Machine Leaning Based Vehicle Model Construction and Validation - Towards Optimal Control Strategy Development for Plug-in Hybrid Electric Vehicles

Yuanjian Zhang[1], *Member, IEEE,* Zheng Chen[2, 3*], *Senior Member, IEEE,* Guang Li[3], *Senior Member, IEEE,* Yonggang Liu[4], *Senior Member, IEEE,* Haibo Chen[5], Geoff Cunningham[1] and Juliana Early[1]

[1]School of Mechanical and Aerospace Engineering, Queen's University of Belfast, BT9 5AG, Northern Ireland.
[2]Faculty of Transportation Engineering, Kunming University of Science and Technology, Kunming, 650500, China
[3]School of Engineering and Materials Science, Queen Mary University of London, London, E1 4NS, UK
[4]State Key Laboratory of Mechanical Transmission & School of Mechanical and Vehicle Engineering, Chongqing University, Chongqing, 400044, China
[5]Institute for Transport Studies, University of Leeds, Leeds, LS2 9JT, United Kingdom.
Email: y.zhang@qub.ac.uk, chen@kust.edu.cn, g.li@qmul.ac.uk, andylyg@umich.edu, h.chen@its.leeds.ac.uk, g.cunningham@qub.ac.uk, j.early@qub.ac.uk
Corresponding Authors: Zheng Chen (chen@kust.edu.cn) and Yonggang Liu (andylyg@umich.edu)

Abstract: Advances in machine learning inspire novel solutions for the validation of complex vehicle models, and spur an easy manner to promote energy management performance of complexly configured vehicles, such as plug-in hybrid electric vehicles (PHEVs). A constructed PHEV model, based on the four-wheel drive passenger vehicle configuration, is validated through an efficient virtual test controller (VTC) developed in this paper. The VTC is designed via a novel approach based on the least square support vector machine and random forest with the inner-interim data filtered by the ReliefF algorithm to validate the vehicle model as necessary. The paper discusses the process and highlights the accuracy improvements of the PHEV model that is achieved by implementing the VTC. The validity of the VTC is addressed by examining the PHEV model to mimic the characteristics of internal combustion engine, motor and generator behaviors observed through the benchmark test. Sufficient simulations and hardware-in-loop test are employed to demonstrate the capability of the novel VTC based model validation method in practical applications. The major novelty of this paper lies in the development of a VTC, by which the vehicle model can be efficiently developed, providing solid framework and enormous convenience for control strategy design.

Key Words: Machine learning, plug-in hybrid electric vehicles (PHEVs), model construction, virtual test controller (VTC), hardware-in-loop (HIL) test.

I. INTRODUCTION

Hybrid electric vehicles (HEVs) have emerged as an efficient solution to a number of key social issues, such as oil consumption reduction and environmental pollution mitigation [1]. Furthermore, plug-in HEVs (PHEVs), equipped with larger battery packs, have gained wide popularity due to the inherent all-electric driving range (AER) and efficient fuel economy [2]. However, to fully explore energy saving potential of PHEVs, proper control strategies need to be carefully

implemented. To now, there has been significant research progress in design of optimal control strategies [3-5], and a slew of efficient solutions emerge and deliver satisfactory energy consumption economy and emissions without sacrificing drivability and ride comfort. However, most of the methods proposed to date cannot adapt to the time-varying driving conditions and supply optimal performance continuously. While, the extensive development of 5th generation (5G) communication technologies [6] gives rise to the opportunities to further promote in energy control of PHEVs by incorporating more environmental information through vehicle-to-vehicle (V2V) [7] and vehicle-to-infrastructure (V2I) communication [8], as well as enabling vehicle-environment cooperative control. For instance, adaptable vehicle control will be anticipated to achieve minimum energy consumption under different driving conditions and environments. To this end, precise models need to be available, and can replicate the complex vehicle powertrain and system communications, thereby simplifying the development procedure of advanced control algorithms. Typically, these are difficult to achieve and many of the precise details of internal combustion engine (ICE), motor and generator performance may be missed. To tackle it, the model construction and validation of a PHEV configuration based on machine learning approaches is introduced to improve the performance of the underlying vehicle model, providing a solid framework for the development of future connected control strategies.

Vehicle models, including backward and forward manners, play an essential role in control strategy development. The backward models are typically the most convenient for energy estimation. In this approach, the calculation is initiated from wheels, and then the required driving torques are decomposed into different energy paths through the transmission system. For example, Zhou et al. construct a backward model and apply dynamic programming (DP) in the energy management of PHEV to achieve optimal energy distribution between ICE and battery pack [9]. Sun et al. similarly exploit a backward model in model predictive control (MPC) algorithm to generate the reference curve for battery state of charge (SOC) in prediction horizon, and employ the designed MPC to determine the energy distribution in real time [10]. Despite widespread applications of these models, they cannot replicate real control processes. In addition, the backward calculation approach may lead to singular points, resulting in difficulties in application of the model. By contrast, the forward modelling approach starts simulation from the driver module. The control units process driving intention, and the corresponding control commands are generated and transmitted to the power units. The intuitive forward calculation manner has made this type of models widely accepted in the development of modern vehicle control systems. Rizzoni et al. propose a framework for forward modeling, and demonstrate its feasibility in development of HEV control algorithms. The framework is verified efficient in vehicle analysis and control algorithm design [11]. On

the basis of this work, a number of forward simulation tools emerge for HEV design. Lin et al. develop an integrated forward hybrid electric simulation tool (HE-VESIM) to mimic the dynamic performance of parallel HEV [12]. The results reveal that the forward models can simplify the vehicle design tremendously. However, they rely on large volumes of component data and detailed understanding of the control system intent, leading to certain difficulties in achieving high precision modelling.

Recent developments in artificial intelligence, particularly machine learning, have provided different insights in problems across many disciplines. A number of examples have appeared in terms of machine learning algorithms applied to the design of PHEVs [13, 14], which have accelerated the development of intelligent control algorithms. Guo et al. employ a deep neural network (DNN) to predict future short-term velocity profiles with preferable accuracy, supplying valuable references for energy management of PHEV through MPC [15]. Sun et al. [16] apply several velocity prediction methods, including neural networks (NN), and highlight the overwhelming performance of NN. With respect to the strategy design, Hu et al. leverage the offline optimization results obtained by Pontryagin's minimum principle (PMP) to train NN, which is then implemented in real-time control to resolve the energy distribution [17]. Li et al. exploit the deep reinforcement learning (DRL) to attain the advanced energy management for serial HEV without the knowledge of future driving conditions [18]. The simulation results showcase the superior performance of DRL method. Despite these promising applications, the application of machine learning in vehicle modelling, to the authors' knowledge, has not been investigated in-depth.

The development of a precise forward model is a vital step in control strategy design, particularly for PHEV. In this study, a forward model for 4-wheel drive (4WD) PHEV is constructed based on the data collected through a series of benchmark tests on a passenger car configuration. The least square support vector machine (LSSVM), random forest (RF) and ReliefF algorithm are integrated to develop a virtual test controller (VTC), which is then employed to validate the model with demonstrable improvements in accuracy. The simulation and hardware-in-loop (HIL) test are performed to validate the feasibility and optimality of the raised VTC in both simulation and practical applications. Different PHEVs are also employed to manifest the anticipated performance of the designed VTC. The major novelty of this study lies in the development of a VTC based on machine learning techniques. The detailed contributions of this work include:

1) A novel model validation method is developed by fully exploring advanced machine learning algorithms, demonstrating the great potential of machine learning methods in engineering practice and inspiring new pathway to

state-of-the-art EV development. The validated model, with the enhanced accuracy in limning behaviors of real PHEV, can underpin further resilient EV development.

2) A novel VTC is constructed by a hybrid machine learning scheme after carefully investigating the characteristics of different methods. Compared with single scheme, the designed LSSVM-RF scheme can fully reap the merits of LSSVM in small sampling learning and RF in ensemble learning, and is therefore more suitable for VTC in model validation with small-scale inputs and complex system dynamics.

3) Comprehensive compare studies between representative advanced and traditional methods in simulation scenarios and practical situations are performed to demonstrate the promising performance of the designed model validation method. Compared with single-scheme based VTC, the developed LSSVM-RF hybrid scheme based VTC manifests significant potential in precisely delineating real PHEV behaviors and robust model validation.

The remainder of this study is organized as follows. The 4WD PHEV and benchmark test are introduced in Section II. Section III elaborates the newly constructed forward model and the model validation method adopted in this study. Section IV evaluates the novel VTC and discussed model validation results, followed by the main conclusions drawn in Section V.

II. 4WD PHEV AND BENCHMARK TEST

The studied 4WD PHEV, as illustrated in Fig. 1, includes one ICE, one generator and two traction motors (designated as motor 1 and motor 2). The vehicle can operate in different modes through the cooperation among ICE, motors and generator, to provide better drivability and potentially greater energy savings, compared with the ordinary two-wheel drive (2WD) PHEVs. The designed operation modes include: pure electric mode (also called EV mode), serial mode and hybrid mode. In the serial mode, ICE and generator are known collectively as an auxiliary power unit (APU). The studied 4WD PHEV, with the format of multiple operation modes and advanced powertrain mechanic-electric dynamics, offers a high-quality platform to investigate the state-of-the art model construction technologies. The comprehensive benchmark tests lay solid support for data-driven based model construction and validation.

