Data-Driven Modelling of a Plug-In Hybrid Electric Vehicle

Adnan Furkan Yildiz[1], Rahman Sahinler[1], Cansu Ozturk[1]

[1]AVL Research and Engineering Turkey, Istanbul, Turkey
furkan.yildiz@avl.com, rahman.sahinler@avl.com, cansu.ozturk@avl.com

Abstract

Growing need for environmentally friendly transportation has led to the rise of hybrid electric vehicles (HEVs). Plug-in hybrid electric vehicles (PHEVs), which are a sub-class of HEVs, provide lower fuel consumption and exhaust gas emissions thanks to their external charging and higher range of electric drive capabilities. Therefore, they are increasingly popular in the automotive industry nowadays. To make a rapid assessment of the control or the energy management strategies applied to a PHEV, the generation of a vehicle model has a crucial role. This paper proposes a rapid-responsive, data-driven, kinematic, and map-based PHEV model in a Python environment. Approximately 2000 km of real-world vehicle measurements were used instead of driving cycles, which enables a wide range of driving maneuvers to be used for modeling and testing purposes. Results of the vehicle model were compared in two categories to express the charge-sustaining (CS) and charge-depleting (CD) behaviors of the model.

1. Introduction

Hybrid electric vehicles (HEVs) consisting of two energy converters, generally an internal combustion engine (ICE) as a primary energy converter and one or more electric motor (EM) as secondary energy converters, are nowadays one of the most promising solutions for not only environmental pollution but also fuel economy [1]. Compared to HEVs, plug-in hybrid electric vehicles (PHEVs) provide lower fuel consumption and exhaust gas emission thanks to their external charging and higher range of electric drive capabilities [2]. Although there are several powertrain topologies of PHEVs in the industry, parallel type-2 (P2) architecture is one of the most used architectures due to its increased energy recuperation potential and the availability of additional hybrid control functions. Therefore, a P2-type PHEV is used for modeling purposes in our study.

P2 architecture permits separate or combined traction from ICE and EM based on the driver's request. Thus, torque distribution between ICE and EM is a substantial topic and is mostly studied under the title of energy management. Energy management strategies in the literature are mainly classified into two categories: rule-based and optimization-based methods. Rule-based approaches are classified into two sub-categories: deterministic and fuzzy. Optimization-based methods, on the other hand, are divided into global, real-time, and local strategies [3]. Global optimization-based methods such as dynamic programming (DP) or the Pontryagin minimization principle (PMP) cannot be used in real-time or fast-response required applications due to their computational complexity. However, rule-based methods as well as real-time and local optimization-based methods namely, equivalent consumption minimization strategy (ECMS), adaptive-ECMS (A-ECMS), robust control (RC), and artificial intelligence (AI) based energy management strategies are useful for real-time or fast-response required applications [4].

Apart from vehicle models, energy management strategies are also developed for various types of vehicles. For instance, an energy management strategy is developed for a fuel cell electric vehicle using model-based reinforcement learning with data-driven model updates in MATLAB [5]. In the study, they used different drive cycles which are Urban Dynamometer Driving Schedule (UDDS), New European Driving Cycle (NEDC), Supplemental Federal Test Procedure (SC03), WLTC, and HWFET. 5.7% fuel economy improvement is obtained on average. A reinforcement learning-based algorithm using ECMS is proposed for a fuel cell hybrid electric vehicle (FCHEV) [6]. The model is proposed in MATLAB environment and up to 26% fuel consumption improvement is obtained for various drive cycles such as HWFET, WVUCITY, and West Virginia University Suburban Cycle (WVUSUB). A naturalistic data-driven predictive energy management is implemented for PHEVs [7]. Extreme learning machine is used as a short-term speed predictor in the paper. The proposed method is compared with back propagation neural networks and support vector machine predictors. Training is done using seven standard drive cycles such as UDDS, HWFET, etc.

