulsion system (100kW continuous, 150kW peak), and 316.8V 37.5Ah battery pack. This is the configuration of a full hybrid vehicle. A representative model for this post-transmission parallel topology is given in Fig. 1 [4]. In this figure, purple color shows the speed, red color shows the power, green color presents the characteristic conversation equations (efficiency, fuel consumptions, and battery voltage) for each component of the powertrain.

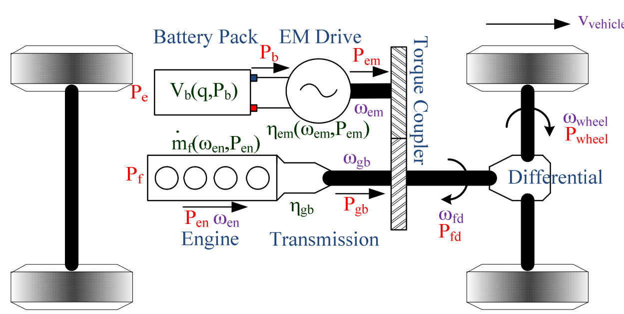


Figure 1. Post-transmission parallel hybrid powertrain topology.

The powertrain can be divided into three main sections according to their dependency on control value. These sections are the common path, the fuel path, and the electric path.

A. Common Path

The common path is the part of the powertrain from the torque coupler to the wheels. This part transfers the total

978-1-4244-8218-4/10/$26.00 ©2010 IEEE

torque from engine and electrical motor to the wheels in order to propel the vehicle. Assuming that wheel speed and wheel mechanical power, which are functions of vehicle physical parameters and drive cycle, are known, the following equations can be derived for the final drive shaft. In these equations, T_{fd} [Nm] is the final drive torque, T_{wheel} is the wheel torque, r_{fd} denotes final drive gear ratio, ω_{fd} [rad/s] stand for final drive rotational speed, ω_{wheel} is the wheel rotational speed, P_{fd} is the final derive mechanical power, and P_{wheel} [W] is the wheel mechanical power.

$$T_{fd} = T_{wheel} / r_{fd} \tag{1}$$

$$\omega_{fd} = r_{fd}\,\omega_{wheel} \tag{2}$$

$$P_{fd} = \omega_{fd} T_{fd} = \omega_{wheel} T_{wheel} = P_{wheel} \tag{3}$$

The electrical path and fuel path share the torque with the coupler with a ratio "u" which is called power split ratio. This is the control variable of the system. If the power split value is zero, vehicle operates like a conventional vehicle and, if the power split value is one, vehicle operates as an all electric vehicle. Selecting correct power split value between zero and one is the key point in propulsion control. This value can also be negative in which engine produces more torque than demand to charge the batteries.

B. Fuel Path

The fuel path is consists of the engine and transmission. The equations for engine mechanical power P_{en}, speed ω_{en}, and torque T_{en} are given below. The shifting strategy is chosen the same way as the conventional one. The r_g is the gear ratio of the transmission for the given shift number. If the losses on transmission are neglected, following equations can be achieved.

$$\omega_{en} = r_g\,\omega_{fd} \tag{4}$$

$$T_{en} = \frac{(1-u)T_{fd}}{r_g} \tag{5}$$

$$P_{en} = \omega_{en} T_{en} = r_g\,\omega_{fd}\,\frac{(1-u)T_{fd}}{r_g} = (1-u)P_{wheel} \tag{6}$$

Engine fuel consumption can be described as a function of the engine rotational speed and engine mechanical power. This function is one of the characteristic maps of the engine. The consumed fuel power can be calculated by multiplying the mass fuel rate with lower heating value of diesel (H_l).

$$P_f = \dot{m}_f(\omega_{en}, P_{en})H_l = \dot{m}_f\left(r_g r_{fd}\,\omega_{wheel}, (1-u)P_{wheel}\right)H_l \tag{7}$$

C. Electric Path

The electrical path consists of the electrical machine, power electronic driver, and battery pack. The electrical machine shaft is directly coupled to the torque coupler whose turns ratio is one. Subsequently, the speed of the electric motor is the same as the final drive shaft rotational speed and torque directly is applied to the final drive shaft. The post-transmission topology lets the electric motor operate independently from gear shifting and gives flexibility in control to operate the machine in more efficient areas. The following equations describe electrical machine speed ω_{em}, torque T_{em} and power P_{em}.