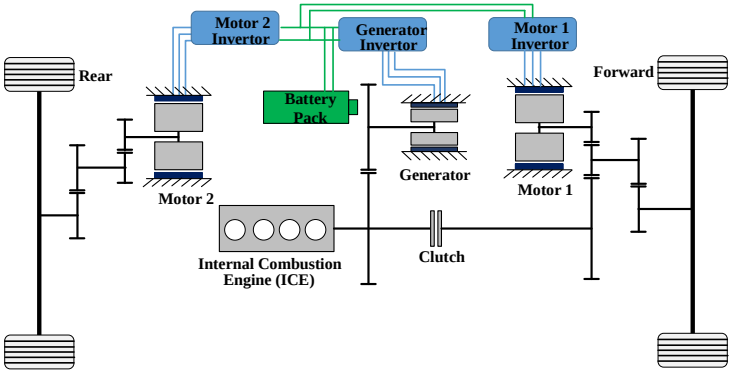


Fig. 1. The schematic of the 4WD PHEV configuration.

The benchmark test, as shown in Fig. 2, is one of the preferred methods for analysis and performance evaluation of vehicles [19]. In this study, the benchmark test is performed on a dynamometer, and involves constant acceleration test, constant speed test, and standard driving cycle test [20]. In addition, the ICE, motors, generator and battery pack are also calibrated. The raw test data are collected through data logger with the sensors for analog and digital quantities, power analyzer and controller area network (CAN) channel. Through benchmark test, different look-up tables can be constructed in terms of data elements related to the performance of powertrain components, mode switch rules and general energy management manners between different sources.


Fig. 2. Benchmark test on the studied 4WD PHEV.

III. 4WD PHEV Model Construction and Validation

A. 4WD PHEV Model Construction

A forward 4WD PHEV model is built in MATLAB/Simulink, and the general framework of the constructed model is shown in Fig. 3. As can be found, there exist four sub-modules: driver, plant, controller and CAN bus. The driver module interprets the driving behavior and outputs the specific driving requirement (power/torque demand) to the controller module. The controller module determines the most suitable operation mode based on the driver demand and current vehicle state, and distributes energy demand among ICE, generator and motors accordingly. The plant module dynamically responds to the given control orders from the controller module, and supplies the actual execution details. The CAN bus module transmits signals among driver, plant and controller modules, and visually displays them. Accordingly, the output signals from the driver module include the acceleration and braking pedal degrees, which are

sent to the controller module for control policy decision. The signals departing from the controller module include the desired motor torque, ICE throttle, target generator load, requested regenerative braking torque, clutch engagement order, and ICE ignition command, which are sent to the plant module for response to the driving requirement. The signals sent out from the plant module involve the instant component and vehicle state, such as SOC, current and voltage of battery, torque, power and speed of ICE, power, current and speed of the motors/ generator, clutch speed, etc. In the CAN bus module, the specific blocks are constructed to collect, display and store signals from other modules. Table I lists the detailed parameters of electric powertrain in the studied PHEV.

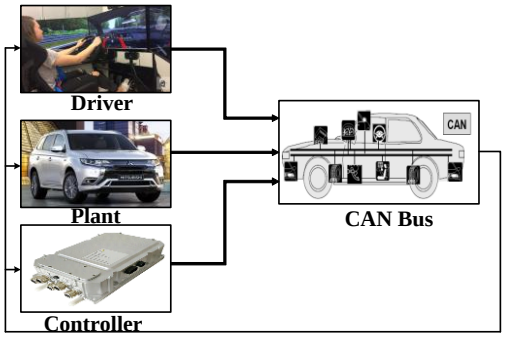


Fig. 3. The schematic of the forward model.

TABLE I
COMPONENT PARAMETERS IN THE STUDIED 4WD PHEV POWERTRAIN

ICE	**Displacement**	**2.0 [L] 16V DOHC**
	Maximum Power	89 [kW] @4500 [rpm]
	Maximum Torque	190 [Nm] @4500 [rpm]
Motor 1	Maximum Power	60 [kW]
	Maximum Torque	137 [Nm]
Motor 2	Maximum Power	60 [kW]
	Maximum Torque	195 [Nm]
Battery	Type	Lithium-ion
	Capacity	12 [kWh]
	Nominal Voltage	300 [V]
	Maximum Battery Power	147 [kW]
	Permitted Battery SOC Variation Range	0.22-0.8
Gear Ratio	Between ICE and final drive	i_{g1}=3.425
	Between motor 1 and final drive	i_{g2}=9.663
	Between motor 2 and final drive	i_{g3}=7.065
	Between ICE and generator	i_{g4}=2.736

The mathematical equations used to describe the simulation process in each component are described below.

a. Driver Model

The driver model prescribes the desired driving behaviors (velocity and acceleration/deceleration intension) and generates the corresponding driving requirement (typically power/torque demand), which will be sent to the controller module. The model input is the target velocity based on the driving cycle data and current feedback velocity, and the model output includes the degree of acceleration pedal and brake pedal positions. In most cases, the driver is represented

by a proportional-integral-differential (PID) controller (simplified to the proportional control herein), as:

$$\begin{cases} P_{acc}(t) = K_p(v_{target}(t+1) - v_{real}(t)) \\ P_{brake}(t) = K_p(v_{real}(t) - v_{target}(t+1)) \end{cases} \tag{1}$$

where P_{acc} and P_{brake} are the desired acceleration and braking pedal positions, v_{target} and v_{real} denote the target velocity and current velocity, K_p is the scale factor, and t is the time step. K_p is tuned based on the average difference between the velocity of real vehicle and that of the constructed model. The tuning process is terminated after the average velocity difference is smaller than 1.5 km/h. When significant velocity error emerges, the constructed forward model cannot simulate the performance of real vehicle accurately, resulting in the deviation in powertrain dynamics and energy consumption.

b. ICE Model

A static model based on the efficiency map obtained through benchmark test, shown in Fig. 4, is employed to characterize the nonlinear fuel consumption performance of ICE. The instantaneous fuel consumption can be calculated, as:

$$\dot{m}_f(t) = f(T_{out}(t), n_{eng}(t)) \tag{2}$$

where $\dot{m}_f$ is the fuel consumption rate, T_{out} denotes the ICE torque, and n_{eng} represents the ICE speed. The load L_{eng} can be calculated, as:

$$L_{eng}(t) = \frac{[P_{eng\_re}(t+1) \times 9550] / n_{eng}(t)}{T_{table\_eng\_100\%}(n_{eng}(t))} \tag{3}$$

where P_{eng_re} is the required ICE power, and $T_{table\_eng\_100\%}$ is the maximum torque corresponding to the current ICE speed.

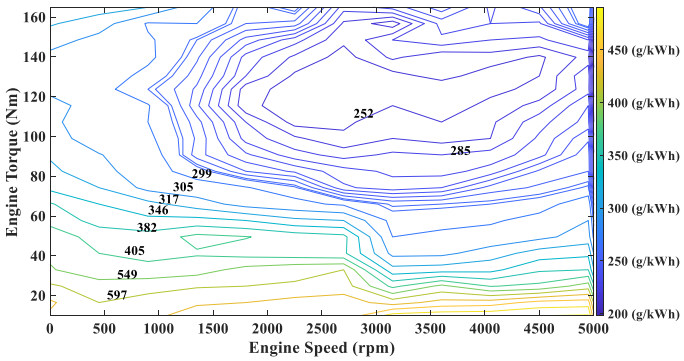


Fig. 4. Fuel rate map of ICE.

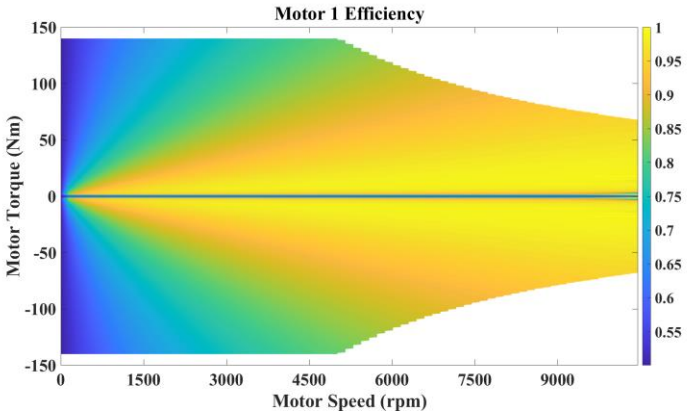
c. Motor and Generator Model

The motors and generator in the studied PHEV are permanent magnet synchronous motors (PMSMs). For ease of modeling the vehicle efficiently, the dynamic characteristics of motors are neglected. Likewise, the static models based on efficiency maps acquired through benchmark test are exploited to describe the power performance of motors and generator. Fig. 5 illustrates the efficiency maps of motors and generator. The efficiency map of generator is the same with that of motor 1. The load of motors and generator can be calculated, as:

