IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 18, 2019, accepted December 11, 2019, date of publication December 19, 2019, date of current version January 7, 2020.

*Digital Object Identifier 10.1109/ACCESS.2019.2960771*

# A Convolutional Neural Network-Based Driving Cycle Prediction Method for Plug-in Hybrid Electric Vehicles With Bus Route

**ZHENG CHEN[1], CHAO YANG[2], (Member, IEEE), AND SHENGNAN FANG[3]**
[1]School of Sciences, Ningbo University of Technology, Ningbo 315211, China
[2]School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China
[3]State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing 100084, China

Corresponding author: Shengnan Fang (fsn@tsinghua.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51975048 and Grant 51805290, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LY18E050015, and in part by the Ningbo Natural Science Foundation under Grant 2018A610124.

**ABSTRACT** Driving cycle prediction plays a key role in energy management strategy (EMS) for hybrid electric vehicles (HEVs). This paper studies a driving cycle prediction method based on convolutional neural network (CNN). Firstly, the k-shape clustering method is used to group the driving cycle data into six different types. Moreover, this method is compared with the k-means algorithm which is often used for clustering driving cycles. Secondly, CNN is adopted to predict the different types of the driving cycles based on the results of k-Shape clustering. Some basic features are selected to construct the input of the networks with no assistance of human experience. In the process of training neural networks, some high-level features which can describe the information of a driving cycle more accurately are extracted, and the deep neural networks are built, which are different from traditional experience-based driving cycle prediction methods. And then, the better performance of the proposed method is illustrated by making a comparison with the traditional machine learning method. Finally, an adaptive energy management strategy for plug-in hybrid electric buses (PHEB) based on deep learning is given, and simulation results prove the effectiveness of the proposed method.

**INDEX TERMS** Plug-in hybrid electric bus, driving cycle prediction, energy management strategy, deep learning.

## I. INTRODUCTION

Nowadays, plug-in hybrid electric buses (PHEB) are widely used in public traffic field thanks to the ease of being charged by the power grid and its better performance in fuel consumption [1], [2]. There are different operating modes for a PHEB, such as electric vehicle (EV) mode, charging-sustaining (CS) mode, regenerative braking mode, and so on, which is helpful to achieve better vehicle fuel economy [3]. Pattern of the energy flow in these operating modes depends on the energy management strategy (EMS) for PHEB [4]. However, EMS is mainly determined by driving cycle types. Therefore, it is very important to recognize the types of driving cycles online. In general, there are two steps to complete this task. First of all, for huge volumes of driving cycle data, various clustering algorithms are used to put data into homogeneous groups, which can obtain the training set of the driving cycle data [5]. It is very difficult to achieve this purpose since any prior knowledge of the groups is unknown. The most popular clustering algorithm for driving cycle data is k-means [6]. However, this algorithm has to construct the features from the driving cycle data, and then cluster these features for corresponding driving cycles. All features of driving cycle data are selected based on the knowledge of the vehicles or the experts' intuition [7], [8], therefore it is difficult to determine these features. Moreover, the features will not reflect the shape of driving cycle time series. For k-Shape clustering method, it is not necessary to construct the features of driving cycle. More importantly, k-Shape clustering algorithm can preserve the shapes of driving cycle time series [9], [10].

The associate editor coordinating the review of this manuscript and approving it for publication was Bin Zhou.

This is the motivation for selecting k-Shape method in this paper. After the clustering process, different types of driving cycles are obtained. In the next step, the type of the driving cycle has to be predicted via statistical method.

In practice, it is hard to predict the driving cycle type since too many random events can affect them [11]. Furthermore, when the number of driving cycles is large, classification becomes a difficult job [12], [13]. A natural idea to predict the driving cycle type is based on the result of clustering method about historical driving cycle data [14], [15]. However, it is impossible to predict the driving cycle type based on the raw driving cycle data. Therefore, some feature parameters also have to be selected from driving cycle data to classify the types of driving cycles [16], [13]. In [17], independent measures to describe the dimensions of urban driving patterns are given, and then the properties which mainly affect emissions and fuel consumption are studied. 62 driving cycle parameters are selected to represent different driving patterns which are collected in real traffic. In [18], average speed and stop time are used to classify the driving cycles. From the researches mentioned above, it can be seen that using less parameters will reduce the computational cost. However, too few parameters cannot give much information of the driving cycles. Therefore, choosing some parameters from the parameter set to give good results of recognizing driving cycles is a very challenging task.

However, the selection of these features depends heavily on the domain knowledge and engineers' experiences [5], [19]. Therefore, it is hard to find out an optimal driving cycle representation. Therefore, it is necessary to find a new method to overcome these drawbacks [20], [21]. In convolutional neural networks (CNN), raw data can be considered as input data without further selection and construction. In this paper, convolutional neural networks, which are the most popular deep neural networks, are used to recognize the types of driving cycles.

There are three contributions in this paper. First, the k-Shape clustering algorithm is used to group the driving cycles into six different driving cycle types. This clustering method is suitable for time series clustering, so it is not necessary to select the features of driving cycles. Second, a method is designed to map the raw driving cycle data onto some feature matrices, which are used as the input data of CNN. The third contribution is the proposed deep neural networks which can learn more complex feature from the low-level feature matrices. Compared with current machine learning method, the proposed deep learning method provides better classification accuracy and requires less human intervention. To the best of our knowledge, it is the first time for a research to use deep learning method to classify the different driving cycle types. Simulation on a large number of real datasets in this paper shows that deep learning method can be a powerful tool for recognizing driving cycle types.

The remainder of this paper is organized as follows. Section II details the k-Shape clustering algorithm and the deep convolutional neural networks used in classifying the driving cycles. In section III, numerical experimental results are given based on a large amount of real data to show the better performance of deep learning compared with the traditional machine learning method. Energy management strategy for PHEB based on the classification results is given in Section IV. Some results and analysis are obtained via simulation to illustrate the effectiveness of the EMS in Section V. The conclusions are summarized in Section VI.

## II. K-SHAPE CLUSTERING ALGORITHM AND CONVOLUTIONAL NEURAL NETWORKS

In this section, k-Shape clustering algorithm and convolutional neural networks related to our research will be introduced. k-Shape clustering algorithm is used to obtain the different driving data types in our study, which are used as training data in the classification model [22], [ 25]. The CNN is used to predict the driving cycles of a bus. There are two main purposes for the proposed classification method with CNN. First, the input data of neural networks are constructed. Second, more complex features are learned by deep networks. Here the raw data only consider the driving cycle data, which are the main factor in constructing the EMS. In the following parts, the proposed methods will be discussed in detail.