$$\omega_{em} = \omega_{fd} \tag{8}$$

$$T_{em} = uT_{fd} \tag{9}$$

$$P_{em} = \omega_{em} T_{em} = uP_{wheel} \tag{10}$$

The battery power P_b demand can be calculated by dividing the mechanical power of the electric machine with the efficiency of the machine, which is a function of the engine speed and mechanical power. This efficiency map η_{em} includes the losses in machine (electrical and mechanical losses) as well as the power electronic driver.

$$P_b = \frac{P_{em}}{\eta_{em}(\omega_{em}, P_{em})} = \frac{uP_{wheel}}{\eta_{em}(r_{fd}\,\omega_{wheel}, uP_{wheel})} \tag{11}$$

The electrical power P_e from the battery storage can be calculated by using the efficiency map η_b of the battery (11). This map defines the losses on the series charging or discharging resistance depending on the battery current direction.

$$P_e = \frac{P_b}{\eta_b(q, P_b)} = \frac{uP_{wheel}}{\eta_b\left(q, \dfrac{uP_{wheel}}{\eta_{em}(r_{fd}\,\omega_{wheel}, uP_{wheel})}\right)\eta_{em}(r_{fd}\,\omega_{wheel}, uP_{wheel})} \tag{12}$$

Efficiency of the battery depends on both battery power and state-of-charge (SOC) q. The SOC is a function of cumulative battery current I_b during any time period. At any given time, the battery current can be calculated similar to equation (13). V_{oc} is the open circuit voltage of the battery, which is a function of the SOC. The state-of-charge variation can be computed from this current value through equation (14). Q_0 is the nominal capacity of the battery pack [Ah].

$$I_b = \frac{P_e}{V_{oc}(q)} \tag{13}$$

$$\frac{d}{dt}q(t,u) = -\frac{I_b(t,u)}{Q_0} \tag{14}$$

III. Equivalent Consumption Minimization Strategy (ECMS)

The main concept in hybrid electric vehicle control is to end the drive cycle with zero electrical energy usage. Ideally, this means the final SOC value is the same as the initial SOC value for a drive cycle (charge sustaining). In reality, because of unknown driving behaviors, the final SOC can differ from the initial one. This difference reflects the additional or less electrical energy usage from the battery pack. The equivalent consumption minimization strategy is an optimization method based on minimizing the fuel as well as the electrical usage while meeting the demand. A weighting coefficient makes consumed fuel power and electrical power comparable to each other. If the weighting function is chosen properly, the final state-of-charge and initial state-of-charge will be the same (charge sustaining) at the end of drive cycle [5].

A. Cost Function

The main focus of this control strategy is the fuel consumption of engine and equivalent fuel consumption of battery electrical energy. The following cost function J_f defines such an approach. In this equation, ζ (.) is a function which converts the battery electrical energy into fuel energy equivalent.

$$J_f = \int_{t=0}^{t_f} P_f(t,u)dt + \zeta\left(\int_{t=0}^{t_f} P_e(t,u)dt\right) \tag{15}$$

The main task is to calculate proper control values, which will minimize this cost function for each time instance. This optimization problem can be solved by using Pontryagin's Minimum Principle [6]. Without considering the final state SOC, a sub optimal Hamiltonian function can be defined as in equation (16). In this equation, *s* is the equivalent coefficient, which depends on the drive cycle. This coefficient can be calculated by using dependency curve for a given drive cycle. This curve can be generated by simulating the model with constant control values, measuring the fuel, battery energy usage and doing the same job for different control values within admissible boundaries (Fig. 2). In Fig. 2, points show each simulation result and lines are fitted lines through these points. Slope of these two lines gives the *s* values for charging and discharging situations. Table I shows the equivalent coefficients for UDDS drive cycle.

$$H(t,u) = P_f(t,u) + s\ P_e(t,u) \tag{16}$$

The power split value (u) is found by minimizing equation (16).

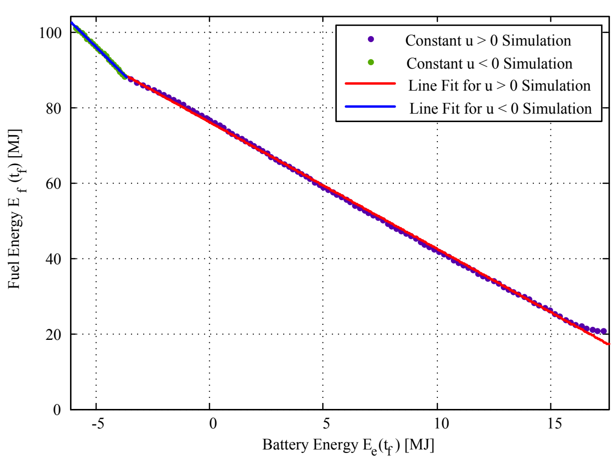


Figure 2. Dependency curve for UDDS drive cycle.