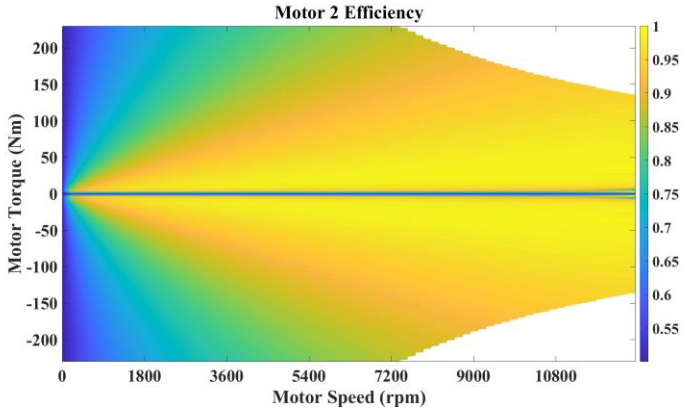
$$\begin{cases} L_{mot}(t) = \dfrac{[P_{mot\_re}(t+1)\times 9550]/n_{mot}(t)}{T_{table\_mot\_100\%}(n_{mot}(t))} \\ L_{gen}(t) = \dfrac{[P_{gen\_re}(t+1)\times 9550]/n_{gen}(t)}{T_{table\_gen\_100\%}(n_{gen}(t))} \end{cases} \tag{4}$$

where L_{mot}, P_{mot_re}, n_{mot} and $T_{table\_mot\_100\%}$ denote the load, required power, speed and maximum torque of motor; L_{gen}, P_{gen_re}, n_{gen} and $T_{table\_gen\_100\%}$ present the same quantities of the generator. The kinematic relationship between engine and motor/generator can be described, as:

$$\begin{cases} n_{eng} i_{g3} = n_{gen} \\ n_{eng} i_{g2} = n_{mot} i_{g1} \end{cases} \tag{5}$$



(a)



(b)

Fig. 5. The motor efficiency maps. (a) Efficiency map of motor 1 and generator. (b) Efficiency map of motor 2.

d. Battery Model

For modelling battery, the temperature and aging influence are neglected in this study, and a simple equivalent circuit model is constructed to characterize the battery electrical performance. The model consists of an open circuit voltage source, varying nonlinearly with SOC, and an internal resistor connected in series manner [21]. By analyzing the electric performance of the model, the variation rate of SOC, i.e., the battery current, can be formulated as:

$$\dot{SOC}(t) = -\frac{V_{oc}(t) - \sqrt{V_{oc}(t) - 4R_{\text{int}}(t)P_{batt}(t)}}{2R_{\text{int}}(t)Q_{batt}} \tag{6}$$

where SOC is the battery SOC, V_{oc} is the open circuit voltage, R_{int} is the internal resistance, P_{batt} denotes the battery power, and Q_{batt} is the battery capacity. Typically, V_{oc} is obtained via interpolating the look-up table which describes the relationship with respect to other indexes, such as battery SOC and temperature. For simplifying the calculation, V_{oc} is assumed to be a constant in this paper. The battery SOC can be obtained by the Coulomb counting method, as:

$$SOC(t) = SOC_{ini} - \frac{\int_0^t i_{batt}}{Q_{batt}} \tag{7}$$

where SOC_{ini} means the initial battery SOC, and t denotes the time step.

e. Control Strategy in Vehicle Model

The logic of controlling mode transition in the studied PHEV is designed by a rule-based method based on the benchmark test. The mode switch is determined by the current vehicle speed *v*, the required tractive power P_{req} and the required tractive torque T_{req}. In general, the energy management principle is to regulate the ICE to operate in the brake-specific fuel consumption (BSFC) line based on the load following method. The energy management strategy in both serial and parallel modes and in charge depleting (CD) and charge sustaining (CS) stage can be expressed in Table II.

TABLE II
ENERGY MANAGEMENT STRATEGY IN CD AND CS STAGE

Mode	Condition	P_{apu}	P_{batt}
Serial	$P_{req} < P_{apu\_opt}$	0	P_{req}
	$P_{apu\_opt} \le P_{req} < P_1$	P_{apu_opt}	$P_{req} - P_{apu\_opt}$
	$P_1 < P_{req}$	P_{apu_max}	$P_{req} - P_{apu\_opt}$
Parallel	$P_{req} < P_{eng\_opt}$	0	P_{req}
	$P_{eng\_opt} \le P_{req} < P_2$	P_{eng_opt}	$P_{req} - P_{eng\_opt}$
	$P_2 < P_{req}$	P_{eng_max}	$P_{req} - P_{eng\_opt}$

In Table II, P_{apu_opt} and P_{eng_opt} are the power corresponding to the optimal operation points of APU and ICE; P_{apu_max} and P_{eng_max} express the maximum power under their specific speeds; $P_{batt\_max\_CD/CS\_s}$ and $P_{batt\_max\_CD/CS\_P}$ represent the

maximum power limit of battery in CD and CS stage when the vehicle is in serial mode and in parallel mode; P_1 equals the sum of P_{apu_opt} and $P_{batt\_max\_CD/CS\_s}$, and P_2 equals the sum of P_{apu_opt} and $P_{batt\_max\_CD/CS\_s}$. The APU, integrating ICE and generator, enables them to operate together to supply tractive power in serial mode. P_{apu_opt} is calculated based on the combined efficiency table data of ICE and generator, and P_{eng_opt} is calculated by means of looking up the efficiency table of ICE. Note that both efficiency tables are obtained via benchmark test. Additionally, P_{apu_opt} and P_{eng_opt} in Table II are 51 kW and 55 kW, respectively.

B. Model Validation

In view of forward vehicle models developed for academic research or third-party applications, it may be difficult to replicate the control logic existing in physical vehicles. In addition, the vehicle's plant performance may differ between the model and real vehicle due to idealization or neglection of transient and nonlinear behaviors which are difficult to model mathematically. Although the well-designed benchmark testing can provide the elaborate depiction for real vehicle operation performance and supply all-rounded information for modeling, the overall accuracy of the built model still needs to be validated and examined. A majority of traditional methods to validate vehicle forward-facing models are based on expert knowledge, wherein the time-consuming manual recalibration depends on subjective understanding on difference between model outputs and real test results. The deviation in understanding on real test results may lead to deficient recalibration, whereas machine learning based models exhibit obvious advantages in regression analysis, providing alternative pathways in multi-aspect engineering applications. In this study, a novel model validation method, incorporating model verification and recalibration, is proposed based on machine learning algorithms. The flowchart of the proposed validation method is shown in Fig. 6.

To validate the model's performance, the same driving cycles are imported into the trained VTC and the constructed forward model, and the component performance in the model is minutely compared with that yielded by VTC. The model verification and recalibration are performed according to the difference between the results from the forward model and VTC. In the model validation, the vehicle plant module and controller module are mainly examined, as their key parameters in both modules are derived through benchmark test. The parameters in the driver module are determined to guarantee the tracking ability under various conditions. Since the driver module is not affected by benchmark results, it is ignored in model validation. The examples of the forward model validation will be described in Section IV. During validation, the VTC construction is an essential step, which should comply with the following rules:

1) VTC should exhaustively reproduce the control strategy of the original vehicle to validate the implementation of the control strategy in the mathematical forward model;

2) VTC can evaluate the consistency between the output of the vehicle plant module and the actual performance of the original vehicle;

3) VTC is enabled to recalculate the parameters in the control module to improve the control accuracy, if required.

Following the prescribed rules, a novel VTC is built by cooperatively employing the LSSVM, RF and ReliefF algorithms. The LSSVM accounts for generating the meritorious interim variables that will become the inputs of the RF based regression analysis after capturing the given inputs of the novel VTC. The outputs of VTC, such as ICE torque, generator torque, and motor 1 torque, are generated by the RF algorithm according to the imported interim variables. To prompt the efficiency of RF based regression, the ReliefF algorithm is exploited to extract the worthy interim valuables that are strongly related to the performance of the control strategy in the studied 4WD PHEV [22]. During applications, the inputs of the novel VTC include the instant vehicle velocity, acceleration, and deceleration. The novel VTC, which is the core component in the designed model validation method, is carefully developed based on the LSSVM-RF based hybrid scheme. This is because the performance of LSSVM in practice will deteriorate with the increase of data labels when dealing with data mapping of complex process; and by contrast, RF, with the distinguished ensemble learning manner, excels at handling data with multiple labels. By this manner, the proposed method obtains qualified capabilities in regression analysis. The co-operation of LSSVM and RF to construct the VTC mitigates the pressure on data preparation before training, as only three types of data are required. Large scale of data goes against the performance promotion of machine learning algorithm application, due to the complex interactions and interferences. By virtue of the joint method, only a small amount of training data is entailed, and what is more, the preferable regression performance is still anticipated. Next, the adopted ReliefF algorithm, LSSVM and RF method will be elaborated.

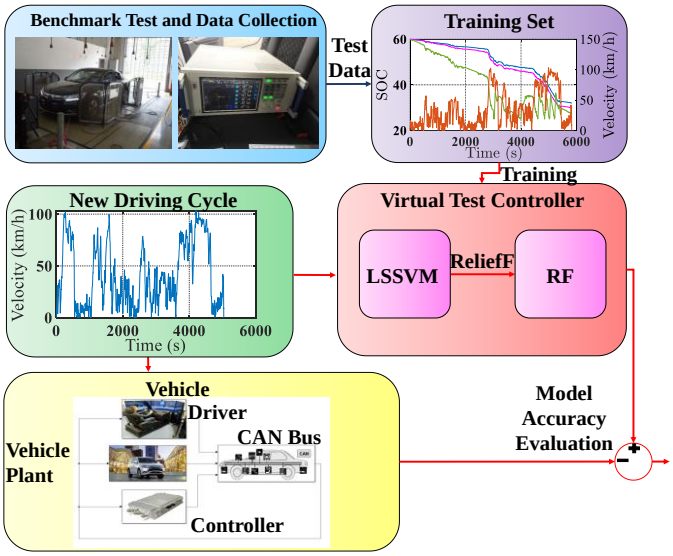


Fig. 6. The flowchart of model validation.

