ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A Methodology to Design an Efficient EM Controller with High Practicability in HEVs Learning

Ehsan Ghasemimoghadam*1, Kazuhide Togai2, Hisashi Tamaki1
[*1] Graduate School of System Informatics, Kobe University, Kobe, Japan
[2] Department of Mechanical Engineering for Transportation, Osaka Sangyo University, Osaka, Japan

DOI: 10.5281/zenodo.1214393

ABSTRACT
This paper presented on the methodology of designing intelligent energy management for hybrid electric vehicles (HEVs). This work outlines the use of the deep neural network (DNN) to design learning-based EM controller, which provides a powerful new framework to control the HEV system while improving the HEV performance. The framework utilizes the DNN technology to inference the new knowledge from the non-causal energy optimization results. Herein, we present intelligent energy controller, which was trained to control the HEV system by represented the current state of the system on preset several driving cycles. In order to prepare the data points, the near-optimal solutions are saved in the data store which obtained by the offline optimization processes at various starting SoC. Through computational examples on the series-parallel structure, the designed controller would be evaluated with a canonical rule-based and the non-causal optimized control strategy that fuel consumption improvement on the proposed EM controller observed and the effectiveness of the proposed approach in the designing high practicable EM controller was confirmed.

KEYWORDS: deep neural network (DNN), energy control system, energy management (EM), hybrid electric vehicle (HEV), machine learning (ML)

I. INTRODUCTION
Nowadays, the energy management (EM) of the hybrid electric vehicles (HEVs) system with the goal of performance improvement is a challenging problem, because of the hybrid-power-source nature [1]. To meet this challenge, it is important to optimize the energy flow through the HEV by designing some strategies. The EM strategies is implemented by a EM controller, which controls the energy flow between all propulsive components while improving the HEV performances such as fuel economy. The desired EM controller as a closed-loop control system observes the current states of the system based on the sensory information and by a sub-optimal control logic drives the actuators that help to improve the HEV performance.
The main favored and promoted approaches had been proposed on the EM of HEVs to improve HEV performance it can be classified into two main categories: 1) heuristic approach 2) optimization-based approach. The popular approach to design EM controller in real-time condition is based on heuristic methods which use rule-based or map-based techniques. These strategies are based on the human experiences, the analysis of power flow in HEV, input-output functions and characteristics of the main propulsive components. These heuristic-based methods lead to achieve robust strategies. However, they cannot achieve the actual optimal solution.
Another approach is to utilize the optimal control techniques via the predefined objective function(s) are known as optimization-based energy management strategies. In general, this approach needs the significant computational demand. This approach as the viewpoint of driving knowledge is classified into non-causal and causal optimization-based EM sub-approaches. The former sub-approaches need the full and as well as a priori knowledge of driving mission that, reasonably, cannot directly applicable to real-time control system of the HEV for achieving to the optimal performance. On the contrary, to incorporate knowledge about driving patterns in real conditions, the predictive information from driving situations in near future is considered into the vehicle energy management strategies, which the optimization process is performed in real-time. However, the exact demand prediction in the real-time controller is difficult and directly related to the accuracy of the prediction information. Furthermore, the non-causal optimization-based EM approach can lead to derive new knowledge for practical conditions and, therefore, designing the EM controller without considering the complex behaviors caused by

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

human drivers. This kind of approach can be realized by the metaheuristic or strict optimization techniques. The metaheuristic techniques can perform the optimal seeking process around an existence solution which can be produced by heuristic methods. This is the main motivation of this research to achieve robustness-optimized strategies. However, the latter techniques can yield the global optimal solution but the reliability of the extracted knowledge in real conditions is lower than the metaheuristic techniques. Moreover, the strict optimization techniques generally need more computational time.

The overall target of this research is proposing a methodology for designing efficient EM controller with high practicability with considering the optimality and implementation aspects via deriving causal knowledge from the non-causal optimization-based EM strategies, which can be applied for any kind of HEV. For this purpose, for balancing between optimality and implementation aspects a new approach for join engineering-oriented operations based on some policies and utilization of optimization techniques was proposed in [2]. Herein, an improved EM controller design framework is proposed. The Fig. 1 shows the conceptual diagram for designing the improved EM controller. At the first phase, a multi-stage optimization problem is designed to find optimal solutions via a well-defined objective function. It can be used for designing non-causal control strategies that was presented in the first paper. Then, a learning-based technique for considering the whole set of features simultaneously instead of manually analyzing the significance of individual features or a small set of features is proposed. The use of machine learning (ML) techniques can save labor-intensive manual process to derive the improved strategy for energy management and as well as by analyzing the optimal or near-optimal data can extract relationships and features. For this sake, the non-causal near-optimal or optimal solutions and related system state trajectories are saved in the data store where are used as the ML data for training and testing the learner as a supervised learning problem. It is our contribution to use the non-causal optimization-based control strategies and train an EM system using the deep neural network (DNN) technology to emulate the non-causal near-optimal solutions generated by an evolutionary computing technique. In the phase 3, the EM controller receives the immediate reaction of the HEV system at each time instance, i.e. short-term knowledge, and then governs the system.

In this paper, to verify the designed EM controller based on the proposed methodology a case study is established. At the first, a systematic method for training EM controller and selection the suitable NN model is realized. The designed EM controller is verified with a canonical control strategy and the non-causal near-optimal control strategy which generated by the proposed optimization techniques in [2]. Consequently, the effectiveness of the proposed method for designing EM controller with high practicability is confirmed.

This paper is organized as follows: Section II introduces the methodology of data preparation for learning. Section III presents the deep neural network technology that we realized for learning-based EM controller. Section IV states the methodology of DNN utilization. The simulation environment, data point's specifications, DNN utilization, an overview of the results and, finally, the learning-based EM controller is used for rule extraction that can be implemented in real-time situations are presented in section V. Section VI concludes this paper.