TABLE I
EQUIVALENT COEFFICIENT FOR UDDS AND HWFET DRIVE CYCLES

Drive Cycle	u_{max}	u_{min}	s_{dis}	s_{chg}
UDDS	1.0	-0.2	6.018	3.369

B. Penalty Function

While solving an optimization problem, the boundary constraints should also be taken into consideration. Computing separately will be hard and may cause problems. Penalty functions are used to implement the constraints into the cost function to be minimized to consider only one function for optimization [7]. This function adds additional positive values to the original cost function when any constraint is violated. The constraints for a hybrid vehicle are given below (17).

$$\begin{aligned} &T_{en} < T_{en.max}(\omega_{en}) \\ &T_{em} < T_{em.max}(\omega_{em}) \\ &-I_{chg.max} < I_b < I_{dis.max} \end{aligned} \tag{17}$$

The penalty function defined for this study is given in equation (18). In this equation, u_{lim} is the u value in which one of the constraints is violated. u_{lim} is reset when the violation is ended. This is an exponential function that increases depending on how much the boundaries are violated. The K value is chosen depending on the average value of the original cost function and desired slope of the penalty function.

$$g(u) = K\left(e^{(u-u_{lim})^2} - 1\right) \tag{18}$$

$$H_{\text{ext}}(u) = H(u) + g(u) \tag{19}$$

The original Hamiltonian function is extended according to equation (19).

IV. Fast Solution Algorithm

The Hamiltonian function to be minimized has nonlinear and complex structure because of the efficiency maps of the engine, battery, and electric machine. Although the function is convex for most of the driving points, the traditional derivative minimization algorithms are insufficient to solve this optimal problem.

Brute force technique is one of the fundamental ways to solve this problem. It basically tries all possible values of the control variable and calculates the cost function for each of them. Then, it selects the optimum control value which gives the minimum cost. The accuracy of this method depends on the number of chosen control values within a give boundary. As the number of tries increases, the accuracy is improved; however, the computing load for the processer increases as well. This load increase slows down the control algorithm. A fast but accurate solution is required for proper result.

One of the fast optimization methods is the steepest decent algorithm which is suitable for convex function minimization. This algorithm uses the first derivative of the function to be optimized as the slope of the iteration. As the iteration closes to the minimum point (optimum point), the slope approaches to zero and finalizes the progress in finite time. This iterative method is given in (20). In this equation, the subscripts represent the iteration index. The f(.) is the function to be minimized and u is the argument of this function. α is the coefficient to regulate the speed and stability of the algorithm. Higher values increase the speed but may threat the stability.

$$u_k = u_{k-1} - \alpha\, \Delta f(u_{k-1}) \tag{20}$$

If it is not possible to obtain the first derivative of the function, virtual first derivative can be formed as in equation (21).

$$u_k = u_{k-1} - \alpha \frac{f(u_{k-1}) - f(u_{k-2})}{u_{k-1} - u_{k-2}} \tag{21}$$

In [8], a faster algorithm by using recurrent neural network has been proposed. This new algorithm is similar to the steepest decent algorithm except the definition of derivative. This method defines a new function sig(.) as in (23) and replaces it with the derivative (22).

$$\frac{du}{dt} = -\alpha \frac{\partial \mathrm{f(u)}}{\partial \mathrm{u}} \quad => \quad \frac{du}{dt} = -\alpha\, sig\left(\frac{\partial \mathrm{f(u)}}{\partial \mathrm{u}}\right)^{v} \tag{22}$$

$$sig(r)^{v} = sign(r)\,|\,r\,|^{v} \tag{23}$$

This algorithm has two tunable parameters (α and v), which give flexibility to increase the convergence speed while sustaining the stability. Fig. 3 compares the solution speeds of both algorithms. In this figure, the optimization process for 585th second of the UDDS drive cycle is given (blue line); part (a) and part (c) show the optimization done with steepest decent algorithm, while part (b) and part (d) show the optimization done with recurrent neural network approach. To compare fairly, the initial values are chosen the same. The initial value for (a) and (b) is -1 and the initial value for (c) and (d) is 1. Each red dot represents the iteration step. From this figure, it can be observed that, independent from the initial value, number of iterations for recurrent neural network based algorithm is lower, which makes this algorithm faster compared with the steepest decent algorithm. In this study, recurrent network based algorithm is used for optimization.