There are several studies in the literature mentioning hybrid vehicle modeling and mostly MATLAB/Simulink environment is used as the simulation platform. For example, Robbio et al. performed a hybrid electric commercial vehicle simulation for fuel consumption and emission evaluation [8]. For the simulations, Advanced Vehicle Simulator (ADVISOR) software was used in the MATLAB environment. The tests are carried out with Worldwide harmonized Light-duty Vehicle Test Cycles (WLTC) in CD and CS modes with a success of approximately 1% deviation. ADVISOR software is used for the modeling of a Sedan vehicle with both parallel and series hybrid electric vehicle topologies [9]. A drive cycle called West Virginia University City Cycle (WVUCITY) is tested with the model. ADAMS and MATLAB/Simulink co-simulation platform is created for a hybrid vehicle model implementation [10], which are used for control logic and modeling of the mechanical inertial components, respectively. Fuel economy comparison is made between hybrid and conventional vehicles using EPA New York City Cycle (NYCC) and Highway Fuel Economy Cycle (HWFET) with an improvement of 8.9% and 14.3%, respectively. A Python-based vehicle model is applied for an electric vehicle (EV) Nissan Leaf using object-oriented programming approach [11]. The model is done using the drive cycles which are Federal Test Procedure (FTP)-75 and NEDC and a 70 km real-world drive is used for verification of the model.

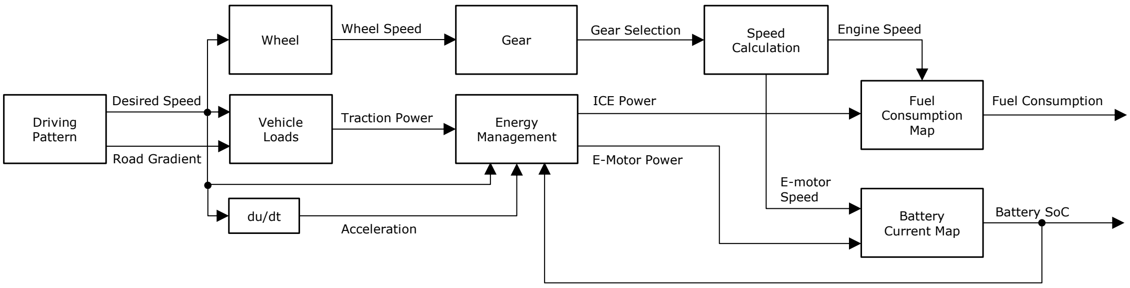

Fig. 1. Block diagram of the vehicle model

Our proposed data-driven vehicle model aims to create representativeness of a PHEV in various real-world driving conditions. To that attempt, we used a kinematic approach where the input variables are the speed of the vehicle in km/h and the grade angle of the road in %. Unlike most of the studies mentioned above, our vehicle model was developed within a Python environment using object-oriented programming. We chose to use the Python environment for two main reasons: its object-oriented capabilities provide a rapid response, and it offers powerful frameworks for machine learning projects that will be the basis of our future work. In particular, our model can evaluate 2000 seconds of vehicle data in just 3-5 seconds. This efficiency surpasses many models in the field, including rule-based approaches as given by [2].

Real-world maneuvers with urban, rural, and highway characteristics were used to create a map-based vehicle model. Instead of constructing a vehicle model that only relies on standard driving cycles, our aim is to make a vehicle model that is built on real-world data and that can represent the state of charge (SoC) and fuel consumption behaviors for most of the conditions such as CD and CS behaviors with acceptable accuracy. This approach leads us to the primary contributions of our work. The creation of a fast-responding vehicle model within the Python environment which can be used in future machine-learning applications. Also, a vehicle model design can be adapted easily for different hybrid vehicles with various powertrain architectures when the wide-ranging operational data of the vehicle is available.

The rest of the paper is organized as follows. In section 2, the utilized data for the modeling and verification is mentioned. Section 3 explains the vehicle modeling approach. In section 4, energy management strategy is presented. Section 5 modeling results are shown. The conclusions are presented in section 6.

2. Input Data

The reference vehicle used for modeling purposes was a plug-in hybrid electric vehicle with a P2 topology and an ICE with a 2-liter engine displacement.

The vehicle measurements driven in various driving conditions were used for modeling purposes. During the trips, the channels were recorded via Controller Area Network (CAN) bus with a 100 ms sampling time. In total, the amount of driven distance of the vehicle is 1908 km which consists of 40.8% urban (0-60 km/h), 16.8% rural (60-90 km/h), and 42.8% highway (> 90 km/h) distributions. While the environmental temperature in the measurements ranges between 6°C and 32°C, the environmental pressure ranges from 850 mbar to 1 atm.

3. Vehicle Modelling Approach

In the paper, a data-oriented vehicle model approach is presented. Our main purpose is to obtain a fast-responding vehicle model which can be used in Python environment. Since the vehicle model will be used in reinforcement learning-based applications in future applications, the duration of the result generation of the model is desired in seconds. Using mathematical modeling instead of physical modeling of each component provides more rapid responses. Therefore, mathematical equations and data-driven map generation techniques are combined to get an accurate vehicle model in Python.

