applied system innovation

MDPI

Article

Soft Sensors for State of Charge, State of Energy, and Power Loss in Formula Student Electric Vehicle

Kanishkavikram Purohit [1], Shivangi Srivastava [1], Varun Nookala [1], Vivek Joshi [1], Pritesh Shah [1,*], Ravi Sekhar [1], Satyam Panchal [2], Michael Fowler [3], Roydon Fraser [2], Manh-Kien Tran [3] and Chris Shum [4]

[1] Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune 412115, India; purohit.kanishkavikram@sitpune.edu.in (K.P.); shivangi.srivastava@sitpune.edu.in (S.S.); nookala.varun@sitpune.edu.in (V.N.); vivek.joshi@sitpune.edu.in (V.J.); ravi.sekhar@sitpune.edu.in (R.S.)
[2] Department of Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada; satyam.panchal@uwaterloo.ca (S.P.); rafraser@uwaterloo.ca (R.F.)
[3] Department of Chemical Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada; mfowler@uwaterloo.ca (M.F.); kmtran@uwaterloo.ca (M.-K.T.)
[4] Motion Research Group, Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada; c2shum@uwaterloo.ca
* Correspondence: pritesh.shah@sitpune.edu.in; Tel.: +91-20-6193-6499

check for updates

Citation: Purohit, K.; Srivastava, S.; Nookala, V.; Joshi, V.; Shah, P.; Sekhar, R.; Panchal, S.; Fowler, M.; Fraser, R.; Tran, M.-K.; et al. Soft Sensors for State of Charge, State of Energy, and Power Loss in Formula Student Electric Vehicle. *Appl. Syst. Innov.* **2021**, *4*, 78. https://doi.org/10.3390/asi4040078

Academic Editor: Andrey Chernov

Received: 23 August 2021
Accepted: 6 October 2021
Published: 13 October 2021

Abstract: The proliferation of electric vehicle (EV) technology is an important step towards a more sustainable future. In the current work, two-layer feed-forward artificial neural-network-based machine learning is applied to design soft sensors to estimate the state of charge (SOC), state of energy (SOE), and power loss (PL) of a formula student electric vehicle (FSEV) battery-pack system. The proposed soft sensors were designed to predict the SOC, SOE, and PL of the EV battery pack on the basis of the input current profile. The input current profile was derived on the basis of the designed vehicle parameters, and formula Bharat track features and guidelines. All developed soft sensors were tested for mean squared error (MSE) and R-squared metrics of the dataset partitions; equations relating the derived and predicted outputs; error histograms of the training, validation, and testing datasets; training state indicators such as gradient, mu, and validation fails; validation performance over successive epochs; and predicted versus derived plots over one lap time. Moreover, the prediction accuracy of the proposed soft sensors was compared against linear or nonlinear regression models and parametric structure models used for system identification such as autoregressive with exogenous variables (ARX), autoregressive moving average with exogenous variables (ARMAX), output error (OE) and Box Jenkins (BJ). The testing dataset accuracy of the proposed FSEV SOC, SOE, PL soft sensors was 99.96%, 99.96%, and 99.99%, respectively. The proposed soft sensors attained higher prediction accuracy than that of the modelling structures mentioned above. FSEV results also indicated that the SOC and SOE dropped from 97% to 93.5% and 93.8%, respectively, during the running time of 118 s (one lap time). Thus, two-layer feed-forward neural-network-based soft sensors can be applied for the effective monitoring and prediction of SOC, SOE, and PL during the operation of EVs.

Keywords: soft sensor; state of charge; state of energy; power loss; neural networks; formula student electric vehicle; battery pack

1. Introduction

Electric vehicles (EV; E-mobility) are gaining wider acceptance in mainstream applications as a greener alternative to the conventional fossil-fuel-based locomotion. E-mobility is a primary focus in the global research efforts to limit and reduce the damage caused to Earth's natural ecosystems due to the effects of the fossil-fuel-based economy. The past decade has witnessed many investigations aimed towards the design, development, and deployment of E-mobility technologies for a more sustainable future. Recent efforts have focused on making E-mobility systems more efficient and affordable. Tran et al. [1] optimised the performance of a hybrid electric vehicle powertrain considering its various components and arrangements. Pappalardo et al. [2] developed a model on the basis of the virtual prototype of an electric vehicle. Vita et al. [3] investigated the integration of electric vehicles with renewable-energy-based power grids and distribution networks.

The most important component of an EV drive train is its battery. The performance and reliability of the battery pack directly impacts the overall functioning of the EV during the drive cycle. Sen and Kar [4] modelled an electric-vehicle battery pack for the effective analysis of the battery management system. Lithium ion batteries are reliable with regard to the power and energy requirements of an EV. Researchers attempted to model lithium ion battery packs to characterise their power and energy performance under varying operating conditions. Jaguemont et al. [5] modelled the lithium ion battery pack of an electric vehicle operating under low-temperature conditions. Cui et al. [6] derived a model for the lead acid battery of an energy-storage power station and applied the extended Kalman filtering method to estimate its SOC. Li and Mazzola [7] modelled the lithium ion battery pack as a an analogue electrical battery system with constant parameters for automotive applications.

The state of charge (SOC) is an important battery-pack performance characteristic. Battery management system (BMS) design is completely dependent upon accurate SOC estimation. Liu et al. [8] explored novel methods of SOC estimation and modelling. Ting et al. [9] applied Kalman filter for estimating SOC of the EV BMS. The accurate modelling of component characteristics based on operating conditions is vital for reasonably estimating performance characteristics of any machinery. Researchers have attempted the modelling and control of various mechanical systems using system identification methods. Shah and Sekhar [10] implemented system identification on the basis of fractional order modelling to identify a closed-loop DC motor system. Sekhar et al. [11] designed a control-theory-based system modelling of nano composite machining. Similar system-identification-based modelling applications include the characterization of complex materials [11,12], robotic arms [13], manufacturing processes [14–16], bolted joints [17], and biofuel engine dynamics [18]. The latest modelling methods involving artificial-intelligence and machine-learning methods exhibited excellent prediction consistency in manufacturing systems [19–21]. In the case of EV systems, SOC, SOE, and PL modelling is difficult due to the highly nonlinear characteristic of the battery under varying operating conditions [22]. Catano et al. [23] developed a dynamic model for SOC estimation in real time. Liu et al. [8] explored novel methods of battery SOC estimation and modelling. Zhang et al. [24] employed fractional order modelling and fractional Kalman filters to estimate the SOC of lithium ion battery packs. Xiao et al. [25] also implemented Thevenin fractional order modelling in SOC estimation of an EV lithium ion battery pack. Meng et al. [26] compared the relative merits and demerits of support vector machine (SVM), single-particle model (SPM), and RC equivalent cell models (ECM) in EV SOC estimation. Many researchers applied various forms of artificial neural networks to accurately model SOC estimation in E-mobility systems [27–29]. Li et al. [30] applied fuzzy neural networks in conjunction with a reduced form of the genetic algorithm for the same. Weigert et al. [31] achieved 95% prediction accuracy of SOC models using artificial neural networks. Li et al. [22] minimised SOC prediction errors to 1.75% using recurrent neural networks. He et al. [27] used unscented Kalman filtering to improve neural network prediction accuracy of SOC estimation. Investigations were also carried out with regard to the fault diagnosis and motor design of FSEV [32–34].

Similar research was carried out for the EV battery state of energy (SOE) estimation. Zhang et al. [35] implemented the covariance matching method to update the battery model parameters on a real time basis. In a similar study, Zhang et al. [36] followed the adaptive unscented Kalman filter approach to estimate the SOE of a lithium ion battery. He et al. [37] developed a Gaussian model for SOE prediction, whereas Zheng et al. [38] applied the moving window energy integral method for the same. Li et al. [39] followed a unique methodology that combined an analytical model with an electrical model to improve SOE estimation. Lin et al. [40] applied a multimodel-probability-based state fusion methodology for SOE estimation. Zhang et al. [41] established a multi time scale observer for SOE assessment. Li et al. [42] explored fractional calculus-based SOE modelling for $LiFePO_4$ battery. Fewer investigations were carried out for power-loss (PL) modelling in EV. In an interesting study, Elpiniki et al. [43] analysed EV PL as a function of charging rate and SOE.

This brief literature review indicates that the past investigations did not include SOC/SOE/PL soft sensor designs for FSEV applications.The exploration of effective PL estimation and modelling strategies is an open research area. Therefore, a novel research scope was identified for the current study to estimate the SOC, SOE, and PL of a formula student electric vehicle (FSEV) battery pack based on neural-network-enabled soft-sensor designs. The following section gives details of the battery-pack modelling and neural-network-based soft-sensor design methodology.

2. Methodology

This section presents the details of the FSEV battery-pack specifications, BMS design constraints, battery-pack modelling procedure, and the soft-sensor design methodology, including neural-network implementation in MATLAB and the specifics of the ANN training algorithm. Lastly, alternative modelling structures considered in the current study for comparison with the proposed ANN soft sensors are presented.

2.1. FSEV Battery Pack Modelling

The FSEV was designed as per the vehicle characteristics specified in the Formula Bharat rule book [44]. The features of the designed vehicle (Table 1) and the Formula Bharat track parameters at the Kari Motor Speedway, Coimbatore, India were fed into OptimumLap software to obtain the required drive cycle. Drive cycle characteristics and FSEV features were put into the vehicle powertrain model created in Simulink, MATLAB. The powertrain model resulted in the input current profile (Figure 1) for further computations. The charging current was limited to 20 A as an additional safety margin considering the 40 A limit specified by the cell manufacturer design guide (Table 2). The FSEV battery pack was designed as per the constraints and specifications provided by the competition rulebook and the A123 cell manufacturer (based on real experimentation datasets) (Tables 2 and 3).

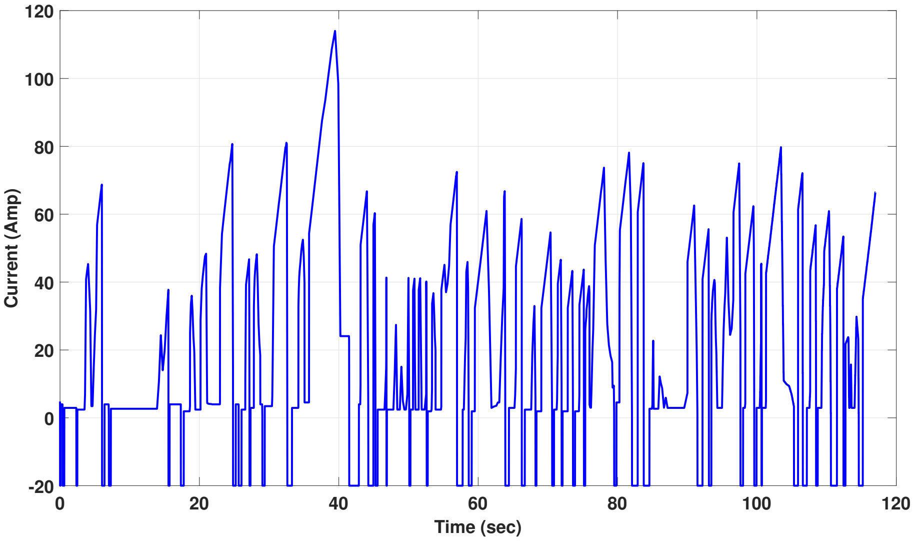

Figure 1. Current profile for a single cell obtained from powertrain model.

Table 1. FSEV vehicle design parameters [45].