V. Adaptation with Neural Network

The second important task is to choose correct equivalent coefficient for the Hamiltonian function. As maintained, each drive cycle has a unique pair of equivalent coefficients (for charging and discharging modes). For that reason, the control algorithm should be adaptable to changes in driving condition and recognize the drive cycle. Neural networks are one of the methodologies used for pattern recognition with excellent mapping capability. It has been proved that any nonlinear lookup table can be fitted inside a proper network.

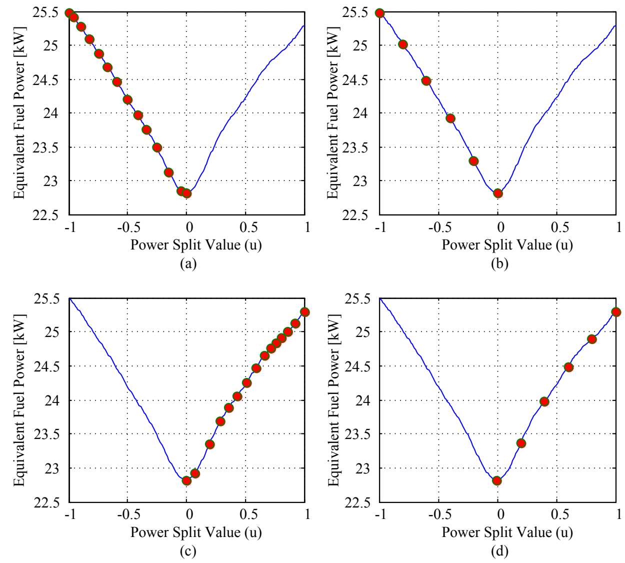


Figure 3. Comparison of speed of minimization algorithms.