### A. K-SHAPE CLUSTERING ALGORITHM FOR DRIVING CYCLES

The driving cycle segments with different window sizes are time series about speed. So k-Shape clustering algorithm lends itself readily to driving cycle data clustering. The definition of clustering of driving cycles in this paper is given as follows: given a dataset of n driving cycles $D = \{V_1, V_2, \cdots, V_n\}$, the process of unsupervised partitioning of $D$ into $C = \{U_1, U_2, \cdots, U_k\}$, in such a way that homogenous driving cycles are grouped together based on a certain measure, is called driving cycles data clustering. Here, $D = \bigcup_{i=1}^{k} U_i$ and $U_i \bigcap U_j = \phi, i \neq j$.

K-Shape clustering algorithm is used to cluster the different driving cycles into the same type of driving cycles, which is a novel algorithm that can preserve the shapes of driving cycle data. As a clustering algorithm, shape-based distance should be given firstly as the following equation [24].

$$SBD(x, y) = 1 - \max_{\omega}\left(\frac{CC_{\omega}(x, y)}{\sqrt{R_0(x, x)R_0(y, y)}}\right) \tag{1}$$

where $CC_{\omega}(x, y) = R_{\omega-m}(x, y), \omega = 1, 2, \cdots, 2m-1$. The above equation is computed by the following equation

$$R_k(x, y) = \begin{cases} \sum_{l=1}^{m-k} x_{l+k} y_l, & k \geq 0 \\ R_{-k}(x, y), & k < 0 \end{cases} \tag{2}$$

It is noteworthy that the values of SBD is between 0 to 2, and 0 indicates perfect similarity for driving cycle types. As for the computation of SBD, it can be computed via the Inverse Discrete Fourier Transform of the product of the individual Discrete Fourier Transforms of the time series [25].

In the following parts, the centroids for time series clustering based on SBD distance measure are given as the following equation:

$$\mu_k^* = \arg\max_{\mu_k} \sum_{x_i} \left( \frac{\max_\omega CC_\omega(x_i, \mu_k)}{\sqrt{R_0(x_i, x_i) R_0(\mu_k, \mu_k)}} \right)^2 \tag{3}$$

The k-Shape clustering method depends on the SBD distance measure and the centroids of time series. The whole algorithm is given as follows. The input is an n-by-m matrix containing n time series of length m that are initially z-normalized. The output is a k-by-m matrix containing k centroids of length m [26].

**Algorithm 1** k-Shape Clustering Algorithm

Initialize iter $\leftarrow$ 0 and IDX$'$ $\leftarrow$ [ ]
While IDX ! = IDX$'$ and iter < 100 do
  IDX$'$ $\leftarrow$ IDX.
  for j = 1 to k do
    X$'$ $\leftarrow$ [ ].
    for i =1 to n do
      if IDX(i) = j, then X$'$ $\leftarrow$[X$'$, X(i)]
      $G(j) \leftarrow ShapeExtraction(X', G(j))$
  for $i \leftarrow 1$ to n do
    $\min dist \leftarrow \infty$.
    for $j \leftarrow 1$ to k do
      $[dist, x'] \leftarrow SBD(C(j), X(i))$.
      if $dist < \min dist$ then $\min dist \leftarrow dist$
        $IDX(i) \leftarrow j$
Assign iter + 1 to iter.

In the above algorithm, $X$ and k are the input driving cycle data. $X$ and $C$ are the output. There are two steps in every iteration of the algorithm. One is assignment step, which updates the cluster member by comparing each driving cycle segment with all computed centroids and by assigning each driving cycle segment to the cluster of the closest centroids. The other is the refinement step, where the cluster centroids are updated to reflect the changes in cluster membership in the previous step. The terminal condition of this algorithm is either no change in cluster membership or the given number of iterations is reached. In addition, the computational complexity of the k-Shape clustering algorithm depends on the number of given driving cycle data.

In order to determine how many types of driving cycles should be clustered, cluster validation is introduced. As for cluster validation, unfortunately, there is no best Cluster Validity Index (CVI) for the k-Shape clustering algorithm. In this study, a proper CVI is used to compare the clustering result of k-Shape clustering algorithm with k-means clustering algorithm [27].

### B. CONVOLUTIONAL NEURAL NETWORKS FOR PREDICTION

Deep neural networks have better performance in many fields such as computer vision and human behavior [28], [29]. Driving cycle data are basically time series, and this is the main reason why deep learning method is used for the learning of driving cycle feature representation. For convolutional neural networks model, there are two new types of layers which are different from traditional neural networks. One is convolutional layers. The other is pooling layers. There are four hyperparameters in convolutional neural networks which have to be determined: number of filters $K$, their spatial extent $F$, the stride $S$ and the amount of zero padding $P$. In pooling layers, pooling function, the size of filter and the stride should be given in advance.

However, driving cycle data are simply a sequence of two dimensional coordinates $(t, v)$. Experiences show that if these driving cycle data are simply treated as input data of both traditional machine learning method and deep learning algorithms, the results are usually unsatisfactory [30]. Therefore, the raw driving cycle data should be transferred into some matrix-type data which are viewed as inputs of deep neural networks.

In order to gather information from driving cycle data more accurately and easily, the whole driving cycle will be divided into several short segments considering the bus stop. And the length of each segment is denoted as $L$. Furthermore, five features are obtained from driving cycle data: (1) maximum velocity, (2) average velocity, (3) minimum acceleration, (4) maximum acceleration, and (5) maximum deceleration. These features are also used by traditional machine learning method such as support vector machine (SVM). In order to capture the short period information and long period information of driving cycles, each driving cycle segment is divided into shorter period $L'$. Here $L' = 4s$ considering the normal speed of the vehicle. For the accuracy of the statistical method, the shorter period $L'$ overlaps between the adjacent driving cycle segments with a shift of $\frac{L'}{2}$. Then the $5 \times \frac{2L}{L'}$ feature matrix is the input of the deep neural network. In this matrix, 5 rows represent the five features above axis, and $\frac{2L}{L'}$ columns are for time axis. In addition, only five simple features are selected in this paper since we hope more complex features can be learned by deep neural networks automatically.

### C. BASELINE MACHINE LEARNING METHOD

For comparisons with the proposed clustering method and deep learning methods, baseline methods should be given. In this paper, k-means clustering method is compared with k-Shape method, and SVM and $k$ nearest neighbor (kNN) method are adopted for the baseline of classification problem since they have been viewed as powerful machine learning methods. In this paper, we train SVM and kNN on a set of 5 driving cycle features which are given above.