Parameter/Specification	Value	Parameter/Specification	Value
Vehicle Mass	300 kg	Tire Rolling Drag	0.03
Drag coefficient	0.7	Power Scaling Factor	95%
Downforce coefficient	0.5	Aero Scaling Factor	90%
Frontal area	0.9 m^2	Grip Scaling Factor	90%
Drivetrain efficiency	90%	Mass Lateral Friction	260 kg
Tire rolling radius	0.203 m	Mass Longitudinal Friction	260 kg
Air density	$1.23\ kg/m^3$	Aero Efficiency	0.7142
Final drive ratio	5	Motor Thermal Efficiency	80%
Longitudinal friction	1.4	Motor Torque Data	90 N·m
Lateral friction	1.5	Motor RPM Data	6500 rpm

Table 2. BMS constraints.

Parameter	Lower Limit	Upper Limit
Charge current	0 A	40 A
Discharge current	0 A	300 A
Cell voltage	2 V	3.65 V
Cell temperature	0 °C	55 °C

Table 3. Battery pack specifications.

Specifications	Value
Drive cycle distance	22 km
Pack voltage	300 V
Range	22 km
Cell voltage nominal	3.3 V
Cell capacity	20 Ah
Pack capacity	20 Ah

The FSEV battery pack was designed to have 90 cells connected in series, all drawing the same amount of current. In the current study, SOC was estimated for one cell assuming ideal operating conditions over one lap of the Formula Bharat track. For cell parameter estimation, the series internal resistance (R_0) was first approximated as per the direct current resistance (DCR) $-$% SOC characteristics prescribed in the A123 manufacturer design guide (based on actual experiments) [46]. The remaining cell parameters, namely, parallel internal resistance (R_1) and capacitance (C_1), were determined by mutual substitution in the following equations (also derived from actual experiments) [47,48]

$$R_0 = -0.000513(SOC) + 0.002733 \tag{1}$$

$$R_1 = 0.001426(SOC)^{-0.771947} \tag{2}$$

$$C_1 = 3297.55log(SOC) + 13,481.96 \tag{3}$$

These R_0, R_1, C_1 values along with the input current profile were utilised to compute voltage drops across R_0, R_1 and C_1. Subsequently, these voltage drops and the input current profile were employed to obtain the total PL due to the cell components R_0, R_1 and C_1. Thereafter, battery pack temperature was estimated using total PL (P_{loss}) as per the following relation

$$C_T * \frac{dT}{dt} = (\text{ambient Temp} - \text{current Temp}) * R_T + P_{loss} \tag{4}$$

where C_T is specific heat capacity (J m^{-3} K^{-1}), and R_T is convection resistance (W m^{-2} K^{-1}) of the cell. Next, the charge capacity of the battery was estimated at all operating temperatures derived in Equation (4) as per the following relation.

$$\text{Capacity} = -0.001238 * T^2 + 0.7507 * T - 94.06 \tag{5}$$

On the basis of the derived temperature–capacity profile, the SOC was estimated as follows.

$$\text{SOC (at time t)} = \text{initial SOC} - \int \frac{\text{I}}{\text{Capacity} * 3600} dt \tag{6}$$

where I is the battery current, and the initial SOC was assumed to be 0.97. Next, open circuit voltage (OCV) was estimated from the experimental OCV, SOC, and temperature data sheet provided by the cell manufacturer [46]

$$\text{OCV} = 3.149 * \text{SOC} - 7.551 \times 10^{-5} * \text{T} + 0.523 \tag{7}$$

Terminal voltage (V_t) was obtained as follows.

$$V_t = \text{OCV} - V_{d1} - V_{d2} \tag{8}$$

where V_{d1} and V_{d2} are the potential differences across R_0 and R_1, respectively. These voltages were determined using the following relations.

$$V_{d1} = I_0 R_0$$
$$V_{d2} = I_1 R_1$$

where I_0 is the current through R_0, and I_1 is the current flowing through R_1 in the equivalent cell model. Subsequently, SOE was calculated as follows.

$$\text{SOE (at time t)} = \text{initial SOE} - \int \frac{\text{P}}{\text{nominal energy}} dt \tag{9}$$

where P is power of the FSEV battery in watts. Nominal energy of the cell was obtained from the design data sheet [46], and the initial SOE was assumed to be 0.97. Figure 2 shows a schematic representation of the SOC, SOE, and PL estimation procedure followed in the current study. The following section gives details of the proposed soft-sensor designs for these FSEV battery parameters.

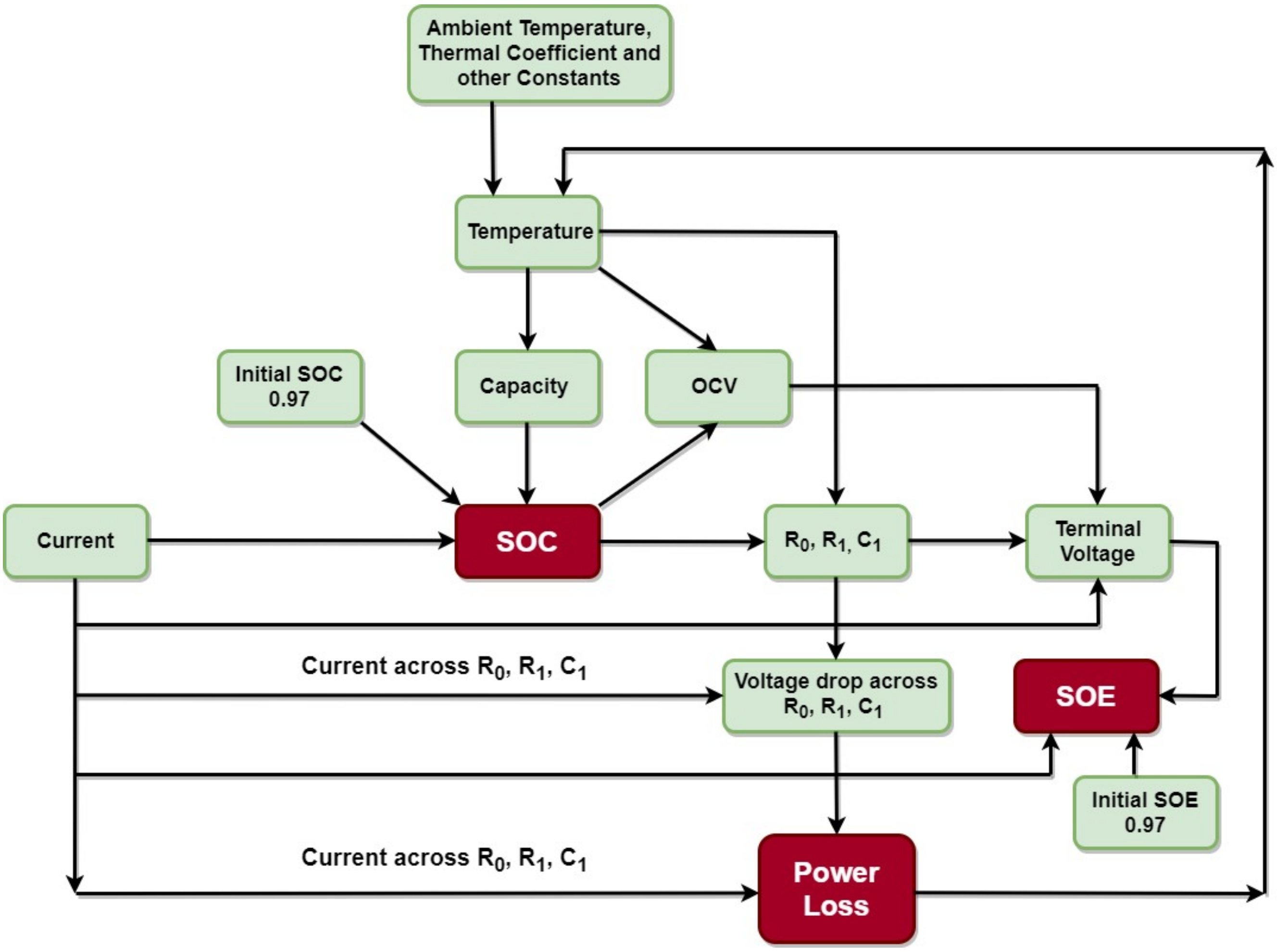
Ambient Temperature, Thermal Coefficient and other Constants
Temperature
Initial SOC 0.97
Capacity
OCV
Current
SOC
R_0, R_1, C_1
Terminal Voltage
Current across R_0, R_1, C_1
Voltage drop across R_0, R_1, C_1
SOE
Initial SOE 0.97
Current across R_0, R_1, C_1
Power Loss

Figure 2. SOC, SOE, and PL estimation procedure.

2.2. Soft-Sensor Design

Data-based modelling generally begins with preprocessing that includes data cleaning and normalization, followed by partitioning data into training, validation, and testing sets [49]. Figure 3 exhibits the two-layer feed-forward artificial neural-network architecture for the soft sensor application proposed in the current work. The two layer feedforward neural network in association with the Levenberg–Marquardt algorithm for data fitting exhibited high fitting accuracy at the cost of lesser training iterations and data points [50]. The vehicle drive cycle characteristics were applied to the input nodes I_1 (time) and I_2 (current). Weights $W_{1,1}, W_{1,2}, \cdots, W_{1,10}$ were assigned to the connections from input node I_1 to the ten nodes in the hidden layer. Similarly, weights $W_{2,1}, W_{2,2}, \cdots, W_{2,10}$ were applied to the connections from input node I_2 to the hidden layer nodes. The output of each hidden layer comprised the sum of the products of all input nodes to their respective assigned connection weights. All hidden layer outputs were treated with sigmoid activation function f_1 (Figure 4), defined as follows.

$$S(x) = \frac{1}{1+e^{-x}} \tag{10}$$

where x is the input to the activation function, and $S(x)$ is the sigmoid activation function. The connections of all activation function outputs to the output layer nodes were also assigned weights $(W_{o,1}, W_{o,2}, \cdots, W_{o,10})$. For the final predicted output, a linear activation function f_2 (Figure 5) was applied to the summation of the products of all f_1 function outputs and their respective weights. This linear activation function is defined as follows.

$$L(x) = A * x \tag{11}$$

where x is the input to the activation function, A is the slope of the activation function and $L(x)$ is the linear activation function. The above modelling procedure was individually carried out for the estimations of SOC, SOE, and PL for a single cell of the FSEV battery pack considered in the current study.

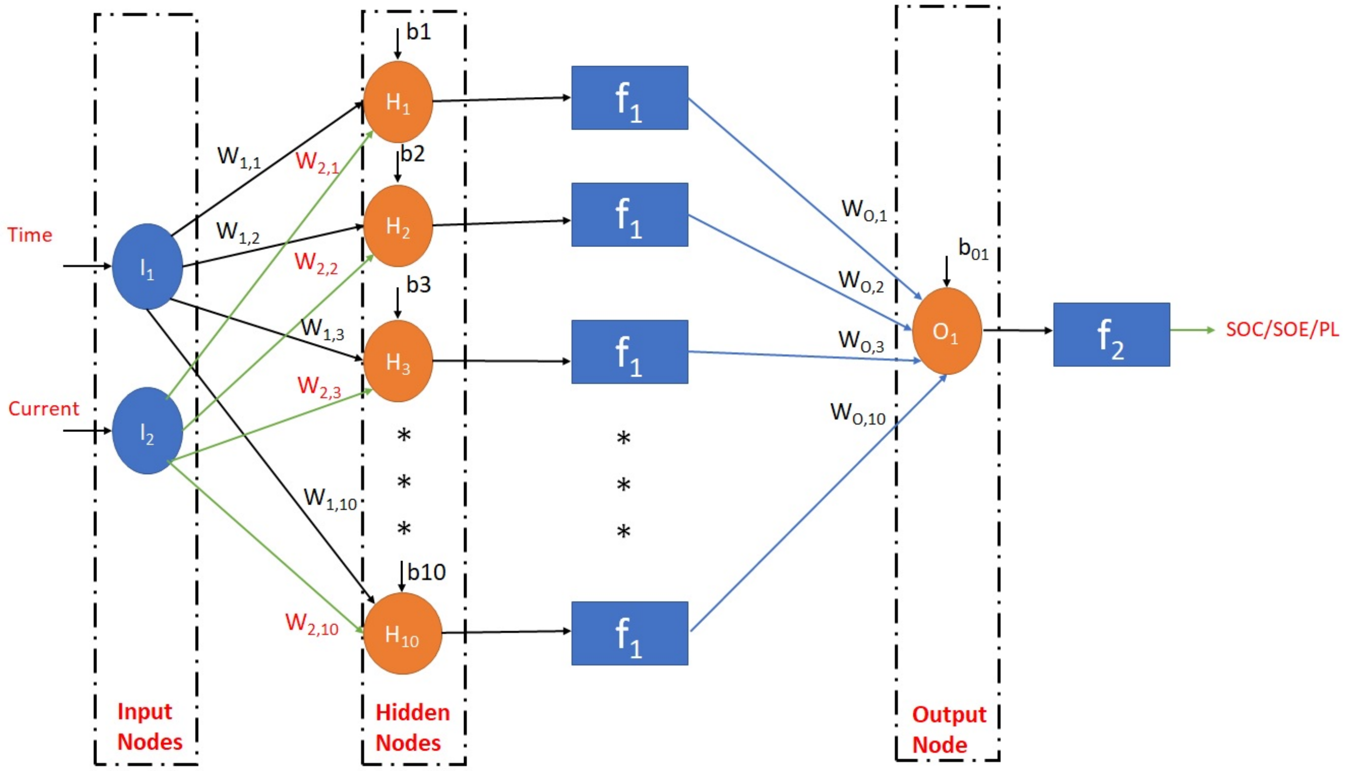

Figure 3. Two-layer feed-forward neural-network architecture for FSEV battery pack SOE/SOC/PL soft-sensor design: sigmoid activation function (f_1) at the hidden layer and linear activation function (f_2) at the output layer.

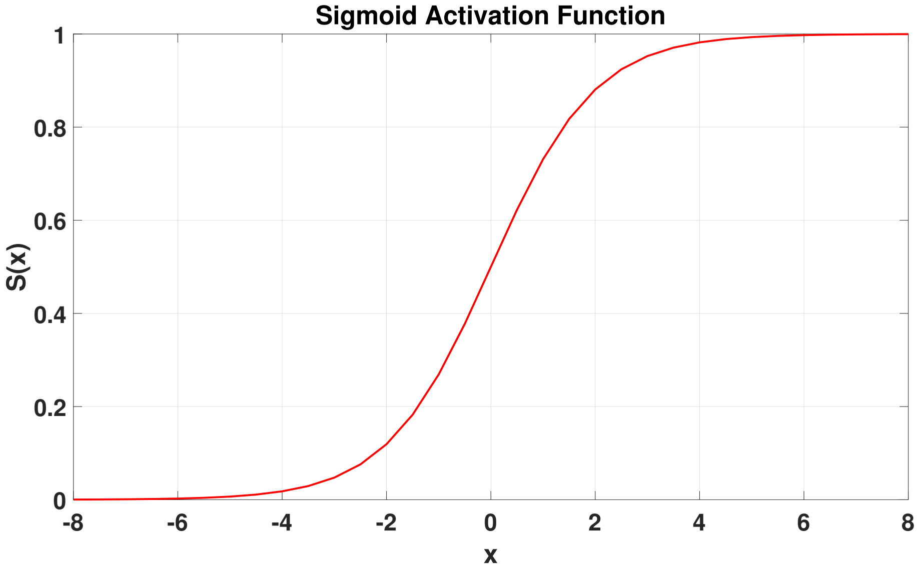

Figure 4. Sigmoid activation function.

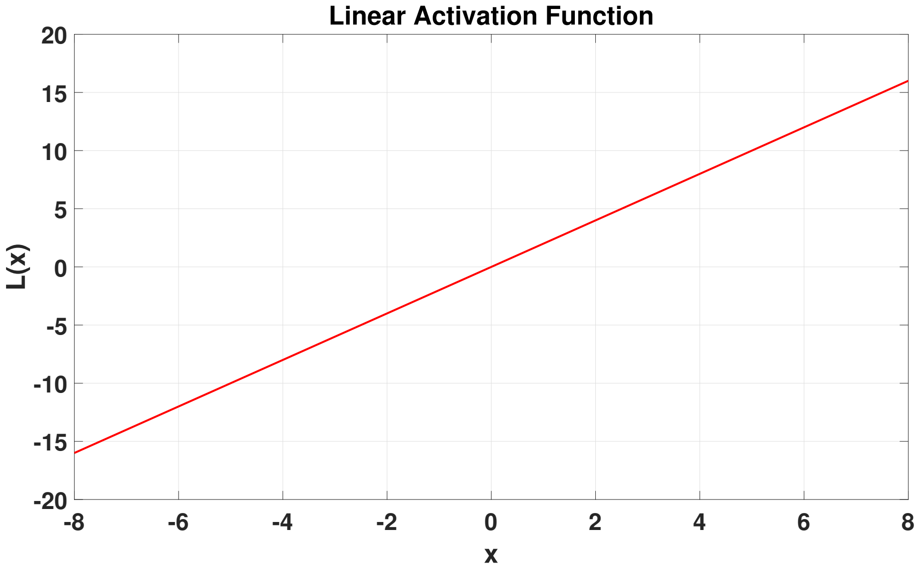

Figure 5. Linear activation function.

Next was the stepwise implementation of the neural-network-based soft sensors in MATLAB.

1. Load the input current profile and SOC/SOE/PL dataset for the FSEV battery pack considered in the current study.
2. Partition the dataset into training, validation, and testing sets. The total dataset available for the current study was partitioned 70% for training, and 15% each for validation and testing (Table 4).
3. Select the neural-network architecture for predictive modelling (two-layer feedforward network with sigmoid transfer function in the hidden layer and a linear transfer function in the output layer).
4. Select the number of hidden neurons (selected as 10 in the present work).
5. Train the neural network using a supervised learning algorithm (Levenberg–Marquardt).
6. Validate and test the trained neural network.
7. Retrain the network if performance is poor in terms of the R-squared value of the testing dataset.

Table 4. Dataset partitions for SOC/SOE/PL soft sensors.

Dataset Partitions	Number of Samples	Data%
Training	82,132	70
Validation	17,600	15
Testing	17,600	15
Total	117,332	100

Table 5 shows salient features of the feed-forward neural network architecture implemented for soft-sensor design. The following subsection provides details of the training algorithm employed in this neural network.

Table 5. Two-layer feedf-orward neural-network features.

Training Method	Levenberg-Marquardt
Sample division	as shown in Table 4
Activation function (f_1)	Sigmoid function
Activation function (f_2)	Linear function
Initial weights and bias	zero
Input layer nodes	2
Output layer nodes	1
Hidden layer nodes	10
Performance index	Mean squared error (MSE)

Levenberg–Marquardt Training Algorithm

The artificial neural network (ANN) training algorithm formulates estimation error as an objective function to be minimised through every training iteration (Figure 6). It distributes the total estimation error among the nodal connection weights. Learning and decay rates are defined to control the upper and lower limits, respectively, of these weights. The optimization algorithm minimises the estimation error over successive training epochs. The transformation function sums up the products of input nodes and respective weights, and yields the output value in conjunction with a bias factor and an activation function. The weighted sums are scaled up to the values of the input nodes by the bias factor to guarantee that the input node values are preserved for further calculations. The activation function ensures that at least the minimal threshold of the calculated nodal outputs is contributed to the next layer [51].

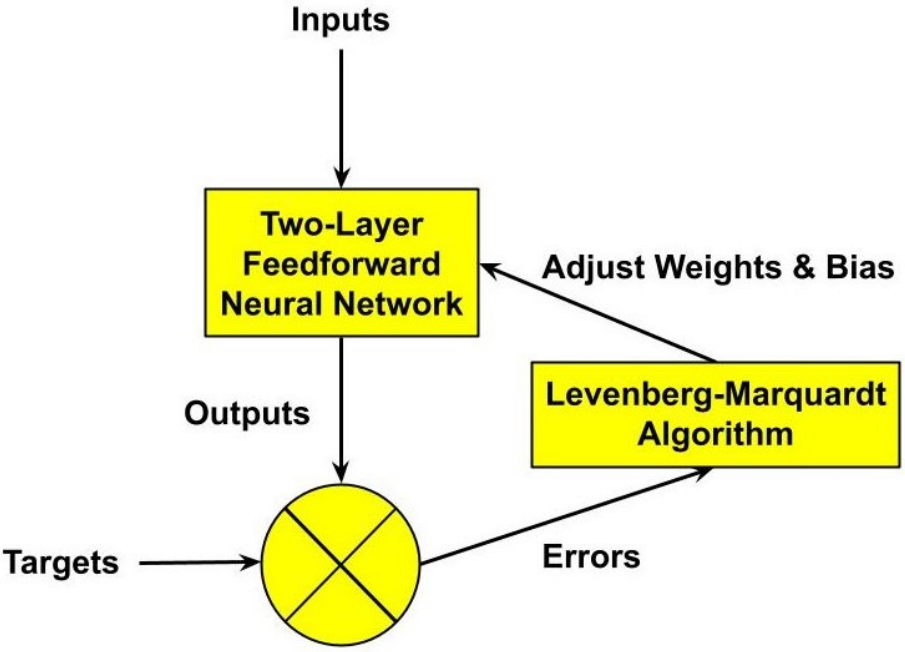
Inputs
Two-Layer Feedforward Neural Network
Adjust Weights & Bias
Levenberg-Marquardt Algorithm
Outputs
Targets
Errors

Figure 6. Supervised learning flow.

The Levenber–Marquardt algorithm employs a Jacobian matrix in conjunction with the gradient vector and expresses the loss function as the sum of squared errors (f).

$$f = \sum_{r=0}^{N} e_r^2 \tag{12}$$

where N is the number of data set instances, and e is the difference between derived and predicted SOC values [52,53]. The Jacobian matrix of this loss function is composed of the error derivatives having N by P dimensions.

$$\mathbf{J}_{i,j} = \frac{\partial e_i}{\partial \mathbf{W}_j} \tag{13}$$

where $i = 1, 2, \cdots, N$ is the number of data set instances, $J = 1, 2, \cdots, P$ is the number of neural network parameters and $\mathbf{W}_j$ is the weight vector corresponding to a particular input. The gradient vector is determined for this loss function as follows.

$$\nabla f = 2\mathbf{J}^T \cdot e \tag{14}$$

The Hessian matrix is approximated as follows.

$$\mathbf{Hf} \approx 2\mathbf{J}^T \cdot \mathbf{J} + \mu \mathbf{I} \tag{15}$$

where $\mathbf{I}$ is the identity matrix, and μ is the Hessian damping factor. Lastly, loss function parameters are improved by the algorithm as per the following expression [54].

$$\mathbf{W}^{i+1} = \mathbf{W}^i - (\mathbf{J}^{(i)T} \cdot \mathbf{J}^{(i)} + \mu^i \mathbf{I})^{-1} \cdot (2\mathbf{J}^{(i)T} \cdot e^i) \tag{16}$$

The network training process is terminated on the basis of the algorithm meeting any one of the stopping criteria depicted in Table 6.

Table 6. Training algorithm stopping criteria.

Sr. No.	Stopping Criteria	Settings
1.	Maximal epochs	1000
2.	Maximal training time	∞
3.	Performance goal	0
4.	Minimal performance gradient	1.00×10^7
5.	Maximal μ	1.00×10^{10}
6.	Maximal validation fails	6

For comparative analysis, the SOC, SOE, and PL of the FSEV considered in the current study were also modelled using linear regression, nonlinear regression, and parametric structures such as ARX, ARMAX, OE, and BJ models. Table 7 depicts the generalized forms of these parametric model structures widely used in the system identification of various systems. The prediction accuracy of all developed models was compared on the basis of R-squared/FIT% and mean squared errors (MSE).

Table 7. Parametric model structures.

Sr. No.	Model Name	Model Structure
1	Output error (OE)	y(t) = [B(z)/F(z)]u(t) + e(t)
2	Autoregressive moving average with exogenous input (ARMAX)	A(z)y(t) = B(z)u(t) + [C(z)/(1 − z^{-1})]e(t)
3	Box Jenkins (BJ)	y(t) = [B(z)/F(z)]u(t) + [C(z)/D(z)]e(t)
4	Autoregressive with exogenous input (ARX)	A(z)y(t) = B(z)u(t) + e(t)

3. Results and Discussion

This section presents the various performance metrics of the proposed FSEV SOC/SOE/PL soft sensors designed on the basis of two-layer feed-forward neural networks. The comparative performance of these soft sensors with regard to the regression and parametric models is also discussed.

3.1. State of Charge Soft Sensor

Table 8 shows the final hidden or output layer weights and biases for the FSEV SOC neural network, and Table 9 depicts the mean squared error (MSE) and R squared (R-sq) values of the proposed SOC soft sensor as per various dataset partitions. The training, validation, and testing datasets attained very impressive MSEs of 8.28 × 10^8, 8.30 × 10^8 and 8.34 × 10^8, respectively. All datasets achieved R squared values of 0.9996 each, showing excellent prediction accuracy for the proposed FSEC SOC soft sensor. Figure 7 shows the regression plot of the FSEV SOC sensor performance relative to various dataset partitions.

Figure 8 shows a graphical presentation of the derived versus soft-sensor-predicted SOC values over one lap time of 118 s. The graph shows that the SOC decreased from 97% at the beginning of the lap to 93.5% at the end of the lap. The soft-sensor model output closely matched that of the derived SOC. Figure 9 shows the error histogram for the FSEV SOC soft-sensor predictions. It depicts a maximal number of testing instances (around 4000) at an error bin of 1.99 × 10^{-5}, followed by comparatively lower instances (around 2500) at 1.34 × 10^{-4} and −9.4 × 10^{-5}. There are around a 1000 instances at the bin corresponding to 2.47 × 10^{-4} error. Remaining error bins (−4.3 × 10^{-4}, −3.2 × 10^{-4}, −2.1 × 10^{-4}, 3.61 × 10^{-4}, 4.75 × 10^{-4} and others) contain lesser instances. This distribution of error instances at very low magnitudes supports the excellent predictability of the proposed FSEV SOC soft-sensor design. Figure 10 depicts the gradient, mu, and validation checks as indicators of the FSEV SOC soft-sensor training states. The gradient continuously decreased till 5.76 × 10^{-8} at epoch 25, whereas mu settled at 1 × 10^{-10} at the same epoch. The gradient is an indicator of the learning rate and step size of the training algorithm. The sloping reduction in gradient over successive epochs indicated the systematic progression of the training algorithm towards the global minimum of the proposed soft-sensor prediction error. In fact, the training algorithm terminated further epochs because the stopping criterion of minimal performance gradient of 1 × 10^{-7} (Table 6) had been attained. A low mu indicated that the Levenberg–Marqaurdt algorithm moved towards Newton's method and away from the gradient descent methodology for faster and more accurate convergence [55]. The Figure 10 validation check depicts that there was no validation fail till epoch 25. Figure 11 shows that the best validation performance of the proposed soft sensor was recorded to be a very low MSE value of 8.30 × 10^{-8} attained at epoch 25.

Table 8. Input and output weights and biases for two-layer feed-forward FSEV SOC soft sensor.

Weight	Value	Weight	Value	Weight	Value	Bias	Value
$W_{1,1}$	−6.7388	$W_{2,1}$	0.1385	$W_{o,1}$	−0.3445	b_1	5.2081
$W_{1,2}$	1.5055	$W_{2,2}$	4.4055	$W_{o,2}$	−0.1561	b_2	−2.9739
$W_{1,3}$	−2.2395	$W_{2,3}$	0.1014	$W_{o,3}$	0.9477	b_3	0.5772
$W_{1,4}$	−4.0909	$W_{2,4}$	5.0135	$W_{o,4}$	−0.0010	b_4	0.5644
$W_{1,5}$	−0.9409	$W_{2,5}$	−4.7907	$W_{o,5}$	1.1354	b_5	1.5813
$W_{1,6}$	−3.0955	$W_{2,6}$	−3.3766	$W_{o,6}$	−0.2886	b_6	0.1238
$W_{1,7}$	−9.3968	$W_{2,7}$	0.4713	$W_{o,7}$	−0.8465	b_7	−5.1520
$W_{1,8}$	−18.7391	$W_{2,8}$	0.6081	$W_{o,8}$	−0.8844	b_8	−6.6339
$W_{1,9}$	−3.9745	$W_{2,9}$	0.9659	$W_{o,9}$	0.2007	b_9	−4.1959
$W_{1,10}$	3.7267	$W_{2,10}$	−1.6611	$W_{o,10}$	−0.1511	b_{10}	3.7865
						b_{o1}	0.1713

Table 9. FSEV SOC soft sensor performance.

Dataset Partitions	MSE	R-sq
Training	8.28 × 10^{-8}	0.9996
Validation	8.30 × 10^{-8}	0.9996
Testing	8.34 × 10^{-8}	0.9996
Overall	-	0.9996

The y label of the test regression plot exhibits the following relation between derived and soft-sensor-predicted SOC.

$$\text{Output (predicted SOC)} = \text{Target (derived SOC)} + 6.8 \times 10^{-8} \tag{17}$$

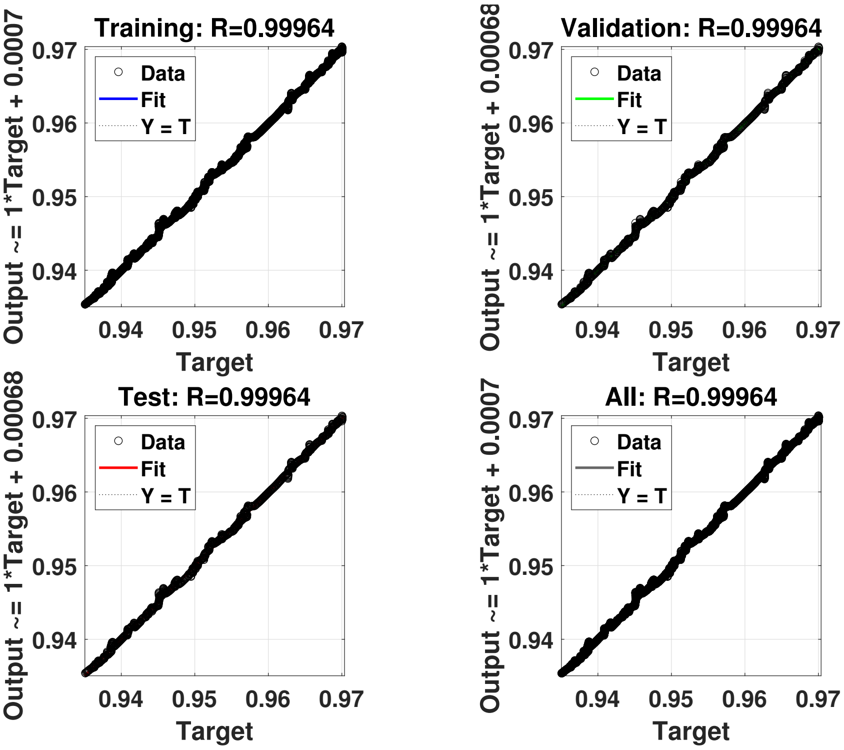

Figure 7. Regression plots of FSEV SOC soft sensor.

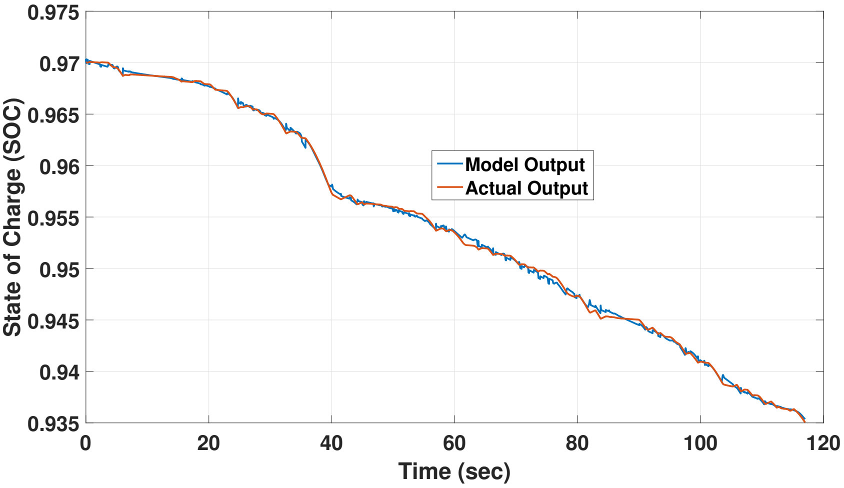

Figure 8. Derived versus soft sensor outputs of FSEV SOC.

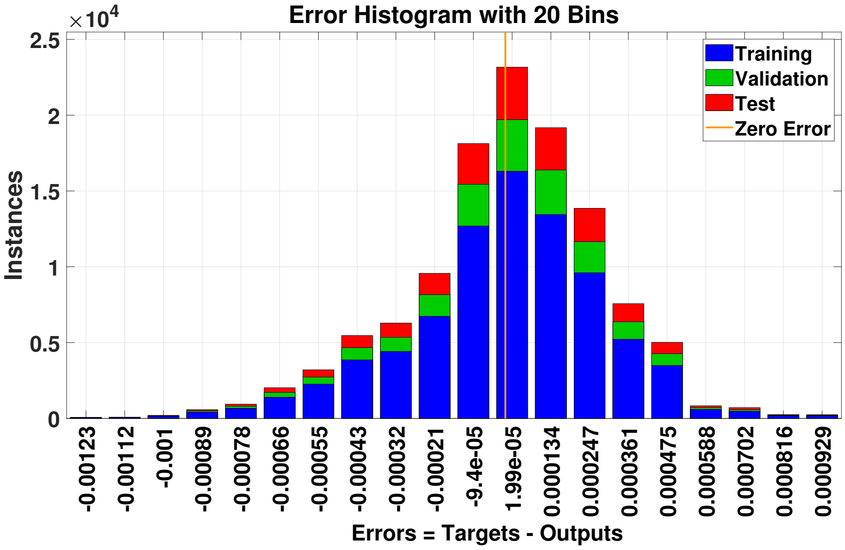

Figure 9. Error histogram of FSEV SOC soft sensor.

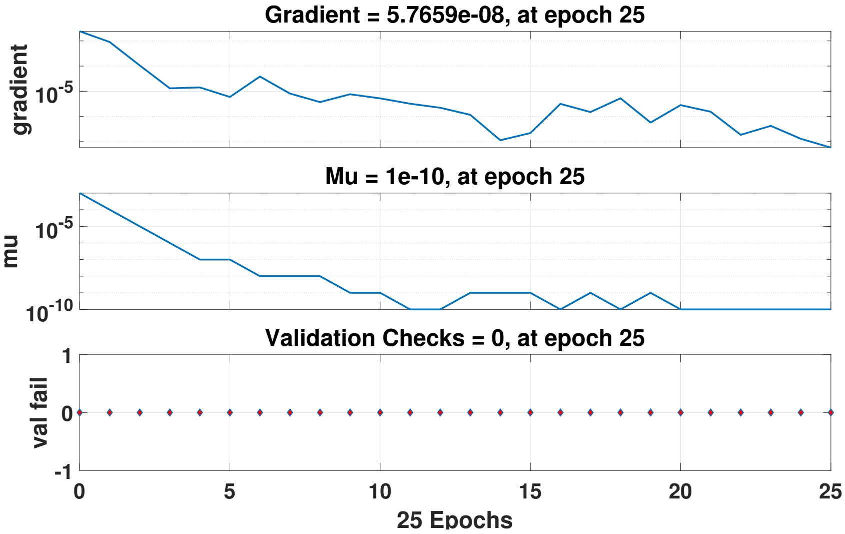

Figure 10. Training states of FSEV SOC soft sensor.

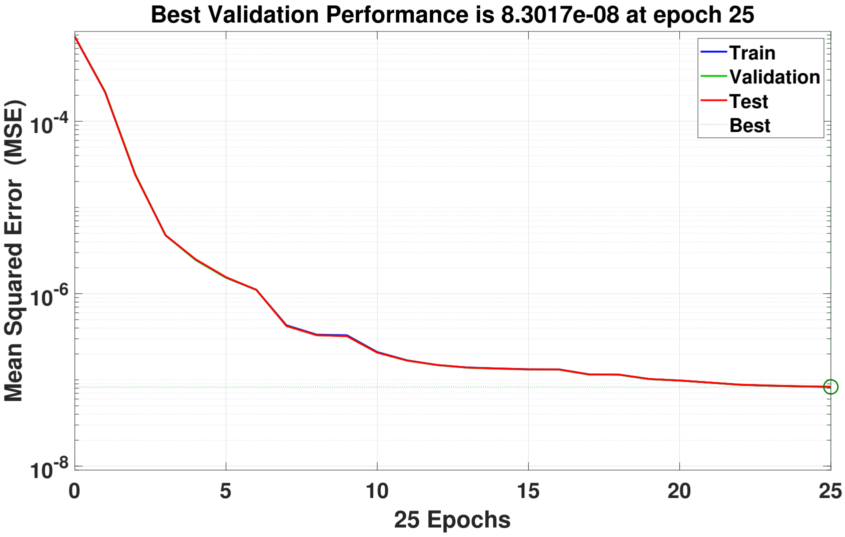

Figure 11. Validation performance of FSEV SOC soft sensor.

The SOC of the FSEV considered in the current study was also modelled using linear and nonlinear regression as follows: Regression–Linear

$$\mathrm{SOC} = 0.9724 - 0.0003157 * t + 1.911 \times 10^{-6} * I \tag{18}$$

Regression—Nonlinear

$$\begin{aligned} \mathrm{SOC} = 0.9721 - 0.0002997 * t + 2.342 \times 10^{-5} * I - 1.191 \times 10^{-7} * t^2 \\ -1.225 \times 10^{-7} * t * I - 2.432 \times 10^{-7} * I^2 \end{aligned} \tag{19}$$

The FSEV SOC was also modelled using parametric structures generally considered for the system identification of control systems.

ARX 441 *A(z)y(t) = B(z)u(t) + e(t)*

$$\begin{aligned} A(z) &= 1 - 0.3499z^{-1} - 0.1942z^{-2} - 0.2977z^{-3} - 0.1582z^{-4} \\ B(z) &= -1.964 \times 10^{-7}z^{-1} - 6.007 \times 10^{-8}z^{-2} - 6.836 \times 10^{-8}z^{-3} + 4.473 \times 10^{-9}z^{-4} \end{aligned}$$

ARMAX 2221 *A(z)y(t) = B(z)u(t) + C(z)e(t)*

$$\begin{aligned} A(z) &= 1 - 1.647z^{-1} + 0.6465z^{-2} \\ B(z) &= -1.663 \times 10^{-7}z^{-1} + 1.162 \times 10^{-7}z^{-2} \\ C(z) &= 1 - 1.42z^{-1} + 0.4327z^{-2} \end{aligned}$$

BJ 22221 *y(t) = [B(z)/F(z)]u(t) + [C(z)/D(z)]e(t)*

$$\begin{aligned} B(z) &= -2.316 \times 10^{-7}z^{-1} - 9.932 \times 10^{-9}z^{-2} \\ C(z) &= 1 - 0.3207z^{-1} - 0.4296z^{-2} \\ D(z) &= 1 - 0.5763z^{-1} - 0.402z^{-2} \\ F(z) &= 1 - 0.2971z^{-1} - 0.7029z^{-2} \end{aligned}$$

OE 221 *y(t) = [B(z)/F(z)]u(t) + e(t)*

$$\begin{aligned} B(z) &= -1.886 \times 10^{-7}z^{-1} + 8.593 \times 10^{-8}z^{-2} \\ F(z) &= 1 - 1.276z^{-1} + 0.2759z^{-2} \end{aligned}$$

Figure 12 shows the composite plot of R-squared (regression) and FIT% (system identification) models explored in the current study. The ANN-based soft sensor attained the highest accuracy among all models. However, the parametric models scored lower mean squared errors as compared to ANN and regression models (Figure 13). All models attained comparable performance with regard to SOC estimation. However, ANN is capable of handling unseen data better than other structures are [56]. Therefore, the ANN-based soft sensor is expected to exhibit higher robustness in real-time applications involving system variations and dynamic operating conditions. The ANN modelling procedure incorporates the partitioning of data into training, validation, and testing sets, ensuring the reliable predictability of the final model. Other modelling techniques such as regression and system identification do not provide this advantage, making them inferior to ANN with regard to soft-sensor applications.

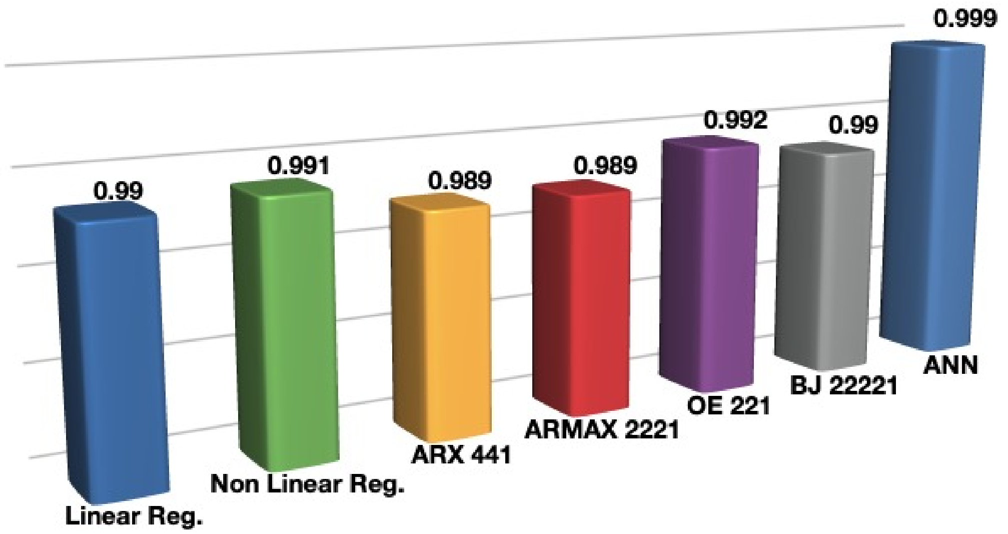

Figure 12. SOC model prediction accuracy: R squared values (linear regression, nonlinear regression, ANN soft sensor) and FIT% (ARX, ARMAX, OE, BJ).

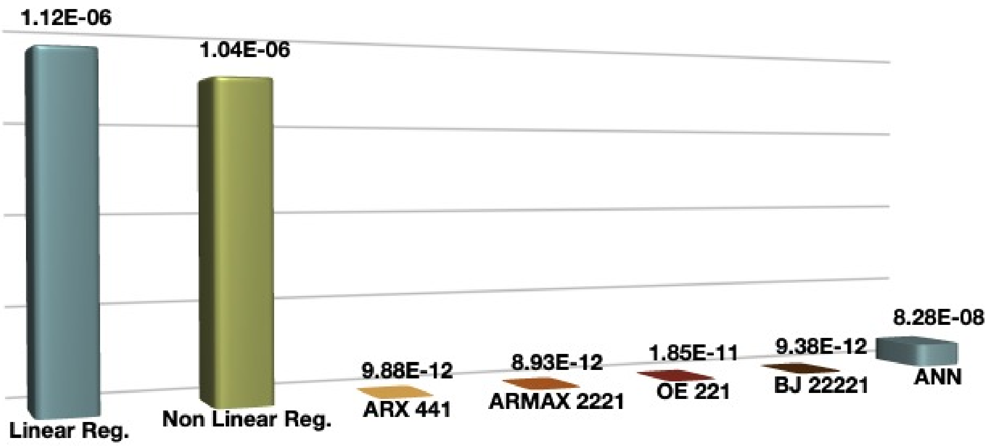

Figure 13. SOC models mean squared errors (MSE).

3.2. State of Energy Soft Sensor

Table 10 shows the input weights, output weights, and biases optimised for the two-layer feed-forward FSEV SOE soft sensor. Table 11 depicts higher MSE of the training (MSE 7.426), validation (MSE 7.4900) and testing (MSE 7.4774) SOE datasets as compared to those of the SOC dataset partitions (Table 9). Nevertheless, the SOE soft sensor attained perfect R squared values of 0.9996 for all datasets. Figure 14 shows the regression plots for the SOE testing, training, and validation datasets. SOE soft-sensor prediction was determined to be correlated with the derived SOE values as follows.

$$\text{Output (predicted SOE)} = \text{Target (derived SOE)} + 4 \times 10^{-4} \quad (20)$$

Table 10. Input and output weights and biases for the two-layer feed-forward FSEV SOE soft sensor.

Weight	Value	Weight	Value	Weight	Value	Bias	Value
$W_{1,1}$	6.8392	$W_{2,1}$	−0.0811	$W_{o,1}$	−0.1686	b_1	−5.2122
$W_{1,2}$	1.5481	$W_{2,2}$	1.9788	$W_{o,2}$	−0.0767	b_2	−1.6627
$W_{1,3}$	−2.7665	$W_{2,3}$	2.5017	$W_{o,3}$	0.0277	b_3	2.2216
$W_{1,4}$	2.6111	$W_{2,4}$	−0.0231	$W_{o,4}$	−0.3693	b_4	−0.6678
$W_{1,5}$	1.1701	$W_{2,5}$	9.4484	$W_{o,5}$	0.0191	b_5	2.1594
$W_{1,6}$	−2.6643	$W_{2,6}$	−2.7645	$W_{o,6}$	0.0396	b_6	−0.0218
$W_{1,7}$	−23.6749	$W_{2,7}$	0.7517	$W_{o,7}$	0.1327	b_7	−8.4033
$W_{1,8}$	−1.0208	$W_{2,8}$	7.0473	$W_{o,8}$	0.0214	b_8	7.0193
$W_{1,9}$	8.2776	$W_{2,9}$	8.9437	$W_{o,9}$	−0.0232	b_9	6.5266
$W_{1,10}$	3.856	$W_{2,10}$	−0.076	$W_{o,10}$	−-0.2172	b_{10}	2.0931
						b_{o1}	−0.0677

Table 11. FSEV SOE soft sensor performance.

Dataset Partitions	MSE	R-sq
Training	7.4276	0.9996
Validation	7.4900	0.9996
Testing	7.4774	0.9996
Overall	-	0.9996

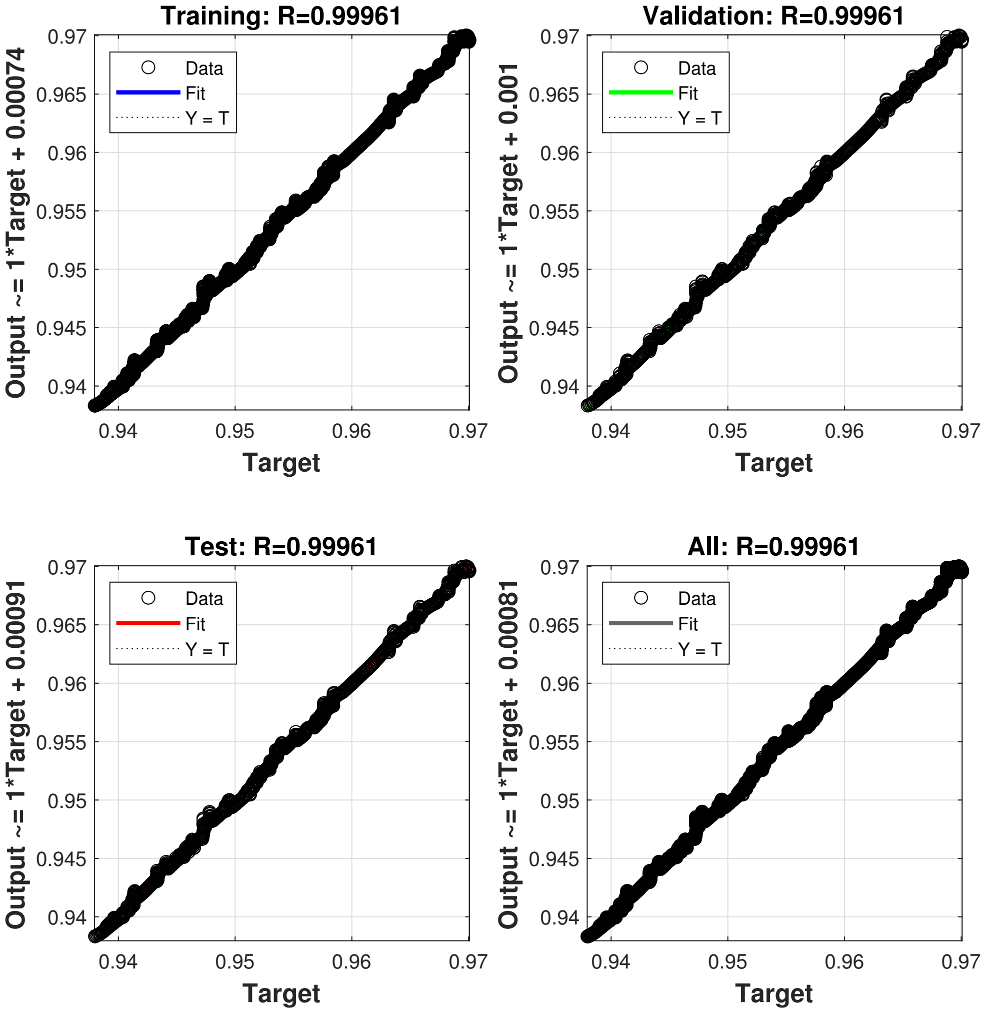

Figure 14. Regression plots of FSEV SOE soft sensor.

Figure 15 shows the plots of derived and predicted SOE outputs for one lap time (118 s) as per the FSEV drive cycle. It is evident that the predicted SOE plot closely matches the derived SOE curve. The figure shows that the SOE dropped from 97% to approximately 93.8% during one lap time period. The final SOE was slightly better than the SOC (93.5%) at 118 s. Figure 16 shows the error histogram for the SOE soft sensor. The maximal number of testing error instances (around 2500) were generated at the error value of 3.88×10^{-5}. More than 2000 testing error instances were detected at the -6.5×10^{-6} error bin. Other major instances numbered approximately 2000, 1900, 1800, and 1700 corresponding to error values of 1.43×10^{-4}, 2.47×10^{-4}, 1.7×10^{-4} and 2.7×10^{-4}, respectively. The remaining testing error instances measured approximately 1500 or lesser for bins corresponding to 3.51×10^{-4}, 4.55×10^{-4}, 3.8×10^{-4}, and 4.8×10^{-4}. Thus, the majority of testing error instances were found at very low values of deviations, confirming the high prediction accuracy of the developed SOE soft sensor. Figure 17 shows the training state indicators for the SOE datasets. The gradient settled at a value of 9.574×10^{-8} at epoch 32, triggering the stopping criterion of the training algorithm. The mu finished at 1×10^{-10}, whereas there were zero validation fails till the epoch 32. Figure 18 indicates that the best validation performance for the FSEV SOE soft sensor was achieved at 7.4901×10^{-8} corresponding to epoch 32. The validation performance of the SOC soft sensor (8.30×10^{-8}) was superior than that of its SOE counterpart. As discussed in the case of SOC modelling, the FSEV SOE was also modelled using regression and system identification methods for performance comparisons with the proposed ANN-based soft sensor: Regression–Linear

$$\mathrm{SOE} = 0.9722 - 0.0002885 * t + 1.258 \times 10^{-6} * I \tag{21}$$

Regression—Nonlinear

$$\begin{aligned}\mathrm{SOE} = 0.9718 - 0.0002669 * t + 2.022 \times 10^{-5} * I - 1.695 \times 10^{-7} * t^2 \\ -1.005 \times 10^{-7} * t * I - 2.25 \times 10^{-7} * I^2\end{aligned} \tag{22}$$

ARX 441 *A(z)y(t)* = *B(z)u(t)* + *e(t)*

$$\begin{aligned}A(z) &= 1 - 0.3732z^{-1} - 0.198z^{-2} - 0.3342z^{-3} - 0.09464z^{-4} \\ B(z) &= -2.078 \times 10^{-7}z^{-1} - 2.821 \times 10^{-8}z^{-2} - 4.796 \times 10^{-8}z^{-3} + 3.061 \times 10^{-9}z^{-4}\end{aligned}$$

ARMAX 2221 *A(z)y(t)* = *B(z)u(t)* + *C(z)e(t)*

$$\begin{aligned}A(z) &= 1 - 0.3427z^{-1} - 0.6573z^{-2} \\ B(z) &= -2.076 \times 10^{-7}z^{-1} - 8.8 \times 10^{-9}z^{-2} \\ C(z) &= 1 + 0.06146z^{-1} - 0.469z^{-2}\end{aligned}$$

BJ 22221 *y(t)* = *[B(z)/F(z)]u(t)* + *[C(z)/D(z)]e(t)*

$$\begin{aligned}B(z) &= -2.236 \times 10^{-7}z^{-1} + 3.515 \times 10^{-8}z^{-2} \\ C(z) &= 1 - 0.3365z^{-1} - 0.2003z^{-2} \\ D(z) &= 1 - 0.7011z^{-1} - 0.2989z^{-2} \\ F(z) &= 1 - 0.5538z^{-1} - 0.4461z^{-2}\end{aligned}$$

OE 221 *y(t)* = *[B(z)/F(z)]u(t)* + *e(t)*

$$\begin{aligned}B(z) &= -1.591 \times 10^{-7}z^{-1} + 1.517 \times 10^{-7}z^{-2} \\ F(z) &= 1 - 1.939z^{-1} + 0.9392z^{-2}\end{aligned}$$

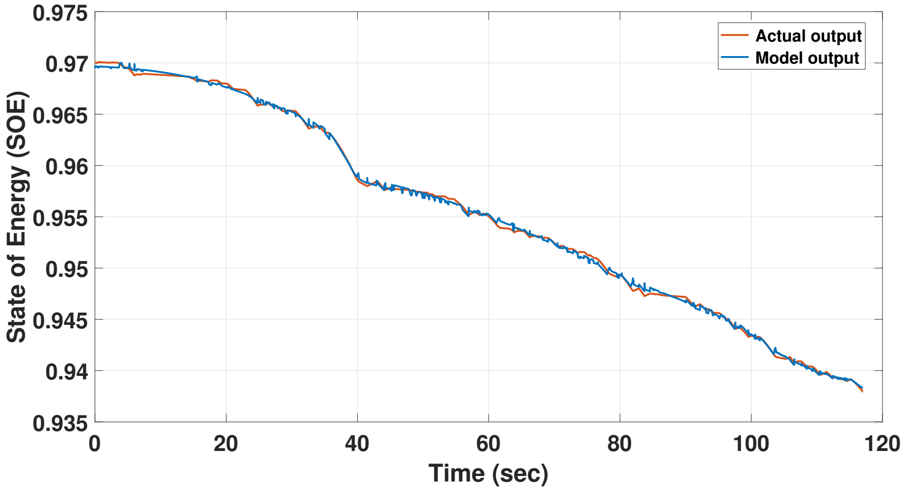

Figure 15. Derived versus soft sensor outputs of FSEV SOE.

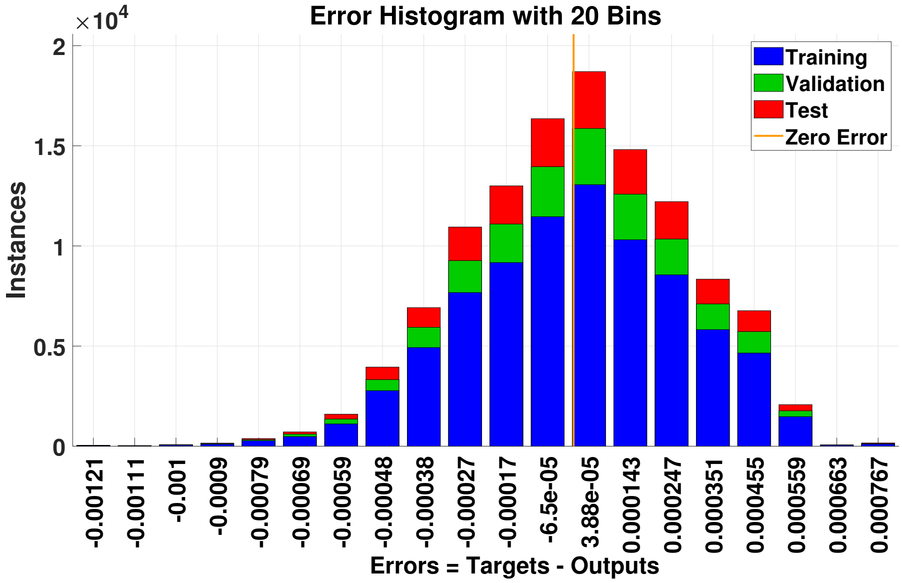

Figure 16. Error histogram of FSEV SOE soft sensor.

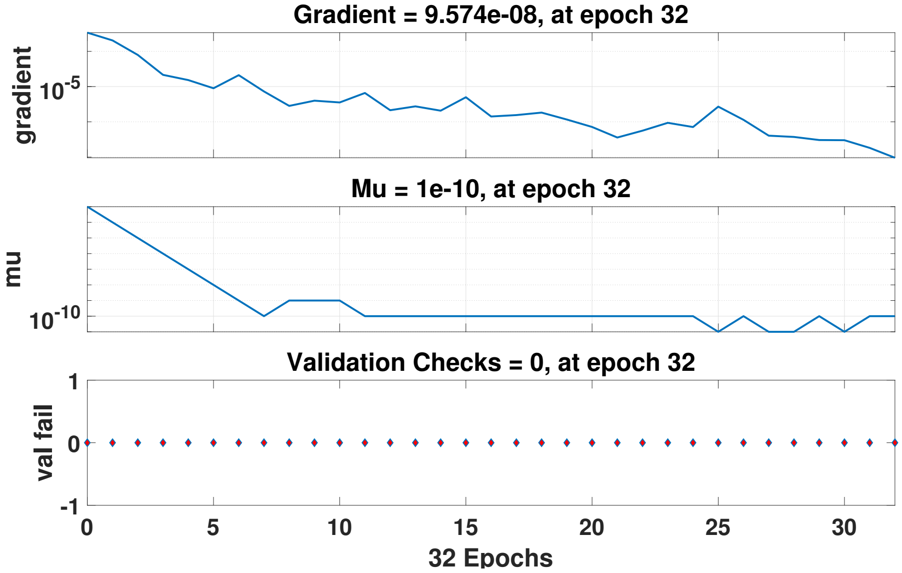

Figure 17. Training states of FSEV SOE soft sensor.

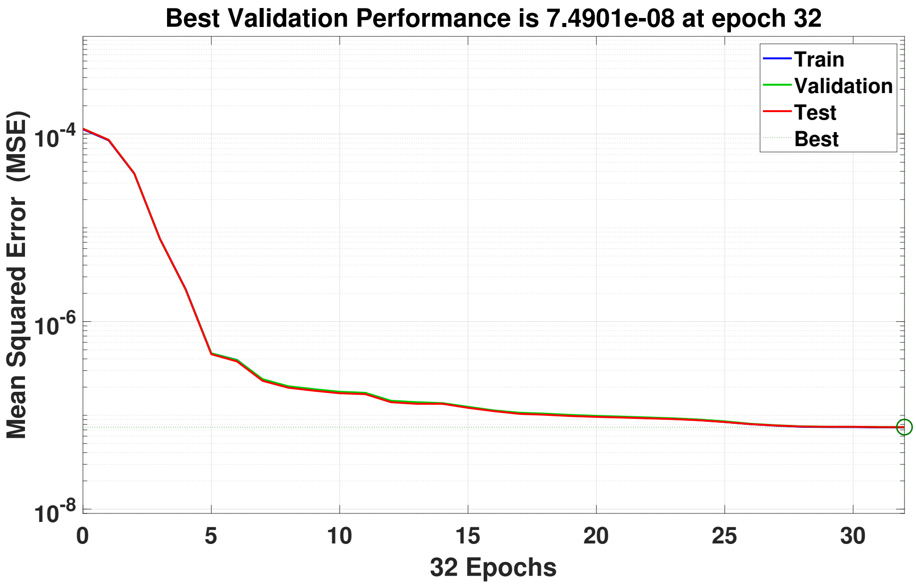

Figure 18. Validation performance of FSEV SOE soft sensor.

Figures 19 and 20 depict the composite plots of R-squared/FIT% and mean squared errors of the explored models respectively. The R-squared metric of ANN was once again superior to that of all other model structures. However, its mean squared error is significantly higher at 7.4276 and was not included in Figure 20 to allow for the better visual comparison of the MSE of the remaining models. Figure 20 shows that system identification-based parametric models achieve lower mean squared errors than that of their regression counterparts.

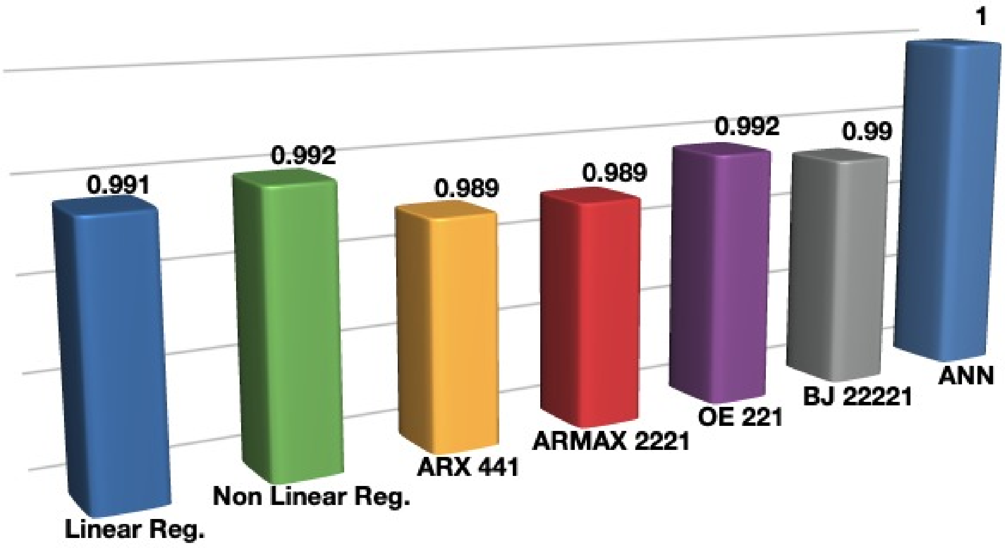

Figure 19. SOE model prediction accuracy: R squared values (linear regression, nonlinear regression, ANN soft sensor) and FIT% (ARX, ARMAX, OE, BJ).

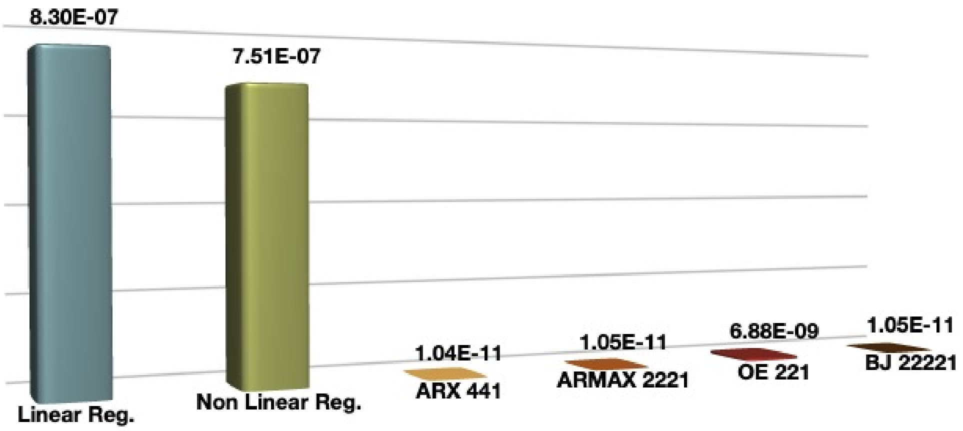

Figure 20. SOE models mean squared errors (MSE).

3.3. Power-Loss Soft Sensor

Table 12 shows the final input weights, output weights, and biases obtained for the two-layer feed-forward FSEV PL soft sensor design. Table 13 shows the details of the R-squared (0.9999) and MSE values for the training (MSE 5.74×10^{-3}), validation (MSE 5.76×10^{-3}), and testing (5.55×10^{-3}) sets of the FSEV PL data. The FSEV PL soft sensor attained higher R-squared metrics over the SOC as well as the SOE soft sensors. Its MSE values fared better than those of the SOE neural network but poorer in comparison to the SOC soft sensor. Figure 21 shows the PL soft-sensor regression plots under the various dataset partitions. The y label of the test regression plot indicates the following relation between output (predicted) and target (derived) PL values.

$$\text{Output (predicted PL)} = \text{Target (derived PL)} + 4 \times 10^{-4} \tag{23}$$

Table 12. Input and output weights and biases for two-layer feed-forward FSEV PL soft sensor.

Weight	Value	Weight	Value	Weight	Value	Bias	Value
$W_{1,1}$	15.2388	$W_{2,1}$	3.0079	$W_{o,1}$	−0.0034	b_1	−12.3583
$W_{1,2}$	−0.6052	$W_{2,2}$	1.0072	$W_{o,2}$	−0.0947	b_2	1.3888
$W_{1,3}$	2.1241	$W_{2,3}$	4.8293	$W_{o,3}$	−0.0131	b_3	−3.1843
$W_{1,4}$	0.7149	$W_{2,4}$	−1.4651	$W_{o,4}$	−6.1720	b_4	3.5928
$W_{1,5}$	0.0627	$W_{2,5}$	−0.935	$W_{o,5}$	−1.7267	b_5	0.7731
$W_{1,6}$	−7.2401	$W_{2,6}$	−1.1123	$W_{o,6}$	−0.0037	b_6	0.0887
$W_{1,7}$	−24.7311	$W_{2,7}$	−11.2042	$W_{o,7}$	−0.0027	b_7	−13.0898
$W_{1,8}$	4.5421	$W_{2,8}$	−0.9068	$W_{o,8}$	1.2184	b_8	2.7065
$W_{1,9}$	4.5675	$W_{2,9}$	−0.9336	$W_{o,9}$	−1.1994	b_9	2.7071
$W_{1,10}$	0.2263	$W_{2,10}$	1.8252	$W_{o,10}$	−0.3606	b_{10}	2.2972
						b_{o1}	7.0070

Table 13. FSEV PL soft sensor performance.

Dataset Partitions	MSE	R-sq
Training	5.74×10^{-3}	0.9999
Validation	5.76×10^{-3}	0.9999
Testing	5.55×10^{-3}	0.9999
Overall	-	0.9999

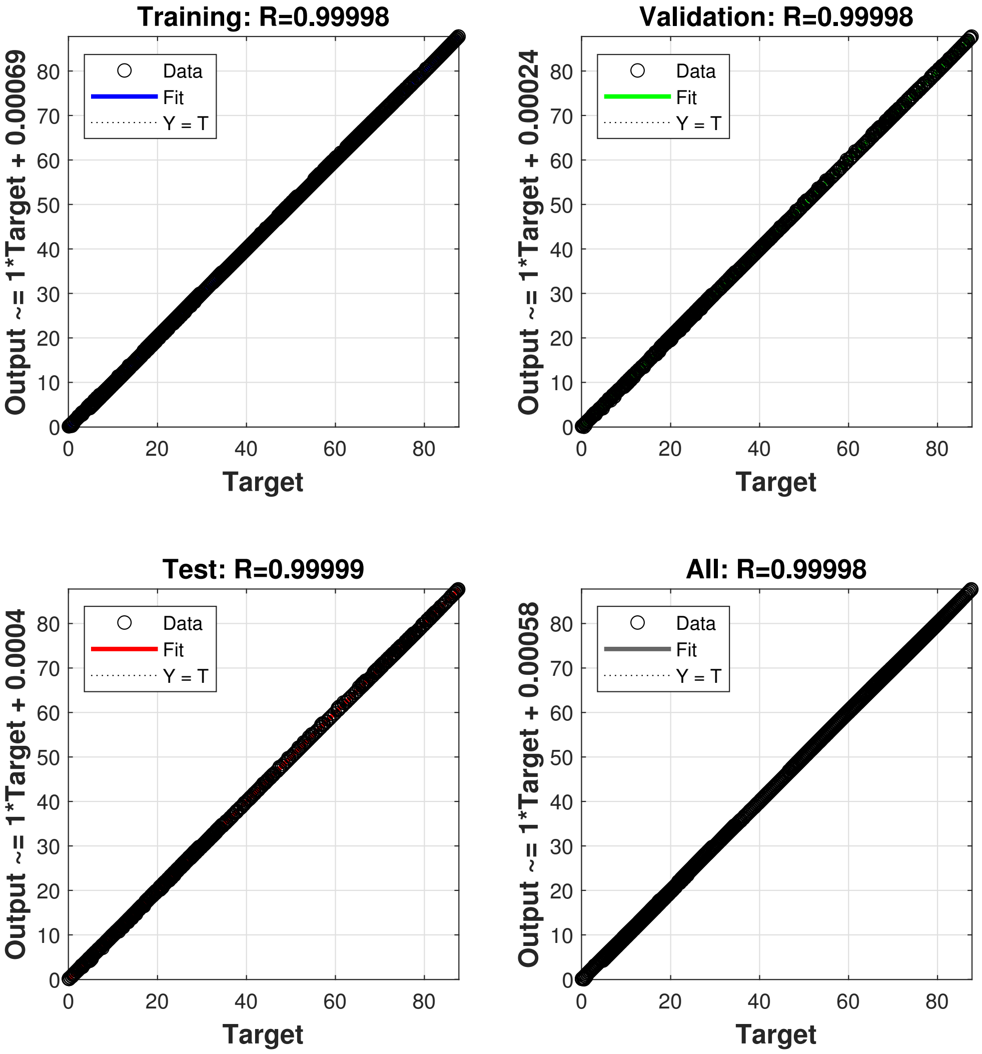

Figure 21. Regression plot of FSEV PL soft sensor.

Figure 22 shows the extremely nonlinear behaviour of the PL (watts) spikes experienced by the FSEV during one lap of the designated track. This figure depicts the dynamic effects of charging and discharging cycles of the FSEV battery on PL characteristics. In spite of the depicted power fluctuations, the proposed neural-network-based PL soft sensor can accurately follow the actual output throughout the lap. Figure 23 shows the error histogram of the data predicted by the PL soft sensor. It depicts the maximal number of testing data instances (around 10,000) at the error bin of −0.01007 followed by comparatively lower instances at 0.03951 (around 4000) and −0.05965 (approximately 2500) bins. The remaining error bins (0.08909, 0.1387, −0.1092 and others) contained lesser instances. The error histogram shows the concentration of a majority of testing error instances at low magnitudes contributing towards the almost perfect predictability of the proposed FSEV PL soft-sensor design. Figure 24 presents the training state indicators for the PL soft-sensor generation. In this case, the training algorithm is terminated at epoch 211 because of reaching the maximal number (6) of validation fails permissible (Table 6). The gradient and mu at epoch 211 were 0.0034063 and 0.0001, respectively. Figure 25 shows that the best validation performance of the proposed PL soft sensor was recorded at 0.00057628, attained at epoch 205. This value was comparatively higher than the best validation performance metrics of the SOC and SOE soft sensors.

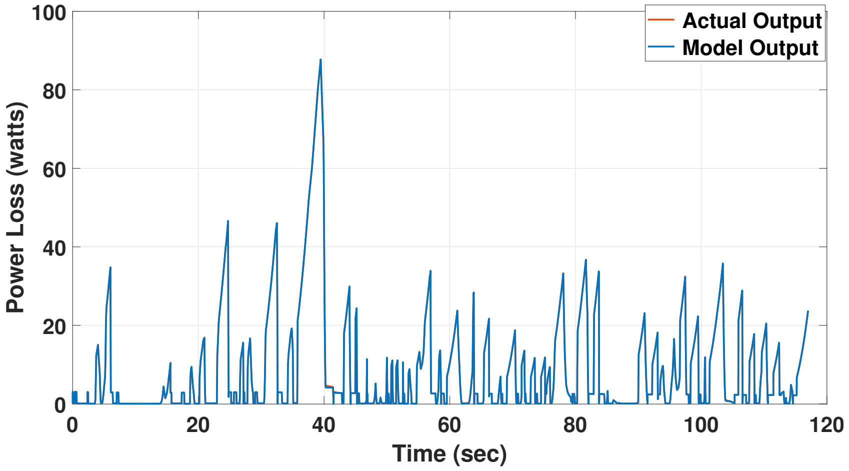

Figure 22. Derived versus soft-sensor outputs of FSEV PL.

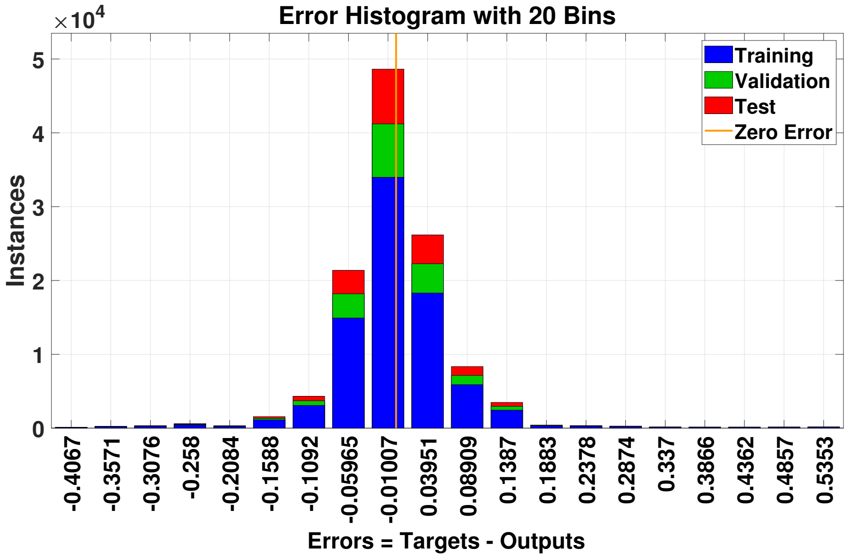

Figure 23. Error histogram of FSEV PL soft sensor.

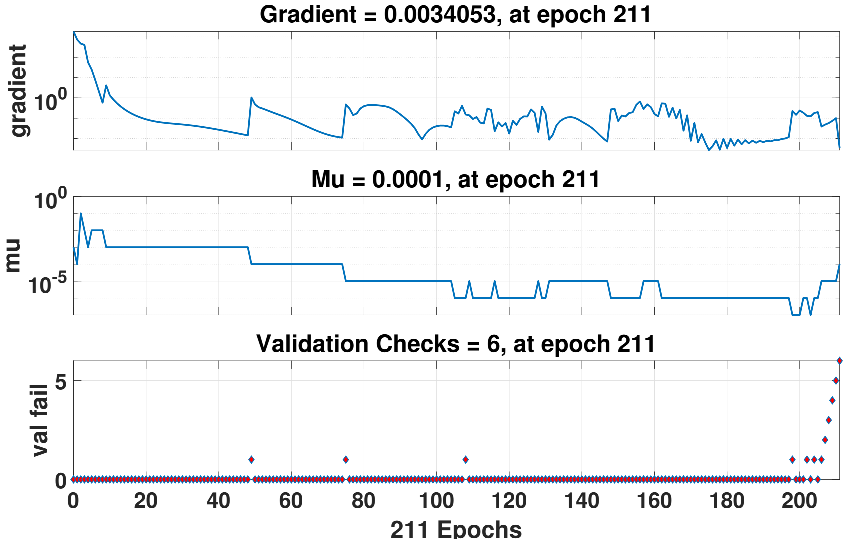

Figure 24. Training states of FSEV PL soft sensor.

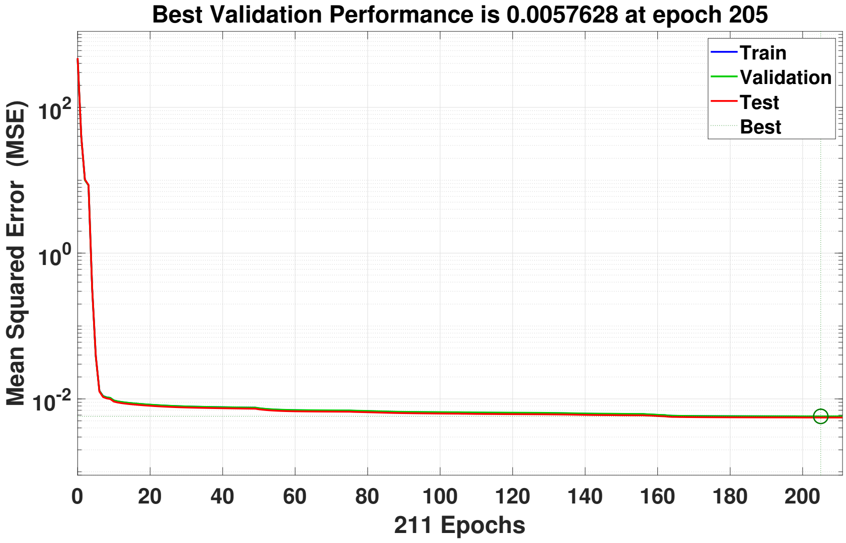

Figure 25. Validation performance of FSEV PL soft sensor.

Below are the regression and parametric model structures explored for the FSEV PL considered in the current work: Regression—Linear

$$\mathrm{PL} = 2.074 - 0.02694 * t + 0.3897 * I \tag{24}$$

Regression—Nonlinear

$$\mathrm{PL} = -0.04033 - 0.006566 * t + 0.05491 * I - 8.419 \times 10^{-5} * t^{2} - 0.0009587 * t * I + 0.006543 * I^{2} \tag{25}$$

ARX 441 *A(z)y(t)* = *B(z)u(t)* + *e(t)*

$$\begin{aligned} A(z) &= 1 - 1.345z^{-1} + 0.2108z^{-2} + 0.07419z^{-3} + 0.06137z^{-4} \\ B(z) &= -0.03105z^{-1} + 0.02019z^{-2} + 0.009579z^{-3} + 0.0006876z^{-4} \end{aligned}$$

ARMAX 2221 *A(z)y(t)* = *B(z)u(t)* + *C(z)e(t)*

$$\begin{aligned} A(z) &= 1 - 1.818z^{-1} + 0.8182z^{-2} \\ B(z) &= -0.02764z^{-1} + 0.02738z^{-2} \\ C(z) &= 1 - 0.4853z^{-1} - 0.03206z^{-2} \end{aligned}$$

BJ 22221 *y(t)* = *[B(z)/F(z)]u(t)* + *[C(z)/D(z)]e(t)*

$$\begin{aligned} B(z) &= -0.03171z^{-1} + 0.03051z^{-2} \\ C(z) &= 1 - 0.4834z^{-1} - 0.04278z^{-2} \\ D(z) &= 1 - 1.823z^{-1} + 0.823z^{-2} \\ F(z) &= 1 - 1.718z^{-1} + 0.7183z^{-2} \end{aligned}$$

OE 221 *y(t)* = *[B(z)/F(z)]u(t)* + *e(t)*

$$\begin{aligned} B(z) &= 0.3303z^{-1} - 0.3212z^{-2} \\ F(z) &= 1 - 0.735z^{-1} - 0.2457z^{-2} \end{aligned}$$

Figures 26 and 27 show the R-squared/FIT% and mean squared errors of the FSEV PL models. The ANN and nonlinear regression PL models attained near perfect prediction accuracy. Linear regression and OE models achieved comparatively worse estimation performance. The mean squared error of the linear regression model was 47.7908 and is excluded from Figure 27 to allow for visual comparisons among the MSEs of the remaining models.

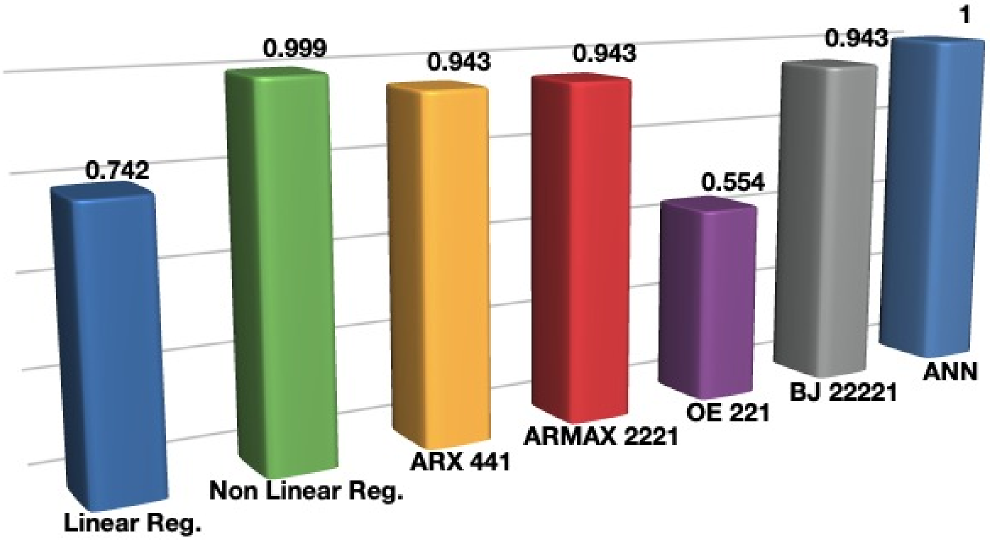

Figure 26. PL model prediction accuracy: R squared values (linear regression, non linear regression, ANN soft sensor) and FIT% (ARX, ARMAX, OE, BJ).