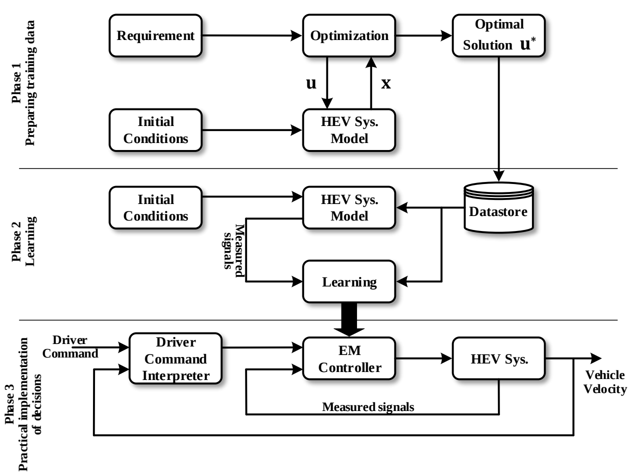

Fig. 1. Conceptual framework of designing the improved EM controller

II. METHODOLOGY OF DATA PREPARATION

As mentioned before, the ML system processes some datasets for learning and as well as evaluating the obtained models. The required data are prepared by the results of the optimum seeking which founded by optimization process that presented in [2]. The first phase in Fig. 1 illustrates the workflow for gathering the data points.

Since the learned model does not have the ability to accurately extrapolate beyond the input space, the ML data are prepared in such a way the input space is covered as possible as. The ML system data are influenced by the initial conditions, which consist of the initial states $\mathbf{x}_0$ of the HEV system and the modelling issues such as driving cycles. However, different HEV structures make various computational complete models that one specific EM controller cannot fulfil the desired HEV performance for all HEVs. To observe the mentioned principle, a diversification in configuration of the initial conditions should be considered. Thus, for preparing the data for ML system, however, the initial conditions can differ but the actual treatment for learning is same and lead to estimate accurately the final model.

Data Preparation

The results of several optimization processes comprise of multi-time-series of discretized decision (action) variables $\mathbf{u}$ as

$$\mathbf{U} = \{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_m\} \quad , \qquad \mathbf{U} \in \mathcal{U} \subset \mathbb{R}^{m\times k} \tag{1}$$

where m is the total number of the optimal solutions obtained by several optimization processes, and $\mathbf{u}$ is subset of the k-dimensional space ($\mathbf{u} \in \mathbb{R}^k$). Besides, the measured signals $\mathbf{x}$ of the HEV system based on the optimal solutions are

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\} \quad , \qquad \mathbf{X} \in \mathcal{X} \subset \mathbb{R}^{m\times l} \tag{2}$$

where $\mathbf{x}$ is subset of the l-dimensional space ($\mathbf{x} \in \mathbb{R}^l$). Generally, the EM optimization problem processes as a finite multistage optimization problem where is limited on finite-state and finite-action that leads to define finite state-action pairs. In other words, the ML system data are defined by a tuple of a vector-valued feature variable, i.e. the measured signals vector $\mathbf{x}$, and a vector-valued target response, i.e. the decision vector $\mathbf{u}$ as

$$\mathbf{X} = \{\langle \mathbf{x}_1, \mathbf{u}_1 \rangle, \langle \mathbf{x}_2, \mathbf{u}_2 \rangle, \cdots, \langle \mathbf{x}_m, \mathbf{u}_m \rangle\} \tag{3}$$

Finally, each tuple makes a data point for supervised learning process. The measured signals of the HEV system consists of the propulsive components characteristics (e.g., the SoC, which reflects charge level of battery) and the vehicle velocity trajectory depending on how large the state space is while the results have reasonable accuracy in the EM studies. The actuations may be the set-points of the main propulsive components. Furthermore, the power split situations where the engine is on is the most important opportunity to select the appropriate data points.

III. DEEP NEURAL NETWORK

The deep neural networks (DNNs) technology is a sub-field of the ML is realized for learning process via adjusting the parameters of neural network layers, i.e. weights [3]. The weights make a correlation type between the measured signal(s), i.e. input, and the actuation(s), i.e. output.

The most well-known network architecture in DNN is the feedforward multilayer neural network [4]. The feedforward multilayer neural network consists of an input layer, one or many hidden layers, and a single output layer. Each layer can have a different number of artificial neurons. The Fig. 2 depicts the standard feedforward multilayer neural network that artificial neurons in each layer represented by the circles that each neuron is connected to all artificial neurons in preceding layer, so-called fully connected layer. The information flows in feedforward multilayer neural network from input layer to output layer without any feedback connection.

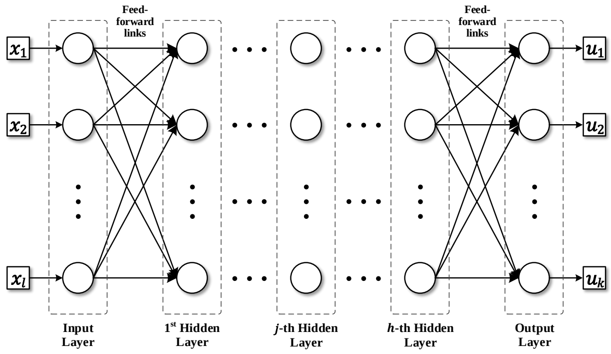

Fig. 2. Feedforward multilayer neural network concept

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Learning DNN Procedure

The process of learning in artificial neural network (NN) is iteratively adjusting the weights and biases and, thereby, allocating significance to certain bits of information and minimizing other bits that lead to learn model which features are tied to which output, i.e. target responses.

The procedure can be classified into two phases: forward phase and backward phase. The DNN technology applies the supervised learning in forward phase, whereas, backpropagation technique calculates and then evaluates the error rate as the performance function and, finally, to update weights and parameters, autonomously. The objective is to minimize the error rate in backward phase. Different error rate formulas to define performance index were proposed. Typically, the performance function J of back-propagation neural network in the DNN technology based on batch idea, which the train dataset is applied to the network before the weights are updated, is mean square error (MSE) measure and using as

$$J = \frac{\sum_{i=1}^{m}\sum_{j=1}^{k}\left(\hat{u}_{ij} - u_{ij}\right)^2}{km} \qquad (4)$$

where $\hat{u}$ is the output of the model and u is the target response. m and k are the size of dataset and target response, respectively.