a. ReliefF Algorithm

The ReliefF algorithm is proved efficient in processing multi-classification problems [23]. The main role of the ReliefF algorithm in this paper is to select the appropriate input parameters for the LSSVM and RF based VTC. During implementation [24], the algorithm seeks k variants from its nearest neighbors with the same class after being provided with the randomly selected instance R_i, and then calls the nearest hit H_j and k-nearest neighbors from other classes, referred to as the nearest misses $M_j(C)$. The weight of all attributes A is then calculated according to R_i, H_j and $M_j(C)$, as:

$$W[A] = W[A] - \frac{\sum_{j=1}^{k} diff(A, R_i, H_j)}{(mk)} + \frac{\sum_{C \neq class(R_i)} \frac{P(C)}{1 - P(class(R_i))} \sum_{j=1}^{k} diff(A, R_i, M_j(C))}{(mk)} \tag{8}$$

where $P(class(R_i))$ represents the probability of R_i [23]. $diff(A,R_i,H_j)$ calculates the difference between the feature for R_i and H_j, and can be calculated, as:

$$diff(A, R_i, H_j) = \frac{|value(A, R_1) - value(A, H_2)|}{\max(A) - \min(A)} \tag{9}$$

where $value(A, R_1)$ denotes the difference between A and R_1, $value(A, H_2)$ is the difference between A and H_2. For the variable j starting from 1, if the distance between R_i and H_j on a certain feature is less than that between R_i and $M_j(C)$, it can be reasonably judged that the model shows a strong capability to distinguish the class, resulting in the increase of the relative feature weighting. When the distance between R_i and H_j is greater than that between R_i and $M_j(C)$, it means the feature behaves poorly in discriminating the similar kinds of nearest neighbors. As a result, the relative weighting is decreased. Fig. 7 shows the feature selection results by the ReliefF algorithm. According to the feature selection results, benchmark test results and inputs of LSSVM, the battery SOC, the speeds of motors, generator and ICE, battery current, and battery voltage tend to be highly correlated with the control performance, and are therefore chosen as the input of the RF based model. To validate the forward model, the torques of ICE, motor 1 and generator are chosen as the output of RF based model. The corresponding data are extracted to train the VTC.

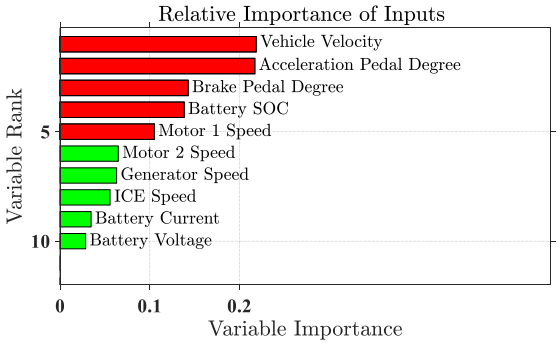


Fig. 7. The flowchart of model validation.

b. LSSVM Model

The LSSVM algorithm is exploited due to its ability of strong regression and classification analysis with few input data labels [25]. To apply LSSVM, a regression model with the initial weight space $\boldsymbol{\omega}$ can be formulated, as:

$$Y(X) = \boldsymbol{\omega}^T \varphi(X) + b \tag{10}$$

where $x \in R^n$, $y \in R$, and $\varphi : R^n \to R^{nh}$ denotes the mapping relationship to the high dimension feature space. For the training set, the sample X can be defined as $\{x_i, y_i\}_{i=1}^{l}$. Thus, a regression model can be built based on the structural minimization principle, as:

$$minJ\left(\boldsymbol{\omega}, e\right) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{\gamma}{2}\sum_{i=1}^{m} e_i^2 \tag{11}$$

subject to:

$$y_i = \boldsymbol{\omega}^T \varphi(x_i) + b + e_i \mathrm{F} \tag{12}$$

where γ is the margin parameter and e_i is the slack variable for x_i. The optimization problem in (11) can be solved by transforming the constraint problem into an unconstrained problem after introducing the Lagrange multiplier α_i, as:

$$L\left(\boldsymbol{\omega}, b, e, \alpha\right) = J\left(\boldsymbol{\omega}, e\right) - \sum_{i=1}^{m} \alpha_i \left\{\boldsymbol{\omega}^T \varphi\left(x_i\right) + b + e - y_i\right\} \tag{13}$$

According to the Karush-Kuhn-Tucker (KKT) condition, the optimal condition can be obtained by taking partial derivation of (13) with respect to $\boldsymbol{\omega}, b, e$ and α, as:

$$\begin{cases} \omega = \sum_{i=1}^{m} \alpha_i \varphi(x_i) \\ \sum_{i=1}^{m} \alpha_i = 0 \\ \alpha_i = \gamma e_i \\ \omega^T \varphi(x_i) + b + e - y_i = 0 \end{cases} \tag{14}$$

Then, the linear correlation can be attained as:

$$\begin{bmatrix} 0 & -Y^T \\ Y & ZZ^T + \dfrac{1}{\gamma} \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{15}$$

By defining the kernel function $K(x, x_i) = \varphi(x)^T \varphi(x, x_i)$, the LSSVM regression can be rewritten into:

$$f(x) = \sum_{i=1}^{m} \alpha_i K(x, x_i) + b \tag{16}$$

c. RF Method

The RF is an ensemble learning method which has been successfully applied in various fields [26]. The popularity of RF is attributed to its easy adaption to a wide variety of regression or classification problems. In the regression problem [27], by assuming that a set of input random vector $\mathbf{X} \in \mathrm{X} \subset c^p$ can be observed, RF is applied to predict the square integral random response $\mathbf{Y} \in \mathrm{R}$ by reckoning the regression function $m(x) = \mathrm{E}(Y|\mathbf{X} = \mathrm{x})$. In addition, the training sample $\mathrm{D}_n = \left((X_1, Y_1), \ldots, (X_n, Y_n)\right)$ needs to be determined from the independent random variables that are distributed with $(\mathbf{X}, \mathbf{Y})$. Based on the training sample, the estimation m_n: $\mathrm{X} \to \mathrm{R}$ can be constructed. The RF consists of a collection of *M* randomized regression trees. In this paper, the classification and regression tree (CART) is used in the RF algorithm [28]. For the *j*th tree in the forest, the predicted value at point **x** is described by $m_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)$, where $\Theta_1, \ldots, \Theta_M$ are random variables with the same distribution with D_n. The output of the *j*th tree can be expressed, as:

$$c = \sum_{i \in \mathrm{D}_n^*(\Theta_j)} \frac{\mathbf{X}_i \in A_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)^{Y_i}}{N_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)} \tag{17}$$

where $\mathrm{D}_n^*(\Theta_j)$ is the set of training data, $A_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)$ denotes the cell containing **x**, and $N_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)$ represents the number of points falling into $A_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)$. Finally, the finite estimation is made by combining the output of each tree:

$$m_{M,n}\left(\mathbf{x}, \Theta_1, \ldots, \Theta_M, \mathrm{D}_n\right) = \frac{1}{M} \sum_{j=1}^{M} m_n(\mathbf{x}, \Theta_j, \mathrm{D}_n) \tag{18}$$

The pseudo code of RF algorithm is provided in Table III, where $nodesize \in \{1, \ldots, a_n\}$ denotes the number of examples in each cell, and $mtry \in \{1, \ldots, p\}$ is the number of possible directions for splitting at each leaf node. The CART-split criterion can be written as follows [27]. For any $(j, z) \in \mathrm{C}_A$,

$$L_{reg,n}(j, z) = \frac{1}{N_n(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_A\right)^2 |_{X_i \in A} - \frac{1}{N_n(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_{AL} |_{X_i^{(j)} < z} - \bar{Y}_{AR} |_{X_i^{(j)} \geq z}\right)^2 |_{X_i \in A} \tag{19}$$

where C_A is the set of all possible cut, $A_L = \{x \in A : x^{(j)} < z\}$, $A_R = \{x \in A : x^{(j)} \geq z\}$, and $\bar{Y}_A (\mathrm{resp.}, \bar{Y}_{AL}, \bar{Y}_{AR})$ means the average of Y_i. For each cell *A*, the best cut (j_n^*, z_n^*) is chosen according to the maximum $L_{reg,n}(j, z)$ over $\mathrm{M}_{\mathrm{try}}$ and C_A, as:

$$(j_n^*, z_n^*) = \arg\max L_{reg,n}(j, z) \tag{20}$$

TABLE III PSEUDO CODE OF RF

for $j = 1, \ldots, M$ **do**
  Pick up a_n points in D_n
  Set $\mathrm{P} = (\mathrm{X})$
  Set $\mathrm{P}_{final} = \varnothing$
  while $\mathrm{P} \neq \varnothing$ **do**
    Let A be the first element of P
    if A contains less than *nodesize* points or if $\mathbf{X}_i \in A$ are equal **then**
      Remove the cell A from P
      $\mathrm{P}_{final} \leftarrow Concatenate\ (\mathrm{P}_{final}, A)$
    else
      Pick up a subset $\mathrm{M}_{\mathrm{try}} \subset \{1, \ldots, p\}$ if cardinality *mtry*
      Choose the most suitable spilt in A by the CART-split criterion
      Cut the cell A according to the split. Name A_L and A_R for the two split cells.
      Remove the cell A from the list P
      $\mathrm{P} \leftarrow Concatenate(\mathrm{P}, A_L, A_R)$
    end
  end
  Compute the estimation value $m_n(\mathbf{x}, \Theta_j, \mathrm{D}_n)$ at $\mathbf{x}$
end

IV. EVALUATION ON NOVEL VTC AND MODEL VALIDATION

The evaluation process on the raised novel VTC and model validation method is divided into multiple steps. Firstly, the effectiveness of the LSSVM and RF based VTC is comparatively studied with several benchmark methods in the simulation environment. After verifying the capacities of the designed VTC in reproducing behaviors of benchmark vehicle, the VTC based model validation is launched successively in simulation. Then, the validated vehicle plant module and controller module in the 4WD PHEV forward vehicle are examined in the HIL test platform to demonstrate the feasibility of the raised method in practical environment. Finally, some discussions in terms of the simulation and HIL validations as well as the applications in different PHEVs are conducted. Note that the torque distribution ratio between motors 1 and 2 is fixed to 0.5 in the derived control strategy. Therefore, only the performance of motor 1 is presented during performance evaluation and comparison.