## III. NUMERICAL EXPERIMENTS FOR PROPOSED APPROACH

In previous researches, k-means method is a frequently used method to cluster the driving cycle data. So the k-Shape clustering method is compared with k-means method.

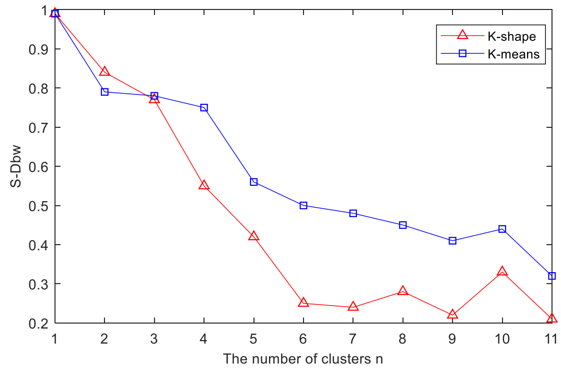


**FIGURE 1.** S-Dbw of k-means and k-Shape method.

The research objects for these two methods are driving cycle segment data whose length is 128s as is mentioned in Section II. The feature parameters in k-means method are as follows: maximum velocity, average velocity, minimum velocity, maximum acceleration, and maximum deceleration for every driving cycle segment. As for k-Shape clustering method, there is no need to select any features from the sequences of velocity of the vehicle. The S_Dbw is adopted as CVI in order to explain the performance of the clustering method, and the expression of S_Dbw can be found in [31], which is a proper CVI used in different clustering methods. The index S_Dbw can be normalized into the range [0, 1]. The smaller value of S_Dbw, the better performance the clustering method gives. Fig. 1 gives the different values of S_Dbw under different numbers about k-means and k-Shape method respectively.

In the above figure, it is obvious that the cluster performance of k-Shape method is better than the results of k-means method under different number of subgroups. In general, the overall trend of S_Dbw for k-means and k-Shape is down before the number of subgroups is 6. And then the value of S_Dbw changes around a fixed value. In addition, the best choice is to divide all the driving cycle segment data into six groups considering both the value of S_Dbw and the computational burden. Therefore, six different driving cycle segment types are considered in this paper. Because the driving cycle segments of every type are selected at random, the difference is not so obvious judged by people. Therefore, the distributions of average velocity and average acceleration for these six driving cycle types are given in Fig. 2, which show the differences among the six types.

From the results of clustering, the six types of driving cycle segments can be described as six road conditions in Table 1 considering the road and traffic information.

In what follows, SVM and kNN method are also used to recognize the different types of the driving cycles, which are trained based on training data sets. These two methods are widely used in previous research. Therefore, results of these two methods are compared with CNN method, which is shown in the following part. In order to assess the performance of our prediction models, K-fold cross-validation is used. For kth part, the other k-1 parts are used to train the model, and calculate the prediction accuracy of the model when predicting the kth part. And the prediction error is given as follows.

$$CV(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N} d(y_i, \hat{f}^{-g(i)}(x_i)) \tag{4}$$

where $d(\cdot, \cdot)$ is distance. As for kNN method, k is an important parameter which determines the results of the classification. The following fig. 3 shows the accuracy of the classification with different k.

From the above figure, it can be seen that the accuracy of kNN model is below 87%. As for SVM method, the result is given in Fig. 4, which is below 86%.

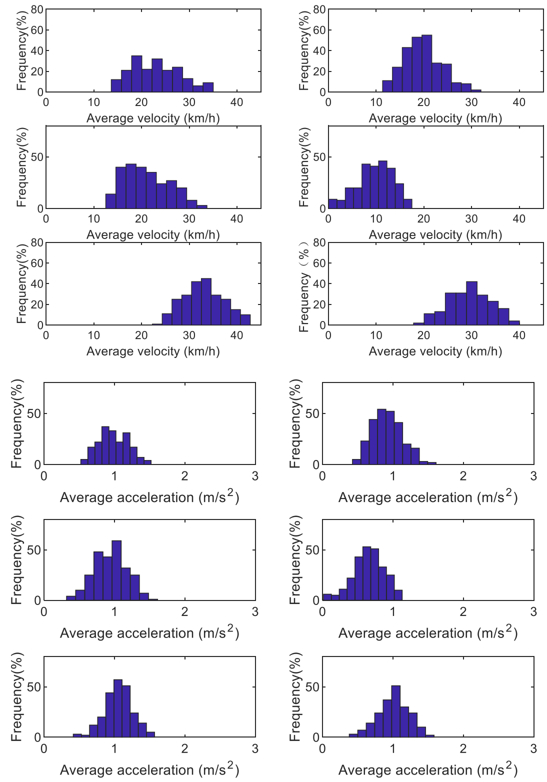


**FIGURE 2.** The distribution of different types of segments.

**TABLE 1.** Description for six different types of driving cycle segments.

| Road type | Traffic condition |
|---|---|
| Flat road | High velocity; Smooth traffic |
| | Middle velocity; Little traffic jam |
| | Low velocity; Traffic jam |
| Distance-varying ramp | Middle velocity; Smooth traffic |
| | Middle velocity; Traffic jam |
| | Low velocity; Traffic jam |