Learning stops when the algorithm achieves an acceptable level of performance such as reducing error or stagnation in the search process in hypothesis space.

It is important, to balance between minimizing the error rate on the training dataset and generalizing the model so that the use of overfitting prevention techniques is inevitable. The proposed idea in this work is using the manual regularization and early stopping methods together throughout in this research is called ESL strategy. However, the early stopping and the regularization methods individually improve the generalization of the NN model but the suitable selection of the maximum fails on validation dataset in early stopping method and, besides, the appropriate the manual regularization ratio is very difficult. Thereby, increasing the maximum fails of validation data set in the early stopping method and decreasing the regularization ratio, leads to overfitting.

With considering the regularization method, the performance function J of learning optimization algorithm is changed so that the additional term for penalizing large weights is added. The advantage of this method is to obtain a smoother NN model with better generalization. Therefore, the modified performance function J_{R} is of the form

$$J_{\mathrm{R}} = (1-\gamma)J + \frac{\gamma}{2}\sum_{i=1}^{n_w} w_i^2 \qquad (5)$$

where n_w is the total number of weights of the network. γ is the regularization ratio and determines the protection against overfitting level is between 0 and 1.

In the other hand, the validation dataset is used to stop training early if the NN model performance on the validation data set fails to improve or remains the same for a pre-defined epoch's number that so-called early stopping method.

IV. DNN UTILIZATION METHODOLOGY

The aim of this section is proposing a way to utilize the DNN technology on non-causal EM data. Various designs of the DNN (with standard fully connected feedforward architecture) realization by the mentioned data points can be considered. The problem in supervised learning is to find an adequate NN model and to tune its parameters. For finding an efficient NN design for EM controller, some issues should be considered that will be discussed in the following.

Dataset's definition: The ML system data points typically split up to train and test datasets. Due to the training algorithm and the strategy of generalizing model, the proportion of the train dataset is isolated as the validation dataset. The train dataset is used for training the hypotheses, whereas, the validation data are employed to evaluate the NN designs by calculating the error rate. In addition, the test dataset does not have any effect on the training process. The ratios of the validation and test datasets are, generally between 10 and 20%. In section II, the process of ML data preparation was stated. The data points are obtained by the finite sequences of time-series of offline optimization results. Therefore, during datasets configuration, the randomization of data points helps to generalize well the resulting network. In other hand, regarding to the power flow in the HEVs, the mentioned datasets are comprised to the proportional ratio of the operating modes, which split up power between energy sources. Fig. 3 illustrates the workflow of the making ML datasets.

NN topology: The exact number of hidden layer and nodes is a challenging issue that has a profound impact on learning results. Various research efforts have been made to propose several methods such as trial and error,

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

heuristic search, exhaustive search, pruning and constructive techniques [6]-[10]. However, the empirically-derived rule-of-thumb methods lead to obtain the NN model with inefficient accuracy. The authors suggest to use trial and error by designing several computational experiments. In order to evaluate the obtained structures, the statistical hypothesis testing methods would be applied.

Training algorithm: Briefly, the training algorithms adjust the parameters of resulting network with considering to minimize the error performance function. They are classified into heuristic and standard numerical methods [5]. The heuristic methods apply some modifications on backpropagation technique such as using momentum or variable learning rate. However, the speed of convergence is significance but additional learning parameters and also failure to convergence in some problems cause to rarely promote in learning problems. On the contrary, the popular and favored methods are based on the standard numerical methods, which is divided to Jacobian-based and gradient-based techniques. However, the performance of calculation and the quality of resulting network affected by the size of network and the characteristics of problem. Thereby, by designing the computational experiments, the appropriate algorithm is revealed.

Neuron's parameters initialization: Generally, the initial values of neurons parameters are set by randomization techniques with symmetric feature between hidden layers. The Nguyen-Widrow initialization method generates the initial values for a layer so that the active regions of the layer's neurons are uniformly distributed approximately evenly over the input space [11]-[12].

The appropriate active functions of layers for such kind of regression problems are utilized by the hyperbolic tangent sigmoid transfer functions for hidden layer(s) and the linear transfer functions at output layer [4].

Finally, the normalization process on the data points not only helps to improve the learning process, it also avoids the overfitting issue. This process is utilized to both the input and output of the network. Therefore, the unnormalized process is performed on the normalized values at output layer to fall into the units of the original target feature.

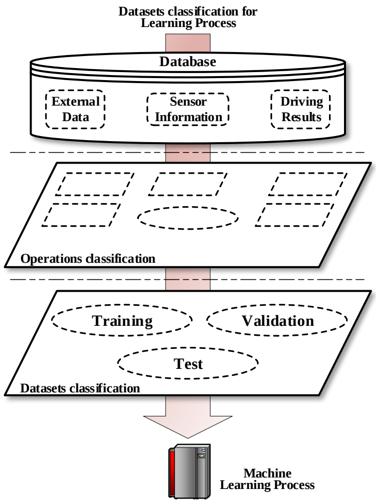

Fig. 3. The proposed scheme of separation concept of the learning data

V. CASE STUDY

In this section, we realized the DNN technology to learn the non-causal near-optimal EM solutions for four standard driving cycles, i.e. Japanese cycle (JC08), Federal test procedure (FTP-75), new European driving cycle (NEDC) and Worldwide Harmonized Light Vehicle Test Procedure (WLTP class 3), so that the generalized knowledge is applied to design EM controller through deep neural network. Then, the obtained NN model is verified with a canonical control strategy under two standard driving cycles, i.e. Artemis (urban) and FTP-72. Regarding to the proposed framework, the realized DNN emulates the near-optimal or optimal management strategy as dictated by local search or evolution strategy algorithms for the current conditions in a way that can be utilized in practical situations. The JC08, FTP-75 and WLTP (class 3) cycles are chosen because they represent well the real traffic conditions, whereas, the NEDC was theoretically designed to assess the fuel consumption for light vehicles.