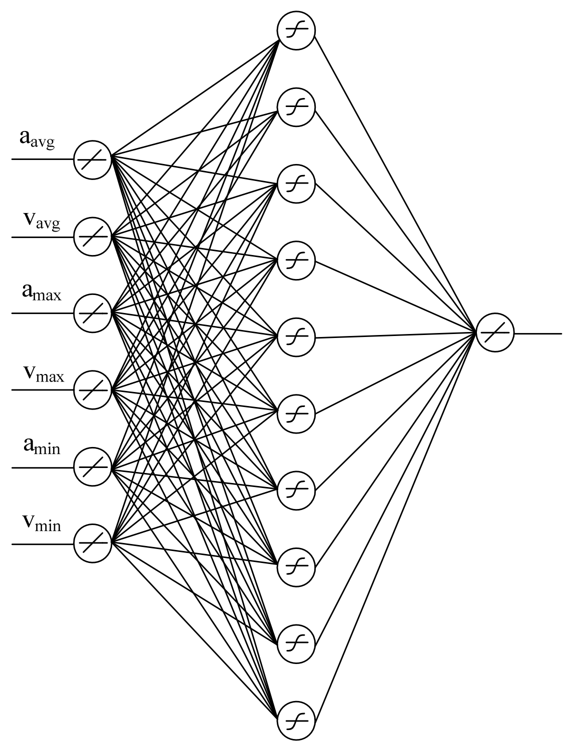
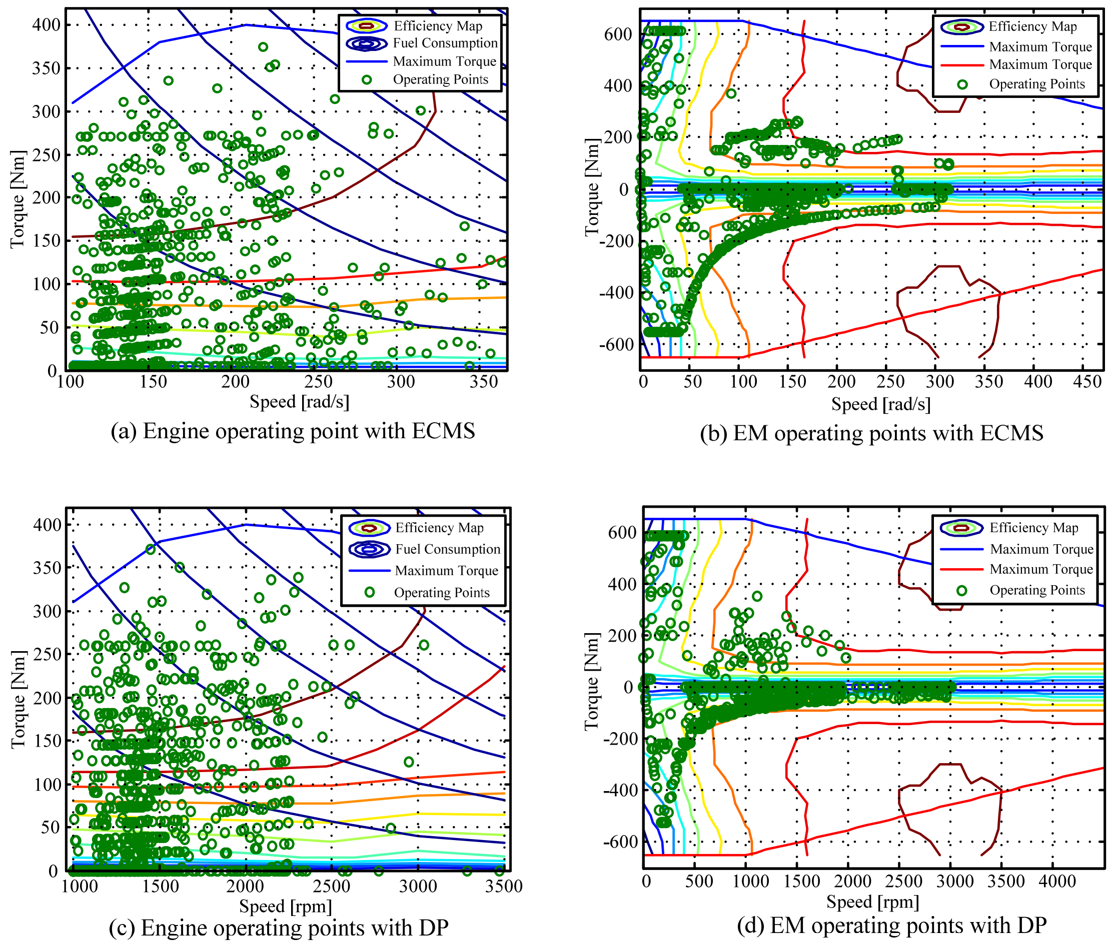


Figure 4. Neural network for drive cycle recognation.



(a) Engine operating point with ECMS

(b) EM operating points with ECMS

(c) Engine operating points with DP

(d) EM operating points with DP

Figure 5. Simulation results for both dynamic programming and EMCS.

TABLE II
FUEL CONSUMPTION FOR DIFFERENT CONTROL METHODS

Control Method	Fuel Consumption (mpg)	Final SOC
Conventional	11.40 mpg	----
Dynamic Programming	14.41 mpg	%60
ECMS	14.42 mpg	%58

A recurrent neural network has been proposed as the drive cycle recognizer and equivalent coefficient tuner [9]. This network tracks the last 200s computed speed data (average, maximum, minimum speed, and acceleration) which are the characteristic values for a drive cycle. This network is shown in Fig. 4. In this figure, circles represent the transfer function for each neuron and lines between these circles represent weighting between the layers. Network has 6 input neurons in input layer, 10 neurons with sigmoid function in hidden layer, and one neural output layer. The output value codes the type of the drive cycle. According to this code, proper values are selected to tune the algorithm. Fig. 6 shows the operation of the network. The network is tested with a combined HWFET and UDDS drive cycle. It can be seen from the figure that, when the cycle is UDDS, the network gives zero and, for HWFET, it gives one at the output.

VI. SIMULATION RESULTS

The model has been simulated with both dynamic programming (DP) and ECMS with the proposed solution method [10]. The dynamic programming considers all possible scenarios to drive the vehicle for a given drive cycle and selects the best scenario with lowest cost. This means the solution by DP is always the most optimum solution. Unfortunately, the DP is not implementable, but it is usually used to compare with the other control strategies. The simulation results for DP and EMCS are given in Fig. 5. In this figure, engine and electric drive operating points are shown over the efficiency maps of them. Cold colors show lower efficient and hot colors show high efficient operating areas. Similarity of the figures proves that ECMS algorithm has been solved properly. Furthermore, the results given in Table II show how much the sup-optimal solution is close to the optimal solution.