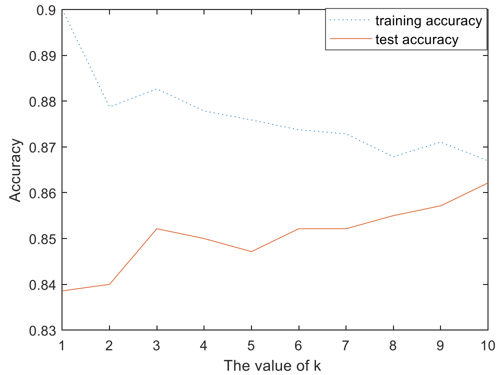


**FIGURE 3.** Accuracy value under different k.

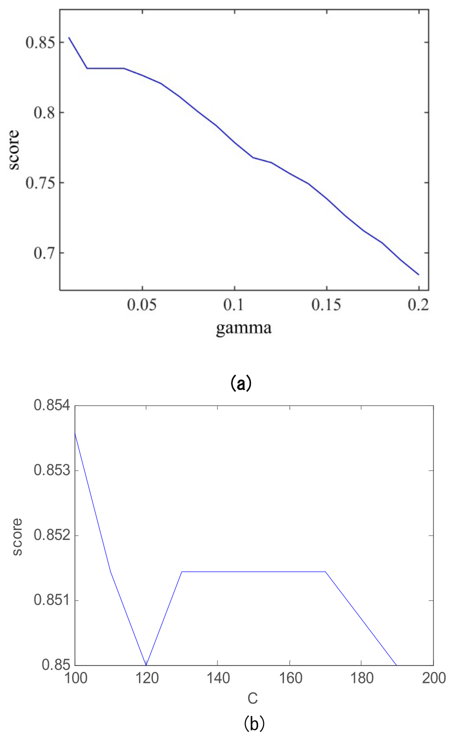


**FIGURE 4.** The results of SVM for different gamma and C.

There are two parameters which will affect the results of SVM method. One is kernel bandwidth: gamma, and the other is regularization parameter: C. In the above two pictures, they give the different values of accuracy according to different values of parameters. It is worth noting that the result of Fig 4. (a) is given when the value of C is optimal, and the result of Fig. 4 (b) is given when the value of gamma is optimal.

In k-Shape clustering algorithm, each driving cycle data are divided into six segments with the same window size, which is easier to deal with in the process of clustering and classification. Then a large number of driving cycle data segments are obtained, which is used as training data of the algorithm. It is expected that the driving cycles are clustered into six different types.

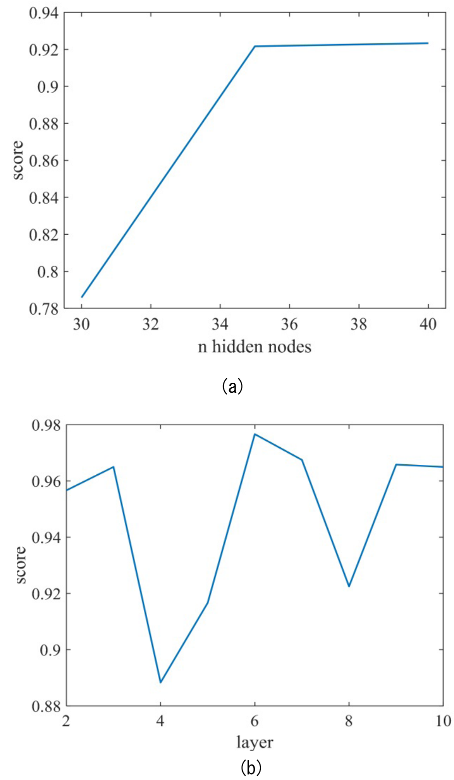


**FIGURE 5.** The accuracy under different numbers of neurons and layers.

In order to obtain better performance of the classification, the driving cycle data sampling rate cannot be set below a certain value since a low sampling rate may result in much effective information. In our experiment, the sampling rate is 1Hz. The dataset is constructed from 300 driving cycles. Here, $L = 128s$, $L' = 4s$. In all the tests, 80% examples are selected as train data, and 20% examples are test data.

The CNN architecture for driving cycles is built with Keras in Python. The network has 6 layers. The first layer is input layer, and then two convolutional-pooling layers are connected. The remaining three are fully-connected layers. The last layer is the Softmax function, which outputs the type of driving cycles. All the above hyperparameters is selected by cross validation. For CNNs, stochastic gradient descent optimizer is used. Its learning rate is 0.08, and decay is 1e-6. When constructing the CNN, two aspects should be considered. One is the number of the neurons in each layer. The other is the number of the layers. In the following Fig.5, the tendency of the accuracy for driving cycle prediction is given as these two numbers become larger.

From the above two pictures, it can be concluded that increasing the number of layers provides better accuracy of prediction than increasing the number of neurons in each layer. In this paper, the accuracies of the driving cycle prediction are above 95% by using CNN method.

It is interesting to investigate what kinds of features are learned by the deep networks. In each hidden layer, the output of this layer can be considered as new features of driving cycles, which is learned by deep networks. These new features are more complicated than the raw feature, but the meanings of these new features are very hard to explain.

From the above analysis, the performance of CNN is better than other classification methods. So it is therefore a potentially desirable method of classification to improve the accuracy of driving cycle prediction.

In application scenarios which require real-time prediction, the proposed deep learning approach has a significant advantage over the traditional methods such as SVM, which relies on some complex handcrafted features. Because only lower level data are needed, given a pre-trained network, the deep learning approach can be used in real-time prediction where lower level data are available online as the bus moves. In contrast, some complex features which are used in traditional prediction method can only be available after the whole trip ends. This restricts traditional methods to be used for online prediction purpose, whereas the deep learning approaches are far more flexible. It is possible to build an online system based on the proposed deep learning approach, e.g. to predict the driving cycle identity based on the data collected at lower level during runtime.

## IV. ENERGY MANAGEMENT STRATEGY FOR PHEB

In this section, the energy management strategy of PHEB will be given through the proposed deep learning method. Since the energy management strategy of plug-in hybrid electric bus based on traditional machine learning method is given in [2], the deep learning based method is our focus in this paper. The powertrain structure of the PHEB considered in this study is single-shaft parallel hybrid configuration which is given in Fig.6. PHEBs with this configuration are widely used in public transport in China due to their simple structures and good controllability [16]. By some proper control strategy, the PHEB will work under different modes to achieve optimal performance of the vehicle. The operating point set of the electric motor (EM) and engine can work in its own best efficiency areas by the adjustment of automated manual transmission (AMT) [13]. The clutch is always used to change the work modes of the powertrain mentioned above. The EM is able to work as a motor or a generator according to different work modes. And the electric quantity of Lithium titanate battery can be charged very quickly. The basic parameters of the PHEB are listed in Table 2.

In order to minimize the vehicle energy consumption, a model should be estimated via Matlab Simulink. In general, two basic approaches are used to construct the mechatronic systems of the PHEV. One is the theoretical model, which contains the functional description between the physical data and its parameters. However, these models are very hard to obtain. Therefore, the other approach is adopted, which is called quasi-static models. These models focus on giving a simple representation which can be easily expressed but remains physically interpretable as much as possible. The model of components of the vehicle is given as follows.