The non-causal near-optimal EM solutions of a series-parallel HEV, which denoted by SPHEV system, by the proposed optimization algorithms in [2] are obtained. The detailed description of this vehicle and the framework of the modelling were presented in [2].

As mentioned before, the feature and target data comprise the set of learning data. The feature data include 5 measured (feature) signals while the target data comprise of two decision variables, i.e. response signals. To realize the DNN technology for the learning-based EM controller, the vector-valued feature signals set are

$$\mathbf{x} = \{C_{\mathrm{b}}, \omega_{\mathrm{e}}, V, P_{\mathrm{tc}}, a\} \tag{6}$$

where C_{b} is the SoC level of the battery; ω_{e} is the engine speed; V is the vehicle speed; P_{tc} is the driver's power demand and a is the vehicle acceleration. The response, i.e. actuations, variables arranged as

$$\mathbf{u} = \{T_{\mathrm{e}}, T_{\mathrm{em2}}\} \tag{7}$$

where T_{e} and T_{em2} indicate the engine and EM2 torques, respectively.
The optimization algorithms to every driving cycle was applied multiple times so that multiple near-optimal trajectories were generated that way in each time the initial conditions are different. In this case study, only initial SoC of battery is changed between 45 and 75% by step 5, whereas, the final SoC was set as 60%. According to the section II, this diversification in the initial conditions leads to diversify in input space. Table 1 illustrates the results of optimization algorithms for finding the non-causal near-optimal EM strategies. The solution criterion of the best optimized solution at each initial SoC and under each driving cycle is based on the minimal fuel consumption among three proposed optimization search algorithms. Fig. 4 shows the trajectories of near-optimal battery SoC obtained under the WLTP (class 3) cycle. The same procedure was applied to every mentioned standard driving cycle. Finally, the trajectory of HEV-system state data, i.e. feature data, and the time series of decision variables, i.e. target response, for every mentioned driving cycle and initial conditions were generated. In addition, the training process and the NN designs evaluation process are run on Matlab platform with Core i5-4460 processor and 8GB DDR3 RAM.

Table 1. The diversification of the DNN experiments

Drive Cycle	Initial SoC ($C_{\mathrm{b},0}$)						
	45%	50%	55%	60%	65%	70%	75%
NEDC	564.5	541.39	524.69	506.07	482.26	461.04	441.65
JC08	370.2	345.052	319.921	298.219	275.101	253.741	230.861
FTP-75	780.818	750.155	735.972	715.033	688.998	668.162	648.543
WLTP (class 3)	1092	1066.9	1048.4	1029	1007.8	986.148	961.962

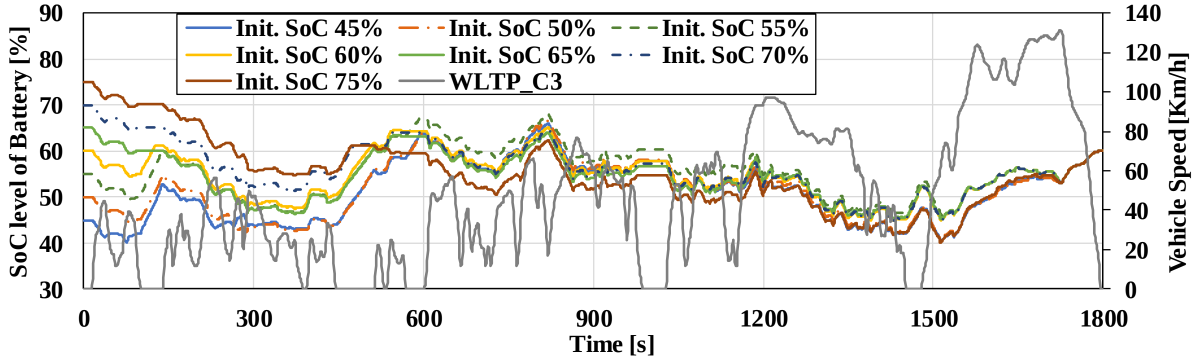

Fig. 4. The near-optimal SoC trajectories on WLTP (class 3) cycle

Selection of an efficient DNN design is a challenging problem in the learning. Thereby, several computational experiments would be established. The experiments are constructed so that the suitable NN design is obtained for the SPHEV system. The diversification of the experiments is based on

- The training algorithms;
- The number of hidden layers;
- The number of artificial neurons n_h (nodes) in the hidden layer(s)

Table 2 shows the diversification of these computational experiments. The experimented network topology includes one and two hidden layers. In the one hidden layer network, the nodes number changes between 5 and 80 by step 5. Whereas, for two hidden layers structure, the nodes of each layer changed between 10 and 50 by step 10. In addition, the results obtained by the regularization ratios (γ), 0.1, 0.5, 0.9 and 0.95. The one hidden

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

layer networks totally comprise to 28 experiments, whereas, the two hidden layer's networks consist of 50 experiments. Each experiment represents a NN design, which runs ten times.

As the viewpoint of numerical approaches of optimization techniques, two learning algorithms are utilized in the DNN experiments. The desired algorithms support training dataset with validation and test datasets. The first applied algorithm is Levenberg-Marquardt (LM) algorithm that is Jacobian-based. Usually, the LM algorithm can obtain lower MSE than any of the other algorithms with significantly convergence rate, whereas, the scaled conjugate gradient (SCG) algorithm is based on conjugate directions without using a linear search at each iteration [5], [13]. The below training termination criterions are considered as follows

- The validation performance criterion, i.e. validation stop, which if it increased more than specific times since the last time it decreased, throughout in this experiment set, the maximum validation failures are defined as ten.
- The regulation performance criterion by the adaptive parameter μ, which is reached to maximum value implied to the convergence in training process. Throughout this case study, the maximum μ is equal to 1E10.