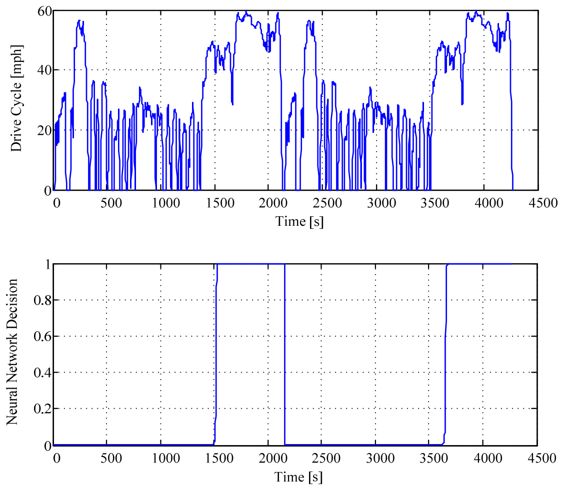


Figure 6. Network output for UDDS and HWFET drive cycle.

VII. CONCLUSION

One of the important concerns in designing control strategies for hybrid electric vehicles is their implementation in real applications. The controllers with better performance are usually very sensitive to vehicle parameters and conditions. Therefore, the accuracy of the calculated value for power split is important. More accurate algorithms mean slower computing times, which are not suitable for real-time applications. In this study, a fast and accurate solution method has been proposed and implemented to ECMS, which is one of the model based controllers. The simulation results for dynamic programming, which shows the optimum solution, and ECMS are compared. The similarity shows the solution algorithm is performing properly. Furthermore, a neural adaptive network is proposed to tune the equivalent coefficient. This network basically recognizes the drive cycle and chooses proper values for ECMS.

REFERENCES

[1] Y. Gurkaynak, A. Khaligh, and A. Emadi, "State-of-the-art power management algorithms for hybrid electric vehicles," in *Proc. IEEE Vehicle Power and Propulsion Conference,* Dearborn, MI, pp. 388-394, Sept. 2009.

[2] G. Paganelli, S. Delprat, T. M. Guerra, J. Rimaux, and J. J. Santin, "Equivalent consumption minimization strategy for parallel hybrid powertrains," in *Proc. IEEE 55th Vehicular Technology Conference*, vol. 4, pp. 2076-2081, Aug. 2002.

[3] P. Pisu and G. Rizzoni, "A comparative study of supervisory control strategies for hybrid electric vehicles," *IEEE Transactions on Control Systems Technology,* vol. 15, no.3, pp. 506-518, May 2007.

[4] A. Sciarretta, M. Back, and L. Guzzella, "Optimal control of parallel hybrid electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 3, pp. 352-363, May 2004.

[5] L. Guzzella and A. Sciarretta, *Vehicular Propulsion Systems: Introduction to Modeling and Optimization,* Berlin Heidelberg, Springer, ISBN: 3-540-25195-2, 2005.

[6] D. E. Kerk, *Optimal Control Theory: An Introduction*, Englewood Cliffs, N.J., Prentice-Hall, ISBN: 0-486-43484-2, 1970.

[7] D. P. Bertsekas, "Globally covergent Newton methods for constrained optimization using differentiable exact penalty functions," in *Proc. 19th IEEE Conference on Decision and Control Including the Symposium on Adaptive Processes*, vol. 19, pp. 234-238, Dec. 1980.

[8] L. Cheng, Z. G. Hou, N. Homma, M. Tan, and M. M. Gupta, "Solving convex optimization problems using recurrent neural networks in finite time," in *Proc. International Joint Conference on Neural Networks,* Atlanta, GA, pp. 538-543, June 2009.

[9] S. I. Jeon, S. T. Jo, Y. I. Park, and J. M. Lee, "Multi-mode driving control of a parallel hybrid electric vehicle using driving pattern recognition," *Journal of Dynamic Systems, Meusurement, and Control,* vol. 124, pp. 141-149, March 2002.

[10] O. Sundstrom and L. Guzzella, "A generic dynamic programming Matlab function," in *Proc. 18th IEEE International Conference on Control Applications*, St. Petersburg, pp. 1625-1630, July 2009.