A block diagram of the vehicle model is demonstrated in Fig. 1. There are two inputs of the vehicle model which are vehicle speed in km/h and road gradient in %. The inputs are used with vehicle road load coefficients to calculate traction power. In addition, the wheel speed was simply calculated from the vehicle speed and tire diameter. While SoC at the previous step and vehicle speed as well as acceleration values at the current step are used in the energy management strategy, the wheel speed is used in the gear selection algorithm to obtain the speed of ICE and EM. Then, the power distributions of ICE and EM are obtained using the generated maps from the data. Finally, while ICE power and speed are used for fuel consumption calculation using a fuel consumption map, the EM power and speed are used for the calculation of SoC at the next step using a battery current map and coulomb counting approach.

Traction power is calculated based on the equation of motion such as given in (1), where V is vehicle speed in km/h, F_0, F_1, F_2 are road load coefficients in N, N/(km/h), N/(km/h)^2, respectively, m is vehicle mass in kg, α is road gradient in degrees, w_i is inertia factor, P is the traction power in kW, g is the gravity and a is acceleration in m/s^2. Equation (1) provides a simplified version of the motion formula, enabling a rapid and effective solution for our approach to vehicle modeling.

$$P = V\,(F_0 + F_1.V + F_2.V^2 + m.g.\sin\alpha + m.a.w_i) \qquad (1)$$

4. Energy Management Strategy

The most critical part of the vehicle model is energy management between ICE and EM. In our model, this step is divided into two sub-steps. The first step is defining hybrid and electric mode selection and the second step is ICE and EM power calculation which will further be used as the inputs for fuel consumption and remaining SoC calculations.

4.1. Mode Selection Strategy

For a mode selection between electric and hybrid drive, a stochastic approach was implemented in the vehicle model. The residency of the hybrid and electric mode selection can be seen in Fig. 2. While ICE and EM operate simultaneously in hybrid mode as three, only the EM operates in electric mode which is expressed as two. The numbers between two and three are the average values of the mode when the vehicle operates in the specified vehicle speed and acceleration region. The average value increases towards three when the vehicle speed and acceleration increase. The frequency of the vehicle driven in hybrid mode increases when the vehicle travels in rural and highway maneuvers, which is an expectation that the efficiency of the ICE is higher in moderate engine speed and torque areas. In our stochastic approach, this observation was used as the baseline. The whole dataset is grouped with respect to defined SoC regions from 0% to 100% with 5% incremental steps. For each group, the residency of the mode channel is obtained and then the values between two and three are mapped between zero and one to get the probability matrix. Finally, the map distribution is adjusted as monotonically increasing. The probability matrices for two sample SoC regions are shown in Fig. 3. The closer the values are to zero, the more frequent the activation of the electric mode. In the first example, the SoC region is selected between 25% and 30%. Since the SoC value is closer to the lower limit, the more frequent hybrid mode activation especially in rural and highway driving regions is seen. Contrary to the first example, the battery and therefore the EM can be used more frequently in this case. That is why, the map generally expresses zero values representing mostly electric mode activation during energy management. Just in rural and highway areas, ICE activation can also be seen when the vehicle travels steadily or with positive acceleration.

4.2. Power Calculation

As for the power calculation, three maps were generated and used to define the necessary power output from EM and ICE in the relevant driving conditions. All axes of the maps are the same and while the x-axis represents the vehicle acceleration axis represents the traction power with a range between ±400 kW with 25 kW increment and the z-axis represents the ICE power output in hybrid mode, EM power output in hybrid mode and EM power output in electric mode in kW.

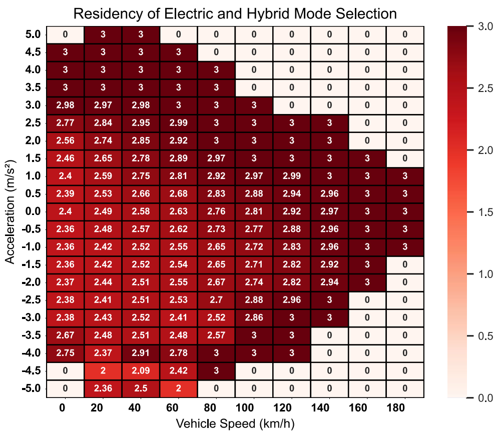

Fig. 2. Residency of the mode selection in the complete data

According to the selected mode in the previous step, the necessary power outputs will be obtained. Finally, while fuel consumption is calculated using the fuel consumption map of the vehicle, the remaining SoC is calculated using the battery current map and the coulomb counting approach.