Table 2. The diversification of the DNN experiments

Experiment's Set Name	No. Hidden layer(s)	Training algorithm	No. Nodes ($\mathbf{n}_\mathrm{h}$)
ESL_LM_2L	1	LM	5 ~ 80
ESL_LM_3L	2		< 10~50,10~50 >
ESL_SCG_2L	1	SCG	5 ~ 80
ESL_SCG_3L	2		< 10~50,10~50 >

The utilized randomization algorithm in this case study is defined by Twister algorithm so that the non-repeatability values during multiple-times starting the training algorithm are generated and, hence, the results can be treated as statistically independent [14]. Throughout in the DNN experiments set, the Nguyen-Widrow initialization method for weights is utilized. Besides, the normalization process is performed by the min-max method so that they fall approximately in the range $[-1,1]$.

The number of unknown parameters is obtained by the number of artificial neurons of network. If $\mathbf{n}_\mathrm{h}$ indicates the vector-valued of nodes number of neural network, the sum of parameters n_p are calculated by (8) for one hidden layer and (9) for two hidden layers.

$$n_\mathrm{p} = 6n_\mathrm{h}(1) + 2(n_\mathrm{h}(1) + 1) \tag{8}$$

$$n_\mathrm{p} = 6n_\mathrm{h}(1) + n_\mathrm{h}(1)(n_\mathrm{h}(1) + 1) + 2(n_\mathrm{h}(1) + 1) \tag{9}$$

Experimental protocols

The objective function of learning algorithms is defined as MSE measure. To compare the results of experiments, the root mean square error (RMSE) measure would be used. The RMSE is the distance, on average, of a data point from the fitted line, measured along a vertical line. The RMSE is directly interpretable in terms of measurement units, and so is a better measure of goodness of fit than a correlation coefficient such as MSE.

NN Design Evaluation Strategy

The aim of this section is to present a method to find the best NN model so that the desired model generalized well and learned effectively among the experiments. Again, each experiment includes ten trials. Multiple trials mitigate the probability of obtaining a poor initial distribution of random weights to begin the iterative optimization process. Briefly, at each trial, the NN model trained by the training data set and evaluated by validation data set. The comparative criterion between experiments is considered based on the validation data RMSE value. Finally, to find the best NN model, the evaluation process is fulfilled by applying the test dataset on the desired NN design.

The utilized hypothesis testing method to make decision for finding the best NN design is Welch's t-test [15] which each experiment is considered as an independent set of the samples, i.e. trials. In evaluation process of two NN designs, the first NN design is defined as null hypothesis, whereas, the second NN design is considered as the alternative hypothesis. Due to the statistical hypothesis testing result, the null hypothesis may be accepted or rejected. If the null hypothesis is rejected means the related NN design has the lower mean validation-data RMSE value would be selected. This process would be performed pairwise on the all NN designs.

After the desired NN design is distinguished, the best model is selected based on the minimum RMSE due to applying test dataset among the ten trials.

RESEARCHERID
THOMSON REUTERS

ISSN: 2277-9655
[Ghasemimoghadam* *et al.*, 7(3): March, 2018] Impact Factor: 5.164
IC™ Value: 3.00 CODEN: IJESS7

Results

1) Learning phase and model selection

As mentioned before, in this work, the LM and SCG algorithms are employed in the computational experiments. The Fig. 5 ~ 8 show the results of the average validation-data RMSE values for the four aforementioned experiment's sets. The regulation ratio γ is changed in 0.1, 0.5, 0.9 and 0.95 values. Due to the increase of the regulation ratio, the error rate is significantly decreased. As can be seen in Fig. 5, by increasing the nodes number, the mean validation RMSE has been decreased. The RMSE index of the NN designs by the SCG algorithm is not remarkably changed (Figs. 6 and 8). Regarding to the results of employed LM algorithm, increasing the nodes in second layer gradually leads to reduce the mean RMSE and variance values (Fig. 7). In addition, the learning process by using SCG algorithm is terminated due to the "validation stop" criterion that shows the validation failures are reached to maximum value. The stop criterion for NN models by the LM algorithm is almost "maximum μ reached".

The NN designs were evaluated by the independent two-sample t-test method. Since the variance of the experiments are not equal, the Welch's t-test as a special kind of t-test was employed. The significance level (α) is set to 0.05. Table 3 shows the results of this evaluation for each experiment's set. Again, the evaluation process was performed between the mentioned four experiment's sets. The mentioned hypothesis testing method revealed that the ESL_LM_3L set by 50 nodes in first and 30 nodes in second hidden layer was the best NN design among other experiments that shows the dominant early stopping performance than the "maximum μ reached" criterion. Consequently, the high regulation ratio with the low nodes number cause to decrease the complexity of the neural network and, hence, achieve to obtain the appropriate NN design for the studied HEV.

The best NN model is obtained by the test-data RMSE value on 10 trials which NN model has the lowest test-data RMSE value is preferable. In this case study, the best trial has the 2.27 test-data RMSE value. Fig. 9 illustrates the linear regression curves of the obtained model with the test-data output and related target responses.

According to the phase 3 in Fig. 1, the NN model was employed in the studied HEV-system as the EM controller. The output of the trained EM controller (LCS) was evaluated with the non-causal near-optimal control strategies (NOCS) generated by the proposed optimization algorithms which applied as the data points in the learning process. Table 4 shows the overall fuel consumption of the LCS in comparison with the NOCS at initial SoC 60%. Figs. 10 and 11 compare the engine and EM2 torques of the NOCS and the LCS under WLTP (class 3) cycle at initial SoC 60%, respectively.