A. Evaluation on LSSVM and RF Based VTC in Simulation

To better assess the LSSVM and RF based VTC, a number of ordinary methods are also employed. In this study, the long-short term memory (LSTM) network [29], LSSVM, back propagation neural network (BPNN) [30] and bidirectional LSTM (Bi-LSTM) [31] are all trained with the same derived through the benchmark test to develop VTCs. The training process will be terminated once the root mean square error (RSME) is less than a certain threshold. The LSSVM in this evaluation is also used to generate the results directly. Besides, a traditional system identification method supplied by system identification toolbox (SIT) in Matlab [32] is also enrolled in the compare study. The adopted machine

learning models are trained by the collected data from a group of benchmark tests. Each group of benchmark test is performed on one standard driving cycle, and the different initial conditions lead to 1500 test groups in total. 70% of the collected data from 1500 groups of benchmark tests is utilized to train the machine learning model, and the remaining 30% data is employed to validate the training effectiveness. Note that the test marked as LSSVM and RF denotes the novel method developed in this work. Figs. 8 to 10 illustrate the estimation errors in engine, generator and motor 1 torque between real test data and different VTCs under the fixed driving cycle. As can be found, the LSSVM and RF based VTC estimates torque output most accurately, compared with other methods, and the torque estimation errors are quite close to those of real benchmark test. The SIT, BP-NN and single LSSVM all raise noticeable deviations from real torque values. The remaining methods, including the raised LSSVM and RF method, LSTM and Bi-LSTM, show strong capabilities in mimicking the real vehicle dynamics. For the traditional SIT, it is quite labored to capture the complex nonlinear dynamics of electric powertrain under various scenarios. Figs. 11 to 13 present the distributions of estimation errors in engine, generator and motor 1 torque. As can be found, LSSVM and RF based VTC contributes to less errors (<5 Nm) in the torque estimation of engine, generator and motor 1. Bi-LSTM achieves the closest performance to the raised LSSVM and RF based VTC method. With the descending capability in regressing behaviors of electric powertrain, larger errors (>10 Nm) are incurred by the methods including SIT, BP-NN and single LSTM.

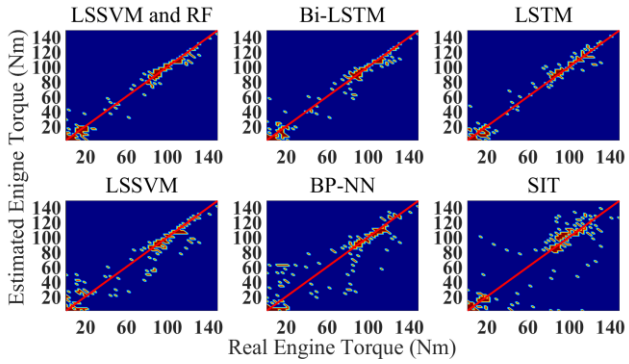


Fig. 8. The scatter plot of engine torque estimation results by different VTCs.

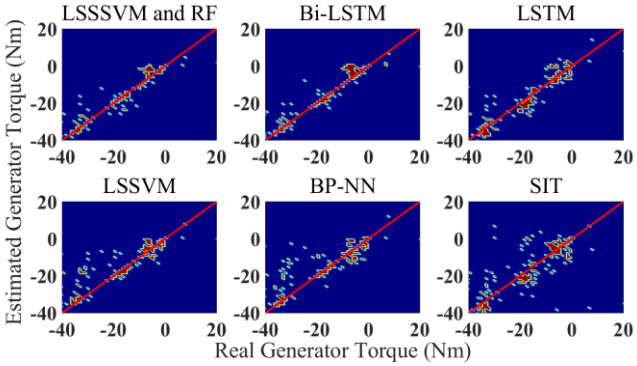


Fig. 9. The scatter plot of generator torque estimation results by different VTCs.

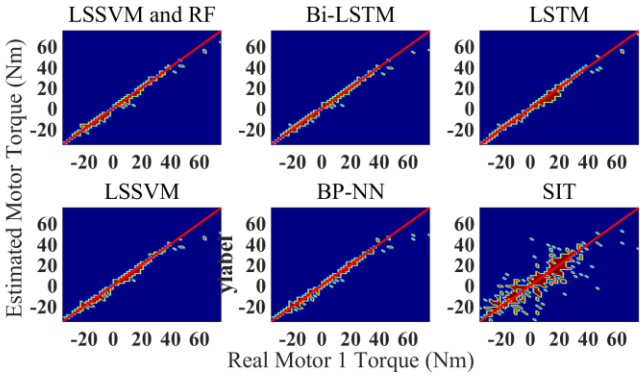


Fig. 10. The scatter plot of motor torque estimation results by different VTCs.

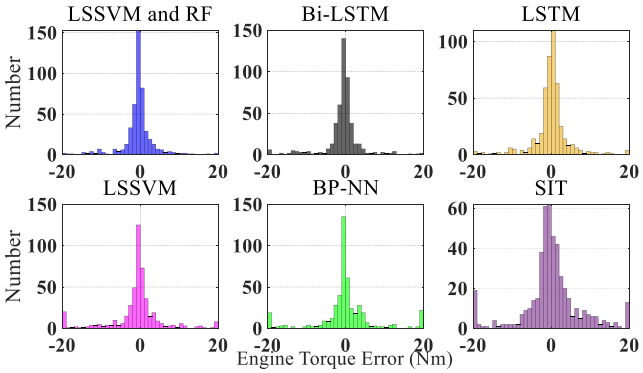


Fig. 11. The histogram of engine torque error distribution by different VTCs.

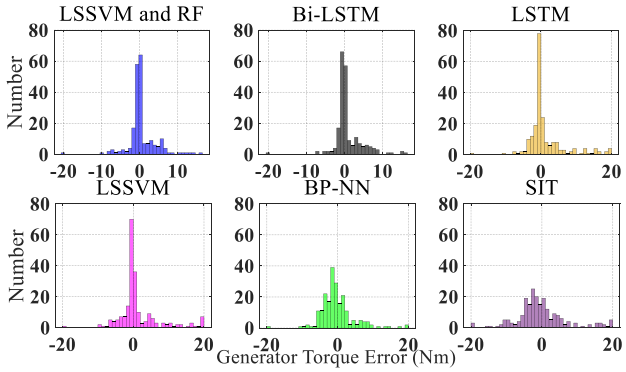


Fig. 12. The histogram of generator torque error distribution by different VTCs.

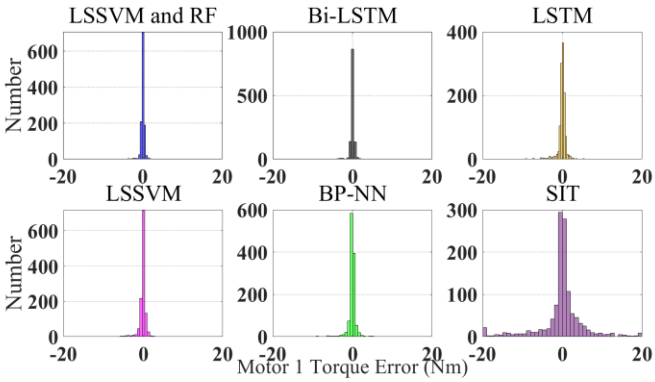


Fig. 13. The histogram of motor 1 torque error distribution by different VTCs.

The improved performance by the LSSVM and RF based VTC can be attributed to the incorporated advantages of LSSVM and RF. LSSVM is known to perform well in processing regressions with few available data labels. During VTC construction, the input includes only the driving cycle data, resulting in that the input data is assigned with the same

label. For RF, the strong performance can be anticipated when multiple input data labels exist. By integrating LSSVM and RF, the number of input data labels is increased, thus promoting full capabilities of RF algorithm and compensating the error induced due to the poor data distribution. Table IV lists the detailed comparison of results by different methods, including RMSE and mean absolute error (MAE) of the observed results, which can be formulated, as:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(h(x_i)-y_i)^2}{N}} \tag{21}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|h(x_i)-y_i\right| \tag{22}$$

where $h(x_i)$ is the estimated value, and y_i is the ground truth data. As can be found, the LSSVM and RF fused method yields better performance, i.e., lower RMSE and MAE, compared with the remaining methods tested. The comparison study highlights the optimal performance of LSSVM and RF based VTC, providing superiority in model validation by the trained VTC.