5. Results

The vehicle model was implemented with the data from various driving conditions. Since our modeling methodology is based on map definitions through the data analytics of the general vehicle behavior and not specifically done for some selected measurements, the complete dataset with a travel distance of 1908 km is used in this section instead of dividing the measurements into modeling and testing groups. In Table 1, the characteristics of the driving conditions for urban, rural, and highway profiles are shown. While the average vehicle speeds of CS measurements are 27.2, 75.7, and 117.3 km/h, the average vehicle speeds of CD measurements are 27.9, 75.4, and 98.2 km/h for urban, rural, and highway profiles respectively.

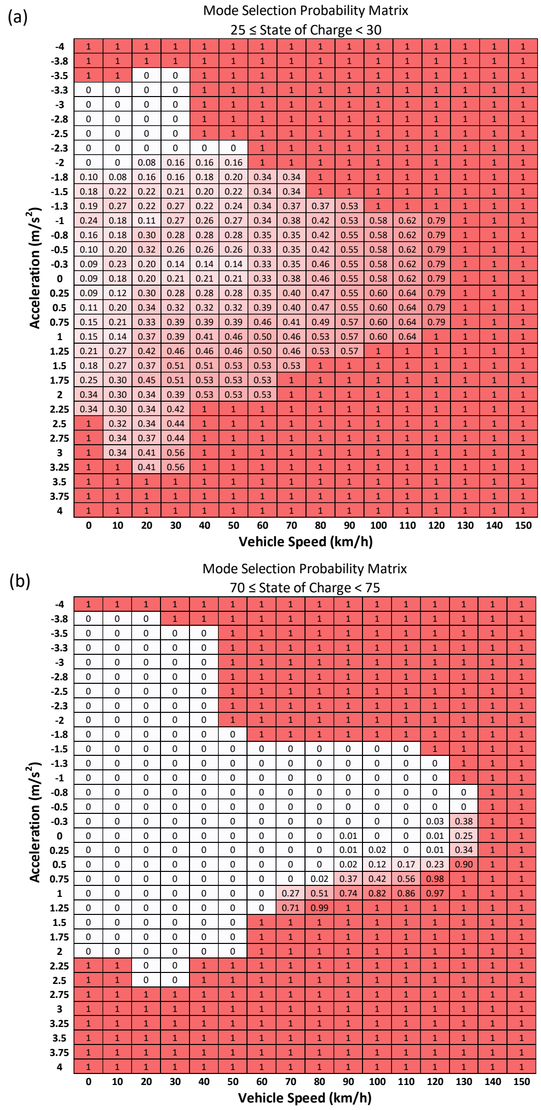

Fig. 3. Mode selection probability matrices for two sample SoC regions, (a) SoC range between 25% and 30%, (b) SoC range between 70% and 75

Table 1. Driving conditions of maneuvers

		CS Measurements		CD Measurements	
		Travel Distance (km)	Average Vehicle Speed (km/h)	Travel Distance (km)	Average Vehicle Speed (km/h)
Driving Profile	Urban	422.6	27.2	348.6	27.9
	Rural	179.1	75.7	141.5	75.4
	Highway	557.0	117.3	258.9	98.2

5.1. CS Mode Behavior

Overall, 1159 km of vehicle data is used for CS mode behavior analysis. As mentioned in the energy management strategy section, the mode selection of the vehicle depends on the output of the probabilistic approach. Therefore, three distinct seed values were used to run the vehicle model with the input signals to take the randomness into account during the testing of the model.

The actual average fuel consumption as well as the model results with different seed values are expressed in Table 2. The actual average fuel consumption of the vehicle can be seen as 7.35 L/100km. On the other hand, the average fuel consumption of the three model results is 6.75 L/100km. The deviation between the model results and the actual fuel consumption is shown as 8.2%.

As for the SoC validation of the model, the average deviation between the actual SoC signal and the modeled SoC signal was considered. For each model run, the average deviation was calculated as 1.83%, 1.96%, and 2.40% respectively. The mean value of the deviations is 2.06%. In addition to the tabular representation, a time-based diagram can be seen for the CS result comparison in Fig. 4. For the time-based representation of the comparison, one of the longest traveled measurements with an average success of the vehicle model on fuel consumption and SoC modeling is selected. In the sample measurement, the SoC of the battery is between 10% and 16% for the model and actual result.