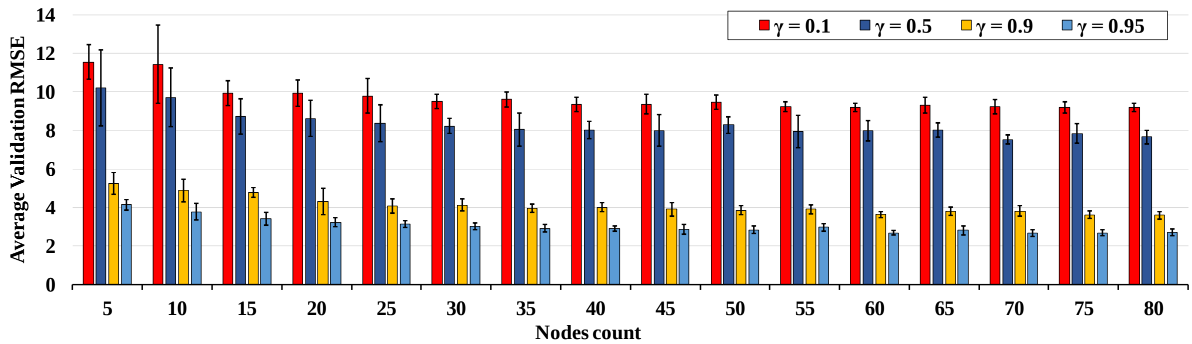

Fig. 5. The error rates on ESL_LM_2L experiment's set

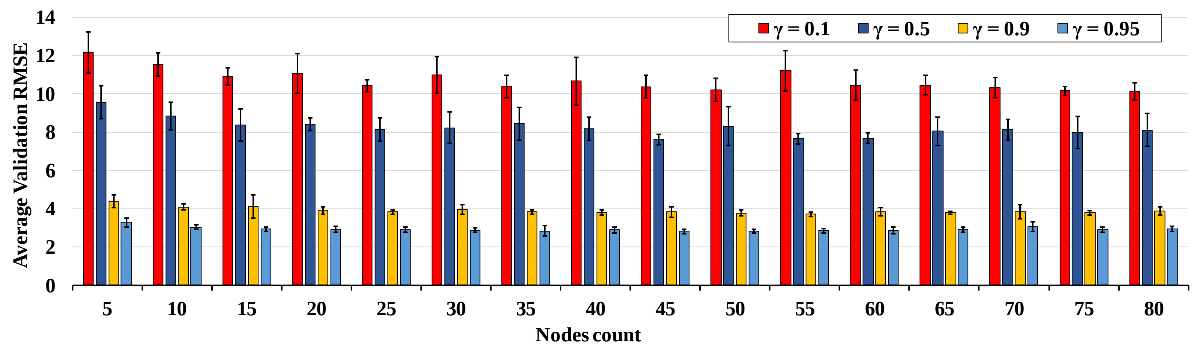

Fig. 6. The error rates on ESL_SCG_2L experiment's set

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

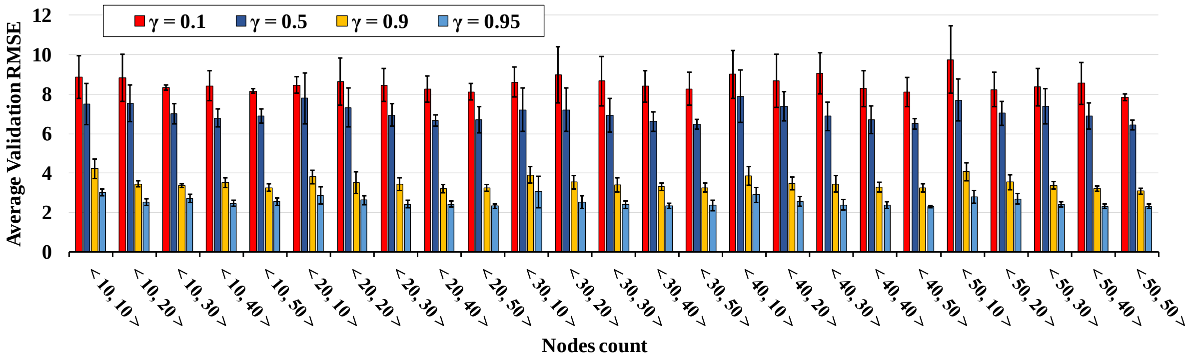

Fig. 7. The error rates on ESL_LM_3L experiment's set

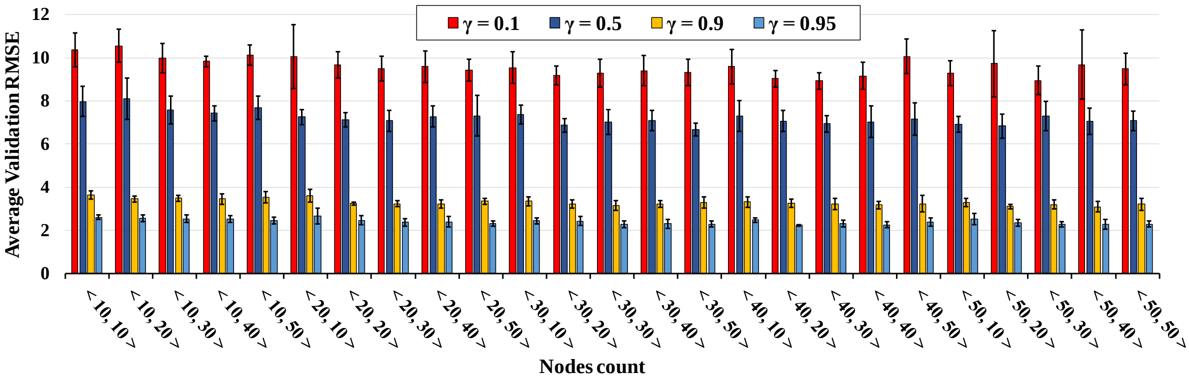

Fig. 8. The error rates on ESL_SCG_3L experiment's set

Table 3. The results of the best NN design

Experiment's Set Name	Best NN design		
	No. Nodes	Mean Validation RMSE	Standard Deviation Validation RMSE
ESL_LM_2L	60	2.6731	0.1087
ESL_SCG_2L	25	2.8834	0.1389
ESL_LM_3L	< 50,30 >	2.4029	0.1335
ESL_SCG_3L	< 40,50 >	2.3915	0.1912

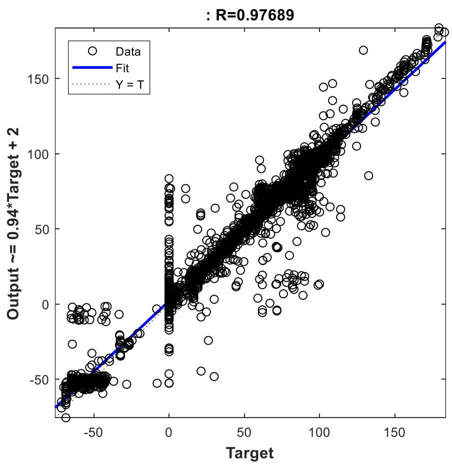

Fig. 9. The linear regression curve of the desired NN model

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Table 4. Increment fuel consumption comparison at initial SoC 60%