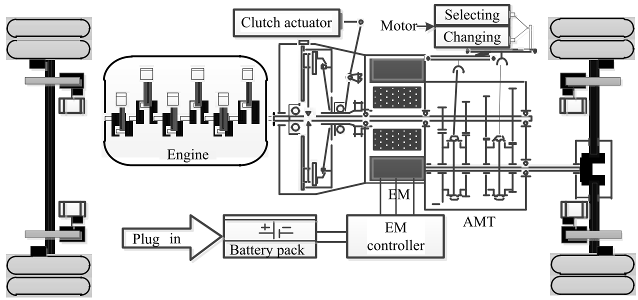


**FIGURE 6.** The configuration of the PHEB powertrain.

**TABLE 2.** Basic parameters of the PHEB.

| Components | parameters |
|---|---|
| Vehicle mass | 14,000kg |
| Engine | YC6G230N, CNG, 6.45L, normal power:170Kw |
| Motor/Generator | max torque:750Nm, normal/peak power: 94/121kW |
| Battery | capacity:60Ah |
| AMT | 6-speed, gear ratio: 6.39/3.97/2.4/1.48/1/0.73 |
| Final Drive | Ratio:5.571 |

The torque of wheel is given as following equation.

$$T_w = \eta_T \cdot i_{AMT} \cdot i_f (T_e + T_m) + T_b \tag{5}$$

The variables in equation (5) are explained as follows: $\eta_T$ is transmission efficiency. $i_{AMT}$ and $i_f$ represent gear ratio of the AMT and the differential gear ratio, respectively. $T_e$ and $T_m$ are the engine torque and the EM torque, respectively. $T_b$ is the braking torque acting on the wheel. Another expression of $T_w$ is given by the equation below.

$$T_w = [mgC_r \cos\theta + \frac{1}{2} C_D \rho_d A V_a^2 + mg \sin\theta + \delta m \frac{dV_a}{dt}] \cdot r \tag{6}$$

where $m$ is the vehicle mass. $g$ is the gravity acceleration, and $C_r$ is the rolling resistance coefficient which is given as follows.

$$C_r = C_1 + C_2 V_a \tag{7}$$

where $C_1$ and $C_2$ are different rolling resistance coefficients. $\theta$ is the road slope angle. $C_D$, $\rho_d$, and $A$ represent air drag coefficient, air density, and frontal area of the bus, respectively. $V_a$ is vehicle speed and $\delta$ is correction coefficient of rotating mass. r is the wheel radius.

In the PHEB, compressed natural gas (CNG) is used by engine, and the equation for CNG consumption rate per unit time $Q_g$ is given as follows.

$$Q_g = \frac{P_e b}{367.1 \rho_g g} \tag{8}$$

where $b$ is the compressed natural gas consumption rate which is a function of the current engine torque and rotational speed. is the engine power calculated through $P_e = T_e\omega_e$. $\rho_g$ is the density of CNG.

Because the EM in the PHEB can be used to drive the bus or recover the kinetic energy to the battery which is called regenerative brake, the EM power can be written as:

$$P_{EM} = \begin{cases} \dfrac{T_m\omega_m}{\eta_{EM}}, & \text{motor} \\ T_m\omega_m\eta_{EM}, & \text{generator} \end{cases} \tag{9}$$

where $\eta_{TM}$ is the EM efficiency, which is a function of the torque and the rotational speed of the EM.

If the thermal-temperature effects and transients are neglected, the physical model of the battery is considered as a static equivalent circuit. According to Kirchhoff's voltage law, the equivalent circuit equation is written as:

$$U(t) = U_{oc}(t) - R_{int}(t)I(t) \tag{10}$$

where $U_{oc}(t)$, $R_{int}(t)$, $U(t)$, and $I(t)$ are the open-circuit voltage, internal resistance, terminal voltage, and internal current of the battery, respectively. The battery SOC is calculated through the following equations.

$$SOC(t) = \frac{Q(t)}{Q_0} \tag{11}$$

$$\dot{Q}(t) = I(t) \tag{12}$$

$$I(t) = \frac{U_{oc}(t) - \sqrt{U_{oc}^2(t) - 4R_x(t)P_{EM}(t)}}{2R_x(t)} \tag{13}$$

where $Q(t)$ and $Q_0$ represent the quantity and the capacity of the battery, respectively. $R_x(t)$ is the resistance of the battery which is described as follows.

$$R_x(t) = \begin{cases} R_{dis}(t), & \text{discharging} \\ R_{chg}(t), & \text{charging} \end{cases} \tag{14}$$

where $R_{dis}(t)$ and $R_{chg}(t)$ are internal resistances when the battery is providing energy or is being charged, respectively.

Several families of energy management strategies have been researched in literature. In order to obtain the global optimal solution of the EMS problem, dynamic programming (DP) is used through the known driving cycle [32]. But this method is not available online since the information of driving cycle is not known in advance. Therefore, DP method is considered as a benchmark or a complementary method for other EMS. In real time application, rule-based method is widely used in HEV/PHEV, which is constructed by a set of rules without any optimization method [33]. However, the rules are based on intuition and remain unchanged. So under some driving cycle, it achieves a solution far from the global optimal solution. Another practical application EMS is Equivalent Consumption Minimization Strategy (ECMS) [34]. In this paper, our ECMS approach is based on ECMS.

It is noted that ECMS is an online control strategy for optimal energy management, which reduces a global optimization problem to an instantaneous minimization problem. ECMS can be expressed by the following formulation.

$$\min_{u(t)} \phi_{fuel}(u(t)) = \int_{t_0}^{t_f} \dot{m}_{fuel}(u(t)) + \frac{s}{H_l}P_{bat}(u(t))dt \tag{15}$$

where $\dot{m}_{fuel}(\cdot)$ is the fuel mass flow of the internal-combustion engine. $P_{bat}(\cdot)$ is the electrical battery power, and $H_l$ is the fuel's lower heating value. $s$ is the equivalence factor which depends on future driving cycles. In theory, by selecting proper value of $s$, the solution will reach the global optimal solution, which is solved by DP. Therefore, finding the proper value of $s$ is a hard task in practical applications since the value of $s$ is affected by various random events. In this paper, the value of $s$ is determined according to the six types of driving cycles. That is to say, an adaptive control is obtained.

As mentioned above, the bus route is classified into six different driving cycle types through k-Shape clustering algorithm. According to different types of driving cycles, different sub-strategies are given. That is to say, the EMS of buses should be given according to these six types. Next, the convolutional neural networks are built by the different driving cycle data and the types of driving cycle, which are regarded as training data of the networks. This convolutional neural network is used to predict the types of the new driving cycles.

In the following part, the energy management strategy for PHEB is proposed as follows. After the bus leaves the stop and reaches a steady speed, the device on buses obtains the driving cycle data as the test data during a time period of 128s. Then these raw driving cycle data will be transferred into input of the deep neural networks via the above method. The output of the deep neural networks is for the types of the driving cycles, which are considered as the prediction of future driving cycle type until the bus arrives at the next stop. Once the type of the driving cycle is given, the energy control strategy on this type is given. This process will be repeated until the bus arrives at the end stop. Here, EMS of the bus is presented according to the driving cycle type. That is to say, the EMS is switched among different sub-strategies based on various driving cycles. The flow chart of the EMS is shown as the following Fig. 7. Here, equivalent consumption minimization strategy is adopted as sub-strategies. This EMS of PHEB is called adaptive ECMS (AECMS). The whole process of the EMS is given as the following figure.

In general, there are two parts to complete the whole control process: the offline part and online part. In the offline part, historical driving cycle data are collected by buses. These time series about the velocity is clustered by k-Shape clustering method. Then, different types of driving cycles are obtained, which are used as input data of convolutional neural networks. Finally, the CNN is built by input data, and the different types of EMS are constructed according to different types of driving cycles. In the online part, the current driving cycle data are acquired by the running bus which sends the data to computing center by wireless communication. The computing center analyzes the data by CNN, and the new driving cycle type is predicted. Then the computing center sends this message to the bus, and the appropriate EMS is

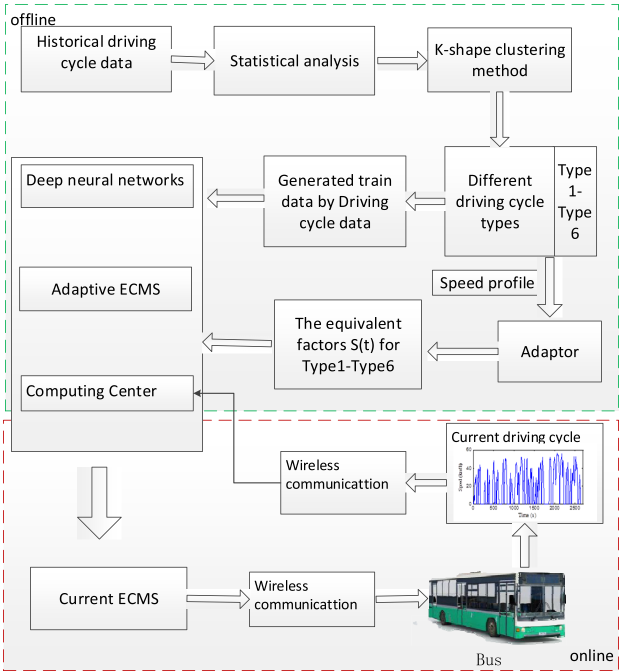


**FIGURE 7.** Flow chart of the proposed EMS.


**FIGURE 8.** A typical bus route map.

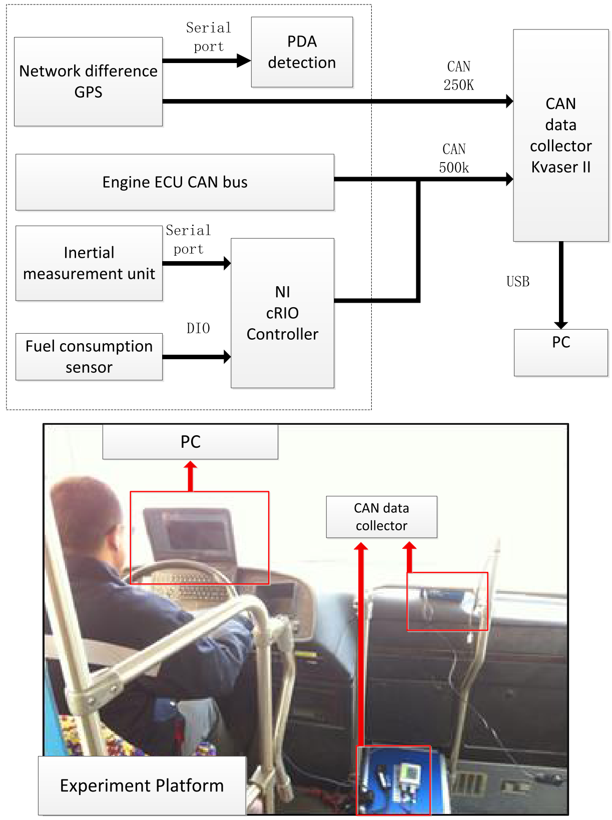


**FIGURE 9.** Road driving cycles collection system.

**TABLE 3.** The fuel economy of ECMS and AECMS.

| Parameter | ECMS | AECMS |
|---|---|---|
| Fuel consumption ($m^3$/100km) | 16.87 | 13.80 |
| Electric consumption(kWh/100km) | 44.56 | 42.79 |
| Cost (RMB) | 120.48 | 104.89 |
| Improvement (%) | -- | 14.86 |

adopted by the bus. After 128s later, the bus sends new driving cycle data to the computing center and the process is repeated until the bus reaches the end stop.

Another difficulty is that the analysis of the big data and the computation of the CNN can not be completed with on board unit as mentioned above. Fortunately, the recent penetration of the mobile wireless internet and cloud computing technology make our approach possible for their application in buses. Simply speaking, a two-way communication system between the buses and the computing center is established. The bus sends the driving cycle data to the computing center. The data are analyzed and the driving cycle is recognized by CNN. Then, the optimal ECMS is sent back to the bus.

## V. SIMULATIONAND ANALYSIS OF EMS

As for the simulation, some points must be explained first. Although the accuracy of CNN is high, the prediction accuracy for driving cycle type in our method may be not ideal. That is because the future driving cycle type prediction is based on the current driving cycle type result obtained by CNN. However, the future driving will be affected by many random conditions, so the predicted result may not be consistent with the real situation. Especially, when some emergent accidents occur. But from the statistical view, the performance of our prediction method is available on average. In following part, some results about the simulation are given.

The driving cycles for simulation in this paper are selected from the B13 bus route in ChangZhou city. This route starts from Jinghaixingcheng (Point A in Fig. 8) to railway station bus centre (Point B in Fig. 8), which includes 24 stops in total.

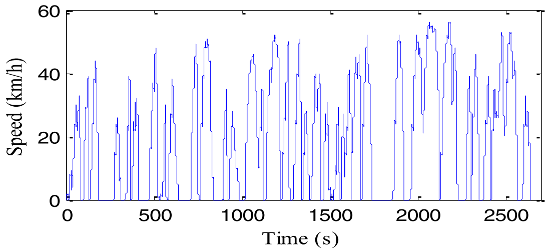


**FIGURE 10.** Driving cycling of the PHEBs.

The driving cycle data is collected by road experiment system which is composed of sensor, data collector and portable computer (PC). The top left part of Fig. 9 is the sensor including the network difference global positioning system (GPS), inertial measurement unit and fuel consumption sensor. The signal is collected by CAN data collector Kvaser II, and sent back to the PC as is presented at the top right of Fig.9. The flowchart of the whole process of collecting data is given in the top half of the Fig.9. The lower half of Fig. 9 shows the experiment platform in the bus. And the associated hardware is represented in this figure.

All the experiments run on a cluster of 8 servers with identical configuration: Dual Intel Xeon E5-2650 processor with the clocks speed at 2.2GHz and 128GB RAM.

The driving cycle for simulation is shown in Fig. 10. In addition, ECMS strategy is widely used in real world application, therefore ECMS strategy is considered as a benchmark for comparison with the proposed method in this paper, which has been used by our team before [35].

The simulation numerical results in Table 3 show the fuel consumption of ECMS and the proposed AECMS method. The results show that the energy consumption generated by the proposed AECMS strategy is significantly lower than that of ECMS strategy; and that the proposed strategy could reduce the energy consumption by giving different parameters in AECMS according to the types of driving cycles which is predicted by constructing the deep neural network.

## VI. CONCLUSION

The main purpose of this paper is to develop deep learning method for improving the accuracy of prediction for different driving cycle types, which could be the first attempt using deep learning to predict the types of driving cycles.

In order to acquire the training data in CNN, k-Shape is used to cluster the driving cycle data. CVI shows that six types of driving cycles are the best choice for k-Shape clustering method. Meanwhile, the result also shows that the performance of k-Shape method is better than that of k-means method.

In the stage of prediction, the input data of deep neural networks is constructed from the raw driving cycle data. Second, this method is developed and its performance on learning a good presentation of driving cycle types is studied. Simulation results show that the proposed method provides higher prediction accuracy than traditional machine learning method. Finally, the results are applied to energy management strategy for PHEB, and better performance of fuel consumption is also obtained. Therefore, deep learning should be a powerful tool for learning driving cycle type features from raw driving cycle data. In addition, deep learning method has great potential to improve the accuracy of prediction of driving cycle types.

## REFERENCES


[1] L. Li, C. Yang, Y. H. Zang, L. P. Zhang, and J. Song, "Correctional DP-based energy management strategy of plug-in hybrid electric bus for city-bus route," *IEEE Trans. Veh. Technol.*, vol. 64, no. 7, pp. 2792–2803, Sep. 2015.

[2] Z. Chen, L. Li, B. J. Yan, C. Yang, C. M. Martinez, and D. P. Cao, "Multimode energy management for plug-in hybrid electric buses based on driving cycles prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2811–2820, Feb. 2016.

[3] Y. J. Huang, C. L. Yin, and J. W. Zhang, "Design of an energy management strategy for parallel hybrid electric vehicles using a logic threshold and instantaneous optimization method," *Int. J. Automot. Technol.*, vol. 10, no. 4, pp. 513–521, 2009.

[4] Z. Chen, L. Y. Zhou, Y. Sun, Z. L. Ma, and Z. Q. Han, "Multi-objective parameter optimization for a single-shaft series-parallel plug-in hybrid electric bus using genetic algorithm," *Sci. China, Technol. Sci.*, vol. 59, no. 8, pp. 1176–1185, 2016.

[5] Z. Chen, R. Xiong, C. Wang, and J. Cao, "An on-line predictive energy management strategy for plug-in hybrid electric vehicles to counter the uncertain prediction of the driving cycle," *Appl. Energy*, vol. 185, pp. 1663–1672, Jan. 2017.

[6] W. Zhou, K. Xu, Y. Yang, and J. H. Lu, "Driving cycle development for electric vehicle application using principal component analysis and K-means cluster: With the Case of Shenyang, China," *Energy Procedia*, vol. 105, pp. 2831–2836, 2017.

[7] A. Fotouhi and M. Montazeri-Gh, "Tehran driving cycle development using the k-means clustering method," *Sci. Iranica*, vol. 20, no. 2, pp. 286–293, 2013.

[8] I. Schifter, L. Diza, R. Rodríguez, and E. López-Salinas, "A driving cycle for vehicle emissions estimation in the metropolitan area of Mexico City," *Environ. Technol.*, vol. 26, no. 2, pp. 145–154, 2005.

[9] A. Capozzoli, M. S. Piscitelli, S. Brandi, D. Grassi, and G. Chicco, "Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings," *Energy*, vol. 157, pp. 336–352, Aug. 2018.

[10] X. Wang, J. Wu, C. Liu, S. Wang, T. Wang, and W. Niu, "Improving failures prediction by exploring weighted shape based time-series clustering," *Qual. Rel. Eng. Int.*, vol. 34, no. 2, pp. 138–160, 2018.

[11] Y. H. Zhang, X. H. Jiao, L. Li, C. Yang, L. P. Zhang, and J. Song, "A hybrid dynamic programming-rule based algorithm for real-time energy optimization of plug-in hybrid electric bus," *Sci. China, Technol. Sci.*, vol. 57, no. 12, pp. 2542–2550, 2014.

[12] Z. Jing, G. Wang, S. Zhang, and C. Qiu, "Building Tianjin driving cycle based on linear discriminant analysis," *Transp. Res. D, Transp. Environ.*, vol. 53, pp. 78–87, Jun. 2017.

[13] Z. Lei, D. Cheng, Y. Liu, D. Qin, Y. Zhang, and Q. Xie, "A dynamic control strategy for hybrid electric vehicles based on parameter optimization for multiple driving cycles and driving pattern recognition," *Energies*, vol. 10, no. 1, p. 54, 2017.

[14] J. Gallus, U. Kirchner, R. Vogt, and T. Benter, "Impact of driving style and road grade on gaseous exhaust emissions of passenger vehicles measured by a Portable Emission Measurement System (PEMS)," *Transp. Res. D, Transp. Environ.*, vol. 52, pp. 215–226, May 2017.

[15] C. M. Martinez, X. Hu, D. Cao, E. Velenis, B. Gao, and M. Wellers, "Energy management in plug-in hybrid electric vehicles: Recent progress and a connected vehicles perspective," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 4534–4549, Jun. 2017.

[16] H. Wang, Y. J. Huang, A. Khaiepour, H. W. He, and D. P. Cao, "A novel energy management for hybrid off-road vehicles without future driving cycles as a priori," *Energy*, vol. 133, pp. 929–940, Aug. 2017.

[17] E. Ericsson, "Independent driving pattern factors and their influence on fuel-use and exhaust emission factors," *Transp. Res. D, Transp. Environ.*, vol. 6, no. 5, pp. 325–345, 2001.
[18] R. Langari and J.-S. Won, "Intelligent energy management agent for a parallel hybrid vehicle-part I: System architecture and design of the driving situation identification process," *IEEE Trans. Veh. Technol.*, vol. 54, no. 3, pp. 925–934, May 2005.
[19] E. Tazelaar, J. Bruinsma, B. Veenhuizen, and P. Bosch, "Driving cycle characterization and generation, for design and control of fuel cell buses," *World Electr. Veh. J.*, vol. 3, no. 1, pp. 812–819, 2009.
[20] Z. Song, J. Hou, S. Xu, M. Ouyang, and J. Li, "The influence of driving cycle characteristics on the integrated optimization of hybrid energy storage system for electric city buses," *Energy*, vol. 135, pp. 91–100, Sep. 2017.
[21] P. Palcu and J. Bauman, "Whole-day driving prediction control strategy: Analysis on real-world drive cycles," *IEEE Trans. Transport. Electrific.*, vol. 4, no. 1, pp. 172–183, Mar. 2018.
[22] J. Yang, C. Ning, C. Deb, F. Zhang, D. Cheong, S. E. Lee, C. Sekhar, and K. W. Tham, "k-Shape clustering algorithm for building energy usage patterns analysis and forecasting model accuracy improvement," *Energy Build.*, vol. 146, pp. 27–37, Jul. 2017.
[23] S. Deb, Z. Tian, S. Fong, R. Wong, R. Millham, and K. K. Wong, "Elephant search algorithm applied to data clustering," *Soft Comput.*, vol. 22, no. 18, pp. 6035–6046, 2018.
[24] J. Zakaria, A. Mueen, and E. Keogh, "Clustering time series using unsupervised-shapelets," in *Proc. IEEE 12th Int. Conf. Data Mining (ICDM)*, Dec. 2012, pp. 785–794.
[25] J. Paparrizos and L. Gravano, "k-shape: Efficient and accurate clustering of time series," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2015, pp. 1855–1870.
[26] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: A survey," *Int. J. Comput. Appl.*, vol. 52, no. 15, pp. 1–9, 2012.
[27] B. Abualhaj, G. Weng, M. Ong, A. Attarwala, F. Molina, K. Büsing, and G. Glatting, "Comparison of five cluster validity indices performance in brain [$^{18}$F]FET-PET image segmentation using $k$-means," *Med. Phys.*, vol. 44, no. 1, pp. 209–220, 2017.
[28] N. Dey, S. Fong, and W. Song, "Forecasting energy consumption from smart home sensor network by deep learning," in *Proc. Int. Conf. Smart Trends Inf. Technol. Comput. Commun.* Singapore: Springer, 2017, pp. 255–265.
[29] S. Fong, J. Li, W. Song, Y. Tian, R. K. Wong, and N. Dey, "Predicting unusual energy consumption events from smart home sensor network by data stream mining with misclassified recall," *J. Ambient Intell. Humanized Comput.*, vol. 9, no. 4, pp. 1197–1221, 2018.
[30] L. Deng and D. Yu, "Deep learning: Methods and applications," *Found. Trends Signal Process.*, vol. 7, nos. 3–4, pp. 197–387, Jun. 2014.
[31] A. Singla and S. Patra, "A fast automatic optimal threshold selection technique for image segmentation," *Signal, Image Video Process.*, vol. 11, no. 2, pp. 243–250, 2017.
[32] L. V. Pérez, G. R. Bossio, D. Moitre, and G. O. García, "Optimization of power management in an hybrid electric vehicle using dynamic programming," *Math. Comput. Simul.*, vol. 73, no. 1, pp. 244–254, 2006.
[33] H. D. Lee and S. K. Sul, "Fuzzy-logic-based torque control strategy for parallel-type hybrid electric vehicle," *IEEE Trans. Ind. Electron.*, vol. 45, no. 4, pp. 625–632, Aug. 1998.
[34] C. Sun, F. Sun, and H. He, "Investigating adaptive-ECMS with velocity forecast ability for hybrid electric vehicles," *Appl. Energy*, vol. 185, pp. 1644–1653, Jan. 2017.
[35] Z. Chen, L. Li, X. Hu, B. Yan, and C. Yang, "Temporal-difference learning-based stochastic energy management for plug-in hybrid electric buses," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2378–2388, Jun. 2019.

**ZHENG CHEN** received the B.S. degree in applied mathematics from Zhejiang University, Hangzhou, in 2000, and the M.S. and Ph.D. degrees in system analysis and integration from the Shanghai University of Science and Technology, in 2008 and 2013, respectively.

From September 2014 to September 2016, he was working as a Postdoctoral Fellow with the State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing, China. He is currently a Lecturer with the School of Sciences, Ningbo University of Technology. His research interests include optimization, vehicle power management of hybrid electric vehicle, and machine learning.

**CHAO YANG** (Member, IEEE) received the Ph.D. degree in control science and engineering from Yanshan University, Qinhuangdao, China, in 2016.

From June 2012 to December 2015, he was a Joint Ph.D. Student, under the Supervision of Dr. L. Li, with the Department of Automotive Engineering, Tsinghua University, Beijing, China, where he was working as a Postdoctoral Fellow with the State Key Laboratory of Automotive Safety and Energy, from January 2016 to September 2018. From April 2017 to April 2018, he was a Visiting Scholar with the University of Victoria, Victoria, BC, Canada. He is currently an Associate Professor with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing. His research interests include energy-efficient control for hybrid electric vehicles, advanced control theory and application on intelligent hybrid vehicles, and nonlinear control for hybrid powertrain. He is currently an Associate Editor of the *IET Intelligent Transport Systems*.

**SHENGNAN FANG** received the B.E. degree in automotive engineering from Tsinghua University, Beijing, China, in 2010, the master's degree in automotive engineering from RWTH Aachen University, in 2012, and the Ph.D. degree in mechanical engineering from Tsinghua University, Beijing, in 2016.

She is currently a Research Associate with the Department of Automotive Engineering, Tsinghua University, Beijing. Her research interests include vehicle dynamics and control and vehicle powertrain modeling and control.