Electric and hybrid mode distributions as well as fuel consumption at the end of the test can be seen in Table 3. It was observed that while the hybrid mode distributions form the majority of the energy management output, the electric mode distributions represent the minor part of the mode selection output for both results because of the CS behavior of the test.

The percentage deviation between the mode distributions is 48.9% and 11.1% for electric and hybrid modes respectively. Since the amount of electric distribution is around 10% to 15% for both results, the percentage deviation is higher when compared to the hybrid mode distribution.

Table 2. Fuel consumption comparison using CS maneuvers

	Average Fuel Consumption (L/100km)
Actual Result	7.35
Model Result #1	6.58
Model Result #2	6.43
Model Result #3	7.24
Average of 3 Model Result	6.75
Deviation (%)	8.2%

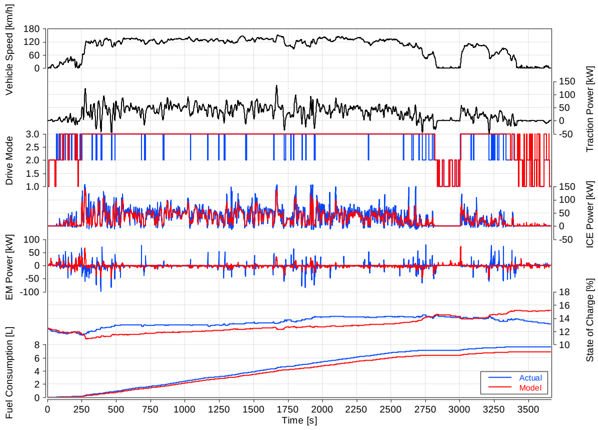

Fig. 4. Model performance comparison on a sample CS measurement

However, the deviation does not affect the fuel consumption or SoC trace that much as can be seen from Fig. 4.

As for the fuel consumption deviation, the deviation between the model output and the actual value is 9.2%. On the other hand, the average and maximum absolute SoC deviations are 0.93% and 2.04%, which can also be seen from Fig. 4.

5.2. CD Mode Behavior

To observe the CD mode behavior of the model, it was tested with 749 km of vehicle data. Like it is mentioned in the CS mode behavior section, the model run is repeated three times to see the effect of the randomness. In Table 4, the fuel consumption results can be seen which are 4.58 L/100km and 4.50 L/km for actual and average model results. The average deviation can also be seen as 9.83%. Here, it must be noted that it does not mean the vehicle always travels in electric mode by mentioning CD measurements. Instead, the electric mode frequency is higher in these measurements than in the CS measurements. Therefore, the fuel consumption is lower in these results but not equal to zero. For the SoC comparison, the average deviation between the modeled SoC and the actual values was considered.

Table 3. Model and actual results on the sample CS measurement

	Actual	Model	Error (%)
Electric Mode Distribution (%)	17.6	9.0	- 48.9
Hybrid Mode Distribution (%)	76.6	85.1	11.1
Fuel Consumption (L/100km)	7.83	7.11	- 9.2

Table 4. Fuel consumption comparison using CD maneuvers

	Average Fuel Consumption (L/100km)
Actual Result	4.58
Model Result #1	4.13
Model Result #2	4.24
Model Result #3	5.14
Average of 3 Model Result	4.50
Deviation (%)	9.83%

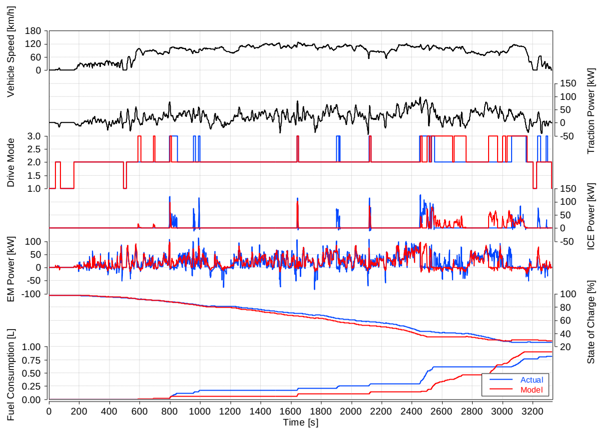

Fig. 5. Model performance comparison on a sample CD measurement

For each model run the average deviations are 4.34%, 4.39%, and 4.30% where the mean deviation is 4.34%.

Like in the previous section, the model and the actual test result comparison is demonstrated for one of the longest sample CD measurements which is expressed in Fig. 5. SoC starts at 100% at the beginning of the test and decreases around 25% for model and actual result.

Electric and hybrid mode distributions as well as fuel consumption at the end of the test can be seen in Table 5. Contrary to the CS measurement, the electric mode activation in the test is much higher than expected. The percentage deviation between the mode distributions can be seen as 7.1% and 58.4% for electric and hybrid modes respectively. Since the amount of hybrid distribution is around 10% to 15% for both results, the percentage deviation is higher, which does not affect the fuel consumption or SoC trace at this rate. As for the fuel consumption and the SoC traces, the fuel consumption of the vehicle model is 11% higher than the actual fuel consumption. Besides, the mean and maximum SoC deviations are 2.93% and 7.98% respectively. Since the measurement characteristic is more effective on the SoC of the battery, the mean and maximum deviations are higher when compared to the values in the CS test results.

6. Conclusions

In this study, generating a fast-responding vehicle model in Python environment was successfully developed. For that purpose, an object-oriented programming structure with a data-driven and map-based modeling approach was combined with data analytics methods for the interpretation of real-world data. According to the results, the vehicle model can evaluate up to 7000 seconds of test data within 10-15 seconds. Additionally, the model can represent fuel consumption and SoC behaviors in urban, rural, and highway driving conditions. Although the average fuel consumption and SoC deviations, which are around 10% and 5% respectively, seem at higher levels when compared to the studies in the literature, the main reason and the advantage of this study is providing a huge amount of real-world maneuver validation of the model. As for future work, the vehicle model will be used in artificial intelligence-based predictive energy management studies.

Table 5. Model and actual results on the sample CD measurement

	Actual	Model	Error (%)
Electric Mode Distribution (%)	84.4	78.4	- 7.1
Hybrid Mode Distribution (%)	10.1	16.0	58.4
Fuel Consumption (L/100km)	1.09	1.21	11.0

7. References

[1] S. Onori, L. Serrao and G. Rizzoni, "Hybrid electric vehicles: Energy management strategies", Springer, London, England, 2016.

[2] R. Cheng and Z. Dong, "Modeling and simulation of plug-in hybrid electric powertrain system for different vehicular application", in *2015 IEEE Vehicle Power and Propulsion Conference (VPPC)*, Montreal, QC, Canada, 2015, pp. 1-7.

[3] C. Liu and Y. Liu, "Energy management strategy for plug-in hybrid electric vehicles based on driving condition recognition: A Review", *Electronics*, vol. 11, no. 3, pp. 342, Jan. 2022.

[4] F. Zhang, L. Wang, S. Coskun, H. Pang, Y. Cui, and J. Xi, "Energy management strategies for hybrid electric vehicles: review, classification, comparison, and outlook", *Energies*, vol. 13, no. 13, pp. 3352, Jun. 2020.

[5] H. Lee and S. W. Cha, "Energy Management Strategy of Fuel Cell Electric Vehicles Using Model-Based Reinforcement Learning With Data-Driven Model Update", *IEEE Access*, vol. 9, pp. 59244-59254, 2021.

[6] H. Sun, Z. Fu, F. Tao, L. Zhu, and P. Si, "Data-driven reinforcement learning based hierarchical energy management strategy for fuel cell/battery/ultracapacitor hybrid electric vehicles", *Journal of Power Sources*, vol. 455, 2020.

[7] X. Tang, T. Jia, X. Hu, Y. Huang, Z. Deng, and H. Pu, "Naturalistic data-driven predictive energy management for plug-in hybrid electric vehicles", *IEEE Transactions on Transportation Electrification*, vol. 7, no. 2, pp. 497-508, Jun. 2021.

[8] R. D. Robbio, M. C. Cameretti, and E. Mancaruso, "Investigation by modeling of a plug-in hybrid electric commercial vehicle with diesel engine on WLTC", *Fuel*, vol. 317, Jun. 2022.

[9] A. Saleki, S. Rezazade, and M. Changizian, "Analysis and simulation of hybrid electric vehicles for sedan vehicle", in *2017 Iranian Conference on Electrical Engineering (ICEE)*, Tehran, Iran, 2017, pp. 1412-1416.

[10] B. Fan, "Modeling and Simulation of A Hybrid Electric Vehicle Using MATLAB/Simulink and ADAMS", M.S. thesis, Mech. Mechatron. Eng., UW, Ontario, Canada, 2007.

[11] S. Howroyd and R. Thring, "An electric vehicle model and validation using a Nissan Leaf: A Python-based object-oriented programming approach", *Advances in Mechanical Engineering*, vol. 10, no. 7, pp. 1-7, Jan. 2018.