Controller	Drive Cycle			
	WLTP_C3	JC08	FTP	NEDC
Non-causal near-optimal	1029	298.219	715.033	506.07
trained NN model	1033	305.06	720.079	515.33

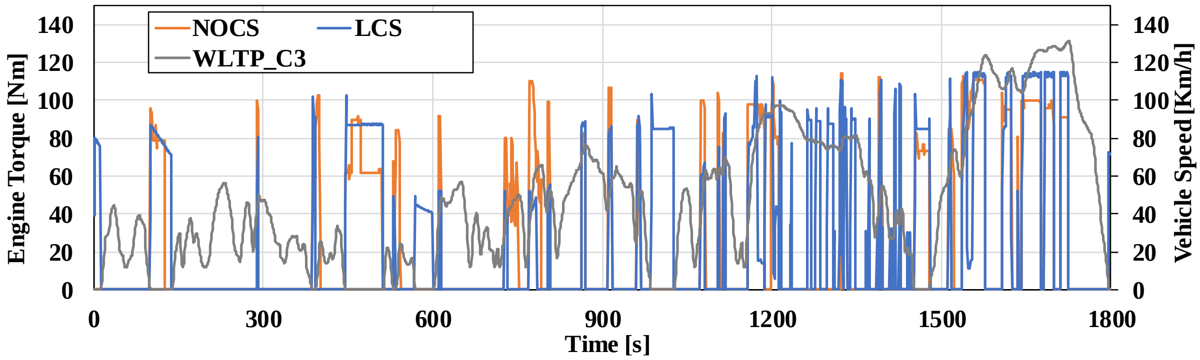

Fig. 10. Training result of NN model for the first decision variable, i.e. T_e

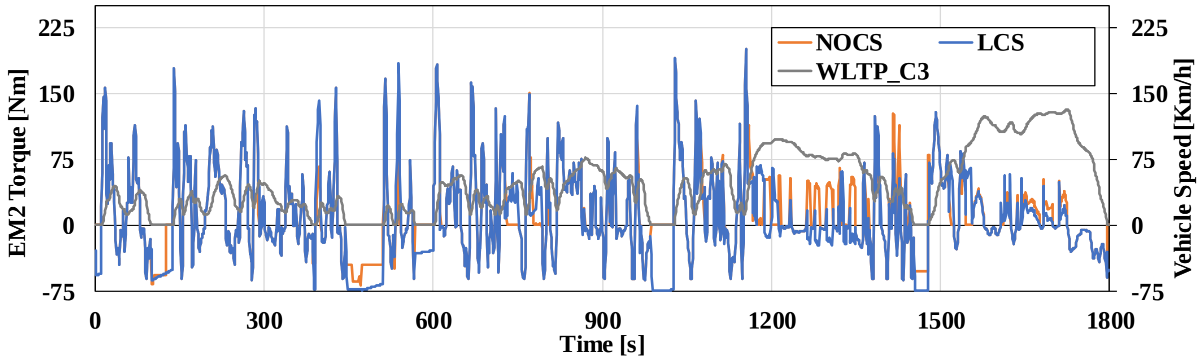

Fig. 11. Training result of NN model for the second decision variable, i.e. T_{em2}

2) Model verification

In order to verify the obtained EM controller, the two standard driving cycles, i.e. FTP-72 and Artemis (urban), are simulated. The performance of the learning-based EM controller is compared with a canonical controller. Therefore, the proposed EM controller and the deterministic rule-based control strategy (PFCS) were applied on the SPHEV system. The overall fuel consumption under aforementioned driving cycles by the two control strategies is shown in the Table 5. The SoC and the engine velocity trajectories based on the two control strategies over the Artemis (urban) drive cycle are illustrated in Figs. 12 and 13.

The SoC pattern changes serve insight to how the controllers manage the battery power and also the engine velocity trajectory reveals the engine operation status. The SoC pattern generated by the LCS was closer to the PFCS's SoC pattern from 0 through 240 seconds. During 240 seconds and 830 seconds, the LCS controller was charging the battery with less power than the PFCS controller. After 830, the LCS's and PFCS's SoC patterns until 920s close together. In other words, the LCS's SoC closely followed the trend of PFCS's SoC pattern. From 920s through the end, the PFCS controller charge the battery with higher power than the LCS controller. The ending SoC generated by both controllers was close together and same as the starting point, i.e. 60%.