TABLE IV
TORQUE ESTIMATION DIFFERENCE BY DIFFERENT METHODS

Items	Method	RMSE	MAE
ICE Torque	LSSVM and RF	0.9037	0.2710
	Bi-LSTM	0.9102	0.2803
	LSTM	0.9671	0.3914
	LSSVM	1.1846	0.5674
	BP-NN	1.5391	0.8865
	SIT	2.1463	1.2597
Generator Torque	LSSVM and RF	3.1285	1.1002
	Bi-LSTM	3.1016	1.0743
	LSTM	3.2736	1.3911
	LSSVM	3.5273	1.6006
	BP-NN	4.2464	2.2800
	SIT	4.9152	3.0165
Motor 1 Torque	LSSVM and RF	1.9748	0.4340
	Bi-LSTM	1.9616	0.4416
	LSTM	2.0543	0.6712
	LSSVM	2.2919	0.9110
	BP-NN	2.3925	1.1580
	SIT	2.9481	2.1671

In Table IV, Bi-LSTM seems to behave quite closely to the proposed method. To better illustrate the behaviors of the adopted VTCs, Figs. 14 and 15 show the torque estimation distribution and estimation error of each step under the specific driving cycle by LSSVM and RF method and Bi-LSTM, of which the latter is obtained by subtracting the real value obtained through the benchmark test. Clearly, the novel LSSVM and RF based VTC can result in smaller estimation error and more concentrated scatter results than Bi-LSTM. The remarkable performance of Bi-LSTM owns to its advanced memory framework in processing regression analysis. In fact, the LSTM and Bi-LSTM, developed based on

recurrent neural network (RNN) model, reach state-of-the-art performance in processing temporal data intrinsically. For the VTC application in model validation, however, it reproduces control orders of benchmark vehicle with the given inputs of each control step, which is not suitable to apply time-sequenced training samples. From this point of view, the LSSVM and RF method are more appropriate for the VTC application in model validation.

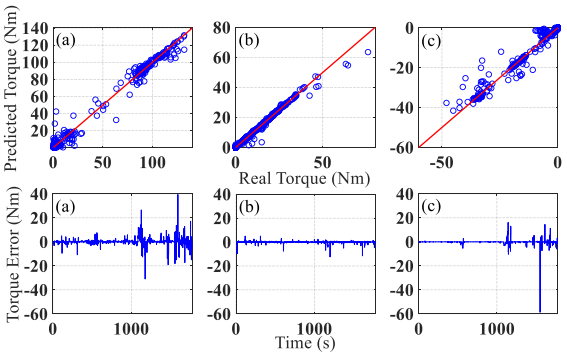


Fig. 14. The scatter plot of estimation results and estimation errors by VTC with LSSVM and RF. (a) ICE torque. (b) Motor Torque. (c) Generator Torque.

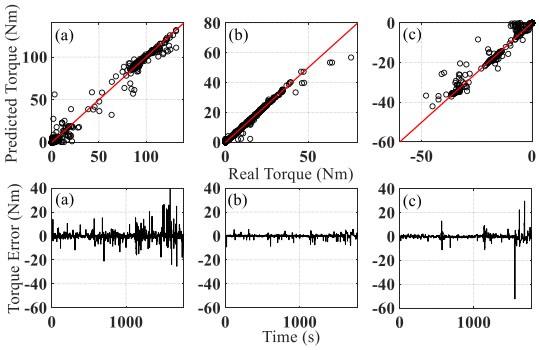


Fig. 15. The scatter plot of estimation results and estimation errors by VTC with Bi-LSTM. (a) ICE torque. (b) Motor Torque. (c) Generator Torque.

B. Model Validation in Simulation

After the advanced capability validation in regression, the proposed VTC is implemented to validate the built forward model. The verification and recalibration are mainly applied in the vehicle plant and control module in this study. Additionally, as the general responses of model to driving requirement, mode switch and energy management, the vehicle speed, component torque and speed, are the main indexes that can be applied to evaluate the effectiveness in model verification and recalibration. The vehicle's main parameters, efficiency maps of components and battery configuration are all acquired through the known specifications and the benchmark test, ensuring integrality and general accuracy of the vehicle plant module. The control module, involving the mode transition strategy and energy management rules, is initially developed based on the analysis of benchmark test data. During verification and recalibration, the mode transition logics and energy management rules are evaluated and returned by the boundary condition maps generated via

VTC, as shown in Figs. 16 to 18. The boundary condition maps for various mode switch are formulated with the discrete torques, required tractive powers, vehicle velocities and battery SOC based on the outputs of VTC when the NEDC, UDDS, US06, JC08, HWFET and WLTC driving cycles are tested. These test cycles cover most of the driving conditions, and can generate sufficient data for constructing the illustrated maps.

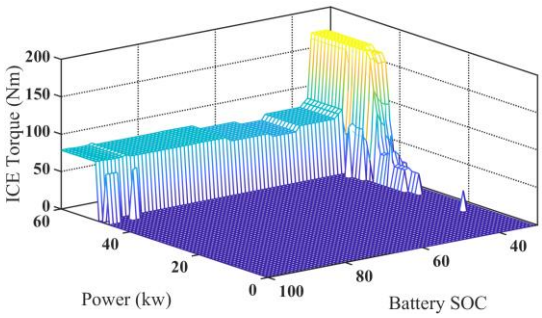


Fig. 16. Boundary map when the operation switches from EV to serial mode.

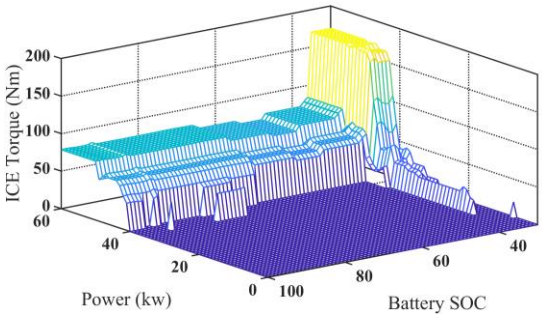


Fig. 17. Boundary map when the operation switches from serial to EV mode.

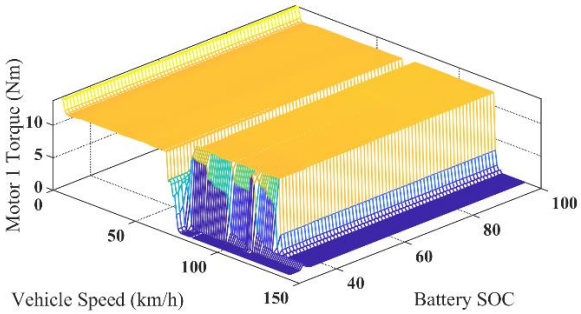


Fig. 18. The boundary condition map that operation mode switches from EV and serial to parallel mode.

During validation, the mode switch conditions of real vehicle can be explored according to the boundary condition maps. Table V lists the control thresholds of the initial vehicle model and those extracted according to the boundary maps. The thresholds in the initial vehicle model are recalibrated and updated to the new values of real vehicle if enough difference is detected. In the fixed operation mode, a series of ICE torques can be interpolated according to the required tractive power and battery SOC. As such, a group of torque combinations from ICE, generator, and motor 1 can be constituted, which are exploited to validate the controller module in the built forward model. Based on the detected difference between model and real vehicle, some critical parameters, e.g., battery power limits and optimal operation points of ICE and APU, can be recalibrated to improve the simulation accuracy of the built model.

TABLE V
SUMMARIZED CONTROL THRESHOLDS IN INITIAL MODEL AND FROM VTC

Operation Model	Control Threshold	VTC	Vehicle Model
CD-CS	Battery SOC	34%	32%
EV-Parallel	Velocity	120 km/h	120 km/h
Parallel-EV	Velocity	65 km/h	70 km/h
EV-Serial	Power	50 kW	55 kW
Serial-EV	Power	20 kW	17 kW

After recalibration, the accuracy of the constructed forward model is rationally addressed. Figs. 19 to 21 illustrate the evaluation results on how the proposed method can improve the control modelling. The critical role of the proposed method in strengthening control modelling is evaluated according to the behaviors of electric powertrain and energy consumption of the vehicle. As can be clearly observed, the model's general dynamic response and the outputted torque of ICE and generator can all closely match those generated through the benchmark test, validating the high accuracy of the built model. Fig. 19 shows the comparison result between the real velocity and simulation values outputted by the model. Based on the inputted driving cycle, the model can be executed to reproduce a reasonably accurate dynamic response. As can be seen from Figs. 20 and 21, the main observable difference between model and test data appears at the mode transition stage. As the forward model is a semi-static model aimed at describing the general performance of vehicle, it would be reasonable that the transient performance of vehicle plant, ICE, motors, generator and battery is difficult to capture.

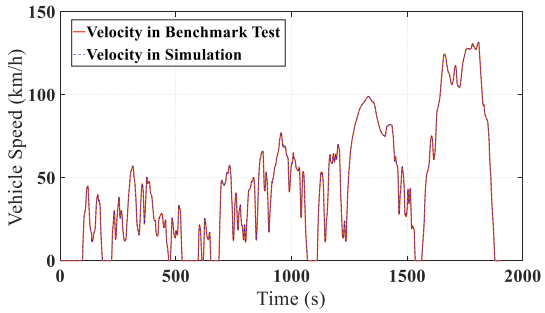


Fig. 19. Driving cycle for validation between simulation and benchmark test.

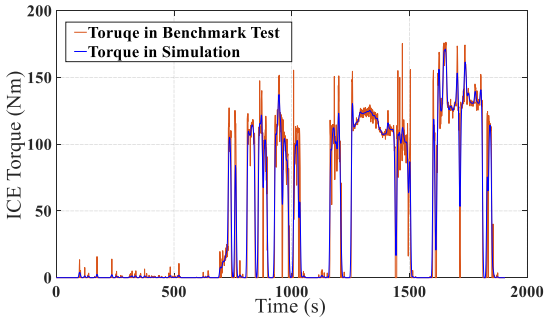


Fig. 20. ICE torque comparison between simulation and benchmark test.

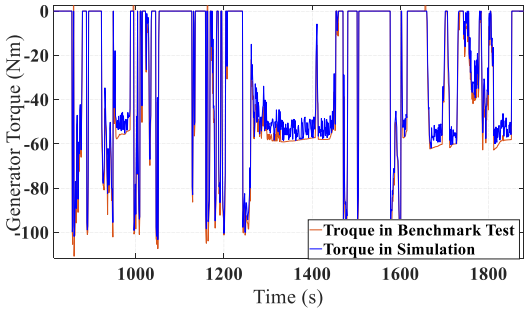


Fig. 21. Generator torque comparison between simulation and benchmark test.

Table VI demonstrates the comparison on fuel consumption obtained from real vehicle and model simulation. Generally, the validated control module in the built model can manage energy flow within the powertrain according to the manners that are quite close to the real vehicle. Note that the fuel consumption in Table VI denotes the total consumption after converting the electric energy consumption to the equivalent fuel amount [33]. The difference in fuel consumption between the real vehicle and validate model is only 1.1%. However, the accuracy of the non-validated model (6.8%) is much worse than that of the validated model. The operation mode percentage also reveals the difference in PHEV hybrid powertrain control systems. The constructed forward model presents close control results to the real vehicle with proper validations, and leads to cognate operation mode percentages. The panoramic analysis on electric powertrain behaviors and energy consumption in the real vehicle and the built model proves that the novel model validation method can prompt the accuracy of the control module in the constructed forward model, contributing to optimal design of vehicle control strategy.

TABLE VI
FUEL CONSUMPTION OBTAINED FROM REAL VEHICLE AND MODEL SIMULATION

Types	Fuel Consumption (L/100km)	Operation Mode Percentage (%)		
		EV Model	Serial Mode	Parallel Mode
Real Vehicle	4.51	56.2	27.3	16.5
Validated Model	4.56	55.7	26.2	18.1
Non-validated Model	4.84	53.1	22.6	24.3

C. Validation in Other PHEVs with Different Configurations

The applicability to various PHEVs with different configurations is a critical index to examine the performance of the novel model validation method. The applications in various PHEVs request proper modifications of control parameters listed in Table V. Table VII lists the application results based on a few PHEVs with different powertrain structures. In Table VII, real vehicle 1, 2 and 3 represent different powertrain architectures, i.e., serial [33], parallel [34] and power-split [35] configurations. According to the investigation results, the developed novel method can comprehensively validate the forward model with different configurations coherently. After validation, the difference in fuel consumption between real vehicles and forward models in three groups is only 1.6%, 1.4% and 1.9%, respectively.

In groups 1 and 3, the fuel consumption in forward models is lower than that in the real vehicle. This is because that the forward model neglects some transient dynamics that are difficult to capture in simulation. The operation percentage validates the performance of the novel model construction method from other perspectives. The similar operation mode percentages in EV, serial and parallel mode between the real vehicles and constructed forward models highlight that the constructed multi-configuration models achieve similar control effect as real vehicles, justifying the approaching control system behaviors.

TABLE VII
ASSESSMENT RESULTS OF NOVEL MODEL VALIDATION METHOD IN PHEVS WITH DIFFERENT CONFIGURATIONS

Groups	Types	Fuel Consumption (L/100km)	Operation Mode Percentage (%)		
			EV Model	Serial Mode	Parallel Mode
1	Real Vehicle 1	18.13	61.3	38.7	-
	Validated Model 1	17.83	62.5	37.5	-
2	Real Vehicle 2	3.46	75.4	-	24.6
	Validated Model 2	3.51	74.2	-	25.8
3	Real Vehicle 3	3.23	54.3	25.5	20.2
	Validated Model 3	3.17	56.1	26.4	17.5

D. HIL Test and Analysis

To further evaluate the role of the LSSVM and RF based VTC in model validation and examine the feasibility of the novel VTC in practical applications, HIL test is performed to provide the real-time environment, permitting to investigate the behaviors of the validated vehicle model in practice. The HIL test platform, as illustrated in Fig. 22, supports the test by cooperatively incorporating host PCs 1 and 2 and real-time controller. Before the HIL test, the vehicle plant module and controller module in the developed forward model are validated and recalibrated by the novel VTC. The controller module, including mode transition and energy management strategy, is compiled in host PC 1 and downloaded to the real-time dSpace based controller. The test scenario is supplied by the host PC 1, and the vehicle plant module is executed in host PC 2. The communication between the controller and host PC 2 is attained via CAN communication.

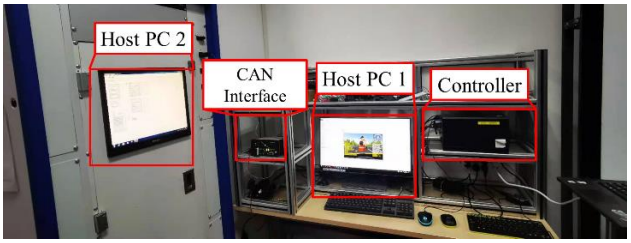


Fig. 22. Facilities for HIL test.

Figs. 23 and 24 illustrate the performance of the built vehicle plant model and the corresponding control strategy in real-time by comparing HIL test results with those collected from the benchmark test on a dynamometer. As can be

found, the HIL test results in terms of ICE and generator torques are generally consistent with those in benchmark test, and only some tiny differences appear. The reason of incurring difference is that the developed model validation and recalibration method cannot fully account for vehicle dynamics. In particular, the operation mode transition in the studied PHEV is realized through multiple intermediate sub-modes. For instance, the switch from serial mode to parallel mode goes across the clutch speed adjustment, clutch engagement and ICE ignition sub-modes, and the switch among these sub-modes is also governed by the specially designed rules. The model validation and recalibration based on the novel VTC, nevertheless, only focuses on macroscopic mode transition due to the limit of benchmark test, leading to the insufficient rule logic construction for sub-mode transition. As shown in Figs. 23 and 24, the difference mainly emerges in intensive acceleration and high driving speed area, where the mode frequently switches among EV, serial and parallel mode, presenting the ascertainable and limited difference. Table VIII lists the difference between HIL test and benchmark test in steady operation mode, and ignores mode transition comparison. Distinctly, the verified and recalibrated vehicle plant module and controller module can attain the promising effect in reproducing behaviors of real vehicle in steady state, indicating the feasibility of the proposed model validation method in practical situations.

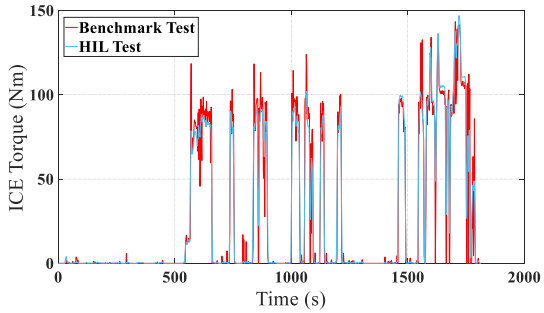


Fig. 23. ICE torque comparison between HIL test and benchmark test.

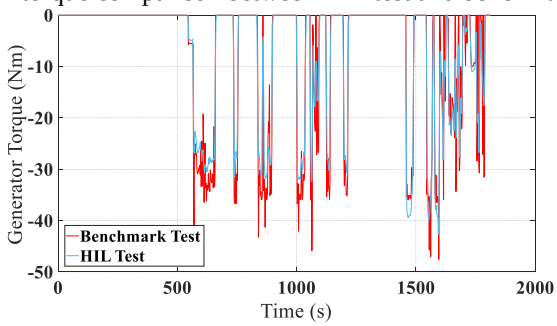


Fig. 24. Generator torque comparison in between HIL test and benchmark test.

TABLE VIII
STATISTICS OF VARIATION BETWEEN HIL TEST AND BENCHMARK TEST

Operation Mode	Average Difference (Nm)	Maximum Difference (Nm)	Accuracy (%)
EV	2.6	4.1	97.4
serial	2.9	4.7	96.9
parallel	2.7	3.6	96.5
Overall	3.1	4.2	96.4

E. Discussion

In this study, the built forward model for the studied 4WD PHEV is validated and recalibrated based the LSSVM and RF based VTC. To evaluate the promising performance of the raised method in mode construction and validation, a series of comparison studies are performed in both simulation and real-time environment. The investigations on VTC with different intelligent algorithms indicate that the LSSVM and RF based method can reproduce the behaviors of benchmark vehicle more accurate than other methods in instant applications. Even though the LSTM and Bi-LSTM raises the similar performance, the application scenarios manifest the LSSVM and RF algorithm is most appropriate to construct the VTC. By means of the novel VTC, the accuracy of the built forward model is refined via recalibration. The simulation study in Part B showcases the superior performance of the developed method in model validation, especially in validating the PHEV hybrid powertrain control system. The study on application capacities in different PHEVs justifies the robustness of the developed novel method in different scenarios. The HIL test further justifies the VTC application potential in model validation. The test results reveal that the accuracy of vehicle plant module and controller module in the furnished forward model is improved, contributing to the high-quality control algorithm design and development for practical application.

Despite the superior performance, the limitations of the proposed method still exist and can be summarized into the following aspects. Firstly, the supplied benchmark test method, test matrix and test equipment limit the data acquisition that describes the process of sub-mode transition, suppressing the performance promotion of VTC. More flexible benchmark test methods, comprehensive test matrix and advanced test equipment implementation need to be developed. Secondly, the inputs of RF in VTC are selected by the ReliefF algorithm, which selects the most relevant data labels with respect to the control strategy of the studied PHEV. However, feature selection by conventional machine learning techniques is a brutal process without the knowledge of physical property, imposing shallow perception on the whole data. Nonetheless, deep learning methods can automatically study the inner connection among different features and highlight preponderance in regression analysis, thus deserving to be further investigated.

V. CONCLUSION

In this paper, a forward 4-wheel drive PHEV model is constructed based on the benchmark test data. A novel method based on the ReliefF, LSSVM and RF algorithm is proposed and demonstrated for the validation of the VTC model, and achieves significant improvements in overall accuracy when compared with other traditional methods. In the simulation based evaluation, the virtual test controller constructed by the proposed method achieves the minimum RMSEs and

MAEs in estimating instant component performance, compared with other benchmark methods. Meanwhile, the verified and recalibrated vehicle model by the novel VTC also presents remarkable accuracy. In the HIL test, the applicable capacity in practical situation by the raised model validation method is carefully investigated. In addition, different PHEVs are examined to showcase the wide adaptivity of the proposed VTC modelling manner to different powertrain architectures. The validated vehicle control module and plant module integrally achieves up to 97.4% accuracy in the HIL test. The results indicate that the model validation method is efficient in improving model accuracy and is verified valuable for optimal control strategy design.

For the future work related to model validation, more attention will be paid to seek methods to validate and recalibrate dynamic models, rather than merely static models. In addition, the deep learning method will be tailored for model validation to pursue better modeling performance.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (No. 61763021 and 51775063), and in part by the EU-funded Marie Skłodowska-Curie Individual Fellowships Project (No. 845102).

REFERENCES

[1] S. Cheng, L. Li, X. Chen, S. Fang, X. Wang, X, Xu, and W. Li, "Longitudinal autonomous driving based on game theory for intelligent hybrid electric vehicles with connectivity." Applied Energy 268 (2020): 115030.
[2] C. Yang, M. Zha, W. Wang, K, Liu, and C. Xiang, "Efficient energy management strategy for hybrid electric vehicles/plug-in hybrid electric vehicles: review and recent advances under intelligent transportation system." IET Intelligent Transport Systems 14.7 (2020): 702-711.
[3] Kim, Sooyoung, and Seibum Ben Choi. "Cooperative Control of Drive Motor and Clutch for Gear Shift of Hybrid Electric Vehicles with Dual-clutch Transmission." IEEE/ASME Transactions on Mechatronics (2020).
[4] C. Yang, Y. Shi, L. Li, and C. Xiang, "Efficient mode transition control for parallel hybrid electric vehicle with adaptive dual-loop control framework." IEEE Transactions on Vehicular Technology 69.2 (2019): 1519-1532..
[5] C. Yang, S. You, W. Wang, L. Li, and C. Xiang,"A stochastic predictive energy management strategy for plug-in hybrid electric vehicles based on fast rolling optimization." IEEE Transactions on Industrial Electronics 67.11 (2019): 9659-9670.
[6] F. Al-Turjrnan, "5G-enabled devices and smart-spaces in social-IoT: An overview," Future Generation Computer Systems-the International Journal of Escience, vol. 92, pp. 732-744, Mar 2019.
[7] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep Reinforcement Learning Based Resource Allocation for V2V Communications," IEEE Transactions on Vehicular Technology, vol. 68, no. 4, pp. 3163-3173, Apr 2019.
[8] Z. Gong, F. Jiang, and C. Li, "Angle Domain Channel Tracking With Large Antenna Array for High Mobility V2I Millimeter Wave Communications," IEEE Journal of Selected Topics in Signal Processing, vol. 13, no. 5, pp. 1077-1089, Sep 2019.
[9] W. Zhou, L. Yang, Y. Cai, and T. Ying, "Dynamic programming for New Energy Vehicles based on their work modes part I: Electric Vehicles and Hybrid Electric Vehicles," Journal of Power Sources, vol. 406, pp. 151-166, Dec 1 2018.
[10] 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 Transactions on Vehicular Technology, vol. 66, no. 6, pp. 4534-4549, Jun 2017.
[11] G. Rizzoni, L. Guzzella, and B. M. Baumann, "Unified modeling of hybrid electric vehicle drivetrains," IEEE/ASME Transactions on Mechatronics, vol. 4, no. 3, pp. 246-257, Sep 1999.
[12] C. C. Lin, H. Peng, J. W. Grizzle, and J. M. Kang, "Power management strategy for a parallel hybrid electric truck," IEEE Transactions on Control Systems Technology, vol. 11, no. 6, pp. 839-849, Nov 2003.
[13] T. Liu, X. Tang, H. Wang, H. Yu, and X. Hu, "Adaptive Hierarchical Energy Management Design for a Plug-In Hybrid Electric Vehicle," IEEE Transactions on Vehicular Technology, vol. 68, no. 12, pp. 11513-11522, Dec 2019.
[14] T. Liu, X. Hu, S. E. Li, and D. Cao, "Reinforcement Learning Optimized Look-Ahead Energy Management of a Parallel Hybrid Electric Vehicle," IEEE/ASME Transactions on Mechatronics, vol. 22, no. 4, pp. 1497-1507, Aug 2017.
[15] J. Guo, H. He, J. Peng, and N. Zhou, "A novel MPC-based adaptive energy management strategy in plug-in hybrid electric vehicles," Energy, vol. 175, pp. 378-392, May 15 2019.

[16] C. Sun, X. Hu, S. J. Moura, and F. Sun, "Velocity Predictors for Predictive Energy Management in Hybrid Electric Vehicles," IEEE Transactions on Control Systems Technology, vol. 23, no. 3, pp. 1197-1204, May 2015.
[17] S. Xie, X. Hu, S. Qi, and K. Lang, "An artificial neural network-enhanced energy management strategy for plug-in hybrid electric vehicles," Energy, vol. 163, pp. 837-848, Nov 15 2018.
[18] Y. Li, H. He, J. Peng, and H. Wang, "Deep Reinforcement Learning-Based Energy Management for a Series Hybrid Electric Vehicle Enabled by History Cumulative Trip Information," IEEE Transactions on Vehicular Technology, vol. 68, no. 8, pp. 7416-7430, Aug 2019.
[19] M. Duoba, R. B. Carlson, and J. Wu, "Test procedure development for "Blended Type" plug-in hybrid vehicles," SAE International Journal of Engines, Article vol. 1, no. 1, pp. 359-371, 2009.
[20] M. D. Eric Rask, Henning Lohse-Busch, Patrick Walsh, "Advanced Technology Vehicle Lab Benchmarking-Level 2 (in-depth)," Argonne National Laboratory: Lemont, IL, USA(2012).
[21] Shu, Xing, et al. "Stage of Charge Estimation of Lithium-ion Battery Packs Based on Improved Cubature Kalman Filter with Long Short-Term Memory Model." IEEE Transactions on Transportation Electrification (2020).
[22] Z. Huang, C. Yang, X. Zhou, and T. Huang, "A Hybrid Feature Selection Method Based on Binary State Transition Algorithm and ReliefF," IEEE Journal of Biomedical and Health Informatics, vol. 23, no. 5, pp. 1888-1898, Sep 2019.
[23] A. Zafra, M. Pechenizkiy, and S. Ventura, "ReliefF-MI: An extension of ReliefF to multiple instance learning," Neurocomputing, vol. 75, no. 1, pp. 210-218, Jan 1 2012.
[24] M. Robnik-Sikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," Machine Learning, vol. 53, no. 1-2, pp. 23-69, Oct-Nov 2003.
[25] H. F. Wang and D. J. Hu, Comparison of SVM and LS-SVM for regression (Proceedings of the 2005 International Conference on Neural Networks and Brain, Vols 1-3). 2005, pp. 279-283.
[26] H. Cao, S. Bernard, R. Sabourin, and L. Heutte, "Random forest dissimilarity based multi-view learning for Radiomics application," Pattern Recognition, vol. 88, pp. 185-197, Apr 2019.
[27] A. Liaw and M. J. R. N. Wiener, "Classification and Regression by randomForest," vol. 23, no. 23, 2002.
[28] G. Biau and E. Scornet, "A random forest guided tour," Test, vol. 25, no. 2, pp. 197-227, Jun 2016.
[29] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," Neural Computation, vol. 12, no. 10, pp. 2451-2471, Oct 2000.
[30] D. Fan, J. Yang, J. Zhang, Z. Lv, H. Huang, J. Qi, and P. Yang, "Effectively Measuring Respiratory Flow With Portable Pressure Data Using Back Propagation Neural Network," IEEE Journal of Translational Engineering in Health and Medicine-Jtehm, vol. 6, 2018 2018, Art. no. 1600112.
[31] Kiperwasser, Eliyahu, and Yoav Goldberg. "Simple and accurate dependency parsing using bidirectional LSTM feature representations." Transactions of the Association for Computational Linguistics 4 (2016): 313-327.
[32] Ljung, Lennart. "System identification toolbox." The Matlab user's guide (1988).
[33] Zhang, Y., Murtagh, M., Early, J., Cunningham, G., Steele, D., and Best, R. "Optimal Control Strategy for the Next Generation Range Extended Electric Bus." SAE Technical Paper, (2020): 2020-01-0844.
[34] Y. Zhang, L. Chu, Y. Ou, C. Guo, Y. Liu, and X. Tang, "A cyber-physical system-based velocity-profile prediction method and case study of application in plug-in hybrid electric vehicle." IEEE transactions on cybernetics 51.1 (2019): 40-51.
[35] Z. Chen, H. Hu, Y, Wu, Y. Zhang, G. Li, and Y. Liu, "Stochastic model predictive control for energy management of power-split plug-in hybrid electric vehicles based on reinforcement learning." Energy 211 (2020): 118931.