Table 5. Increment fuel consumption comparison

CS \ DC	Artemis (Urban)	FTP-72
PFCS	262.077	468.798
LCS	258.706	460.001

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

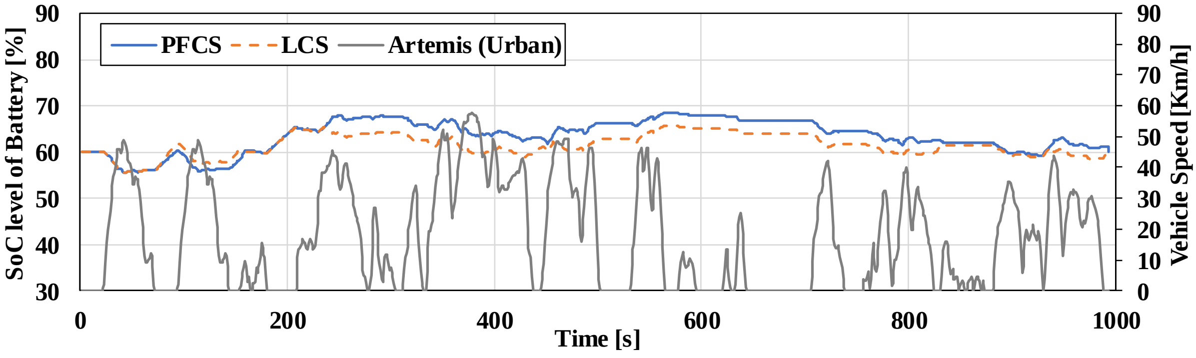

Fig. 12. The battery SoC under Artemis (Urban) cycle

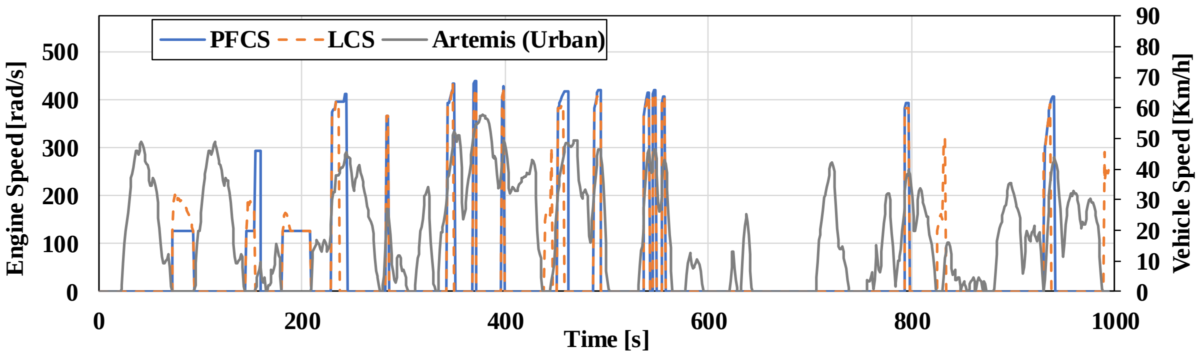

Fig. 13. The engine torque under Artemis (Urban) cycle

VI. CONCLUSION

In this paper, a learning-based framework for designing an efficient EM controller with high practicability in the HEVs has been presented. This framework utilizes the DNN technology for learning non-causal near-optimal energy settings based on the offline optimization processes. The utilized optimization techniques results cannot be used for in-vehicle control since it requires the full driving characteristics and computational demand. Therefore, the non-causal near-optimal solutions are saved in data store where are used as the learning data for training, validating and testing the learner. The learning work has been done to evaluate the effectiveness of the optimal paths generated by the utilized optimized techniques at various starting and fixed ending SoC under the several standard drive cycles.

This research involves the performance verification of the learned EM controller by designing several computational experiments which evaluated with the PFCS and NOCS. The first challenge is selection the suitable NN model which it generalized well with lowest complexity in NN parameters. For this sake, we suggested a method to realize the DNN technology and applying the hypothesis testing method for various NN designs. Then, the controller has been implemented inside the SPHEV model in Matlab platform for performance evaluations which consist of the SoC level, engine velocity and the EM2 and engine torques during the four driving cycles and increment fuel consumption over each driving cycle. The output from the learning-based EM controller are the sub-optimal engine and EM2 torques, which are then checked to make sure the HEV variables are within the appropriate constraints. The designed EM controller is verified with the PFCS controller under two standard driving cycles. Output of computational experiments on the SPHEV confirmed that the suggested methodology is suitable for designing an efficient EM controller with high practicability.

VII. REFERENCES

[1] X. Zhang, C. Mi, "Vehicle power management," Springer-Verlag London Limited, pp. 10-50, 2011.
[2] E. Ghasemimoghadam, K. Togai, H. Tamaki, "A methodology to design an efficient EM controller with high practicability in HEVs - Modeling and Optimization," Int. J. Eng. Appl. Sci. (IJEAS), Vol. 5, pp. 36-47, 2018.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

[3] M. H. Beale, M. T. Hagan, H. B. Demuth, "Neural network toolbox user's guide," The MathWorks Inc., pp. 156-173, 2017.
[4] J. Patterson, A. Gibson, "Deep learning a practitioner's approach," Octall Publishing Inc., pp. 41-80, 2017.
[5] M.T. Hagan, H. B. Demuth, M. H. Beale, O. De Jesús, "Neural network design", pp. 413-435, 2014.
[6] B.D. Ripley, "Statistical aspects of neural networks. In networks and chaos-statistical and probabilistic aspects", Chapman & Hall London, pp. 40-123, 1993.
[7] I. Kanellopoulos, G. Wilkinson, "Strategies and best practice for neural network image classification", Int. J. Remote Sensing, Vol. 18, pp. 711-725, 1997.
[8] X. YAO, "Evolutionary artificial neural networks", Int. J. Neural Systems, Vol. 4, pp. 203-222, 1993.
[9] Y. Le Cun, J.S. Denker, S.A. Solla, "Optimal brain damage, in advances in neural information processing systems 2", pp. 598-605, 1990.
[10] D. Stathakis "How many hidden layers and nodes?", Int. J. Remote Sensing, Vol. 30, pp. 2133-2147, 2009.
[11] D. Nguyen, B. Widrow, "Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights", in Proc. Int. Joint Conf. Neural Netw., San Diego, vol. 3, pp. 21-26, 1990.
[12] A. Pavelka, A. Procházka, "Algorithms for initialization of neural network weights", Sborník Príspevku 12 Rocníku Konf. MATLAB 2004, Prague, vol. 2, pp. 453-459, 2004.
[13] M.F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning", Neural Netw., vol. 6, pp. 525-533, 1993.
[14] M. Matsumoto, and T. Nishimura, "Mersenne twister: a 623-dimensionally equidistributed uniform pseudorandom number generator," ACM Trans. Modeling Comput. Simulat., vol. 8, no. 1, pp. 3-30, 1998.
[15] B.L. Welch, "The generalization of 'Student's' problem when several different population variances are involved", Oxford University Press, vol. 34, pp. 28-35, 1947.

CITE AN ARTICLE

Ghasemimoghadam, E., Togai, K., & Tamaki, H. (n.d.). A METHODOLOGY TO DESIGN AN EFFICIENT EM CONTROLLER WITH HIGH PRACTICABILITY IN HEVS LEARNING. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY*, *7*(3), 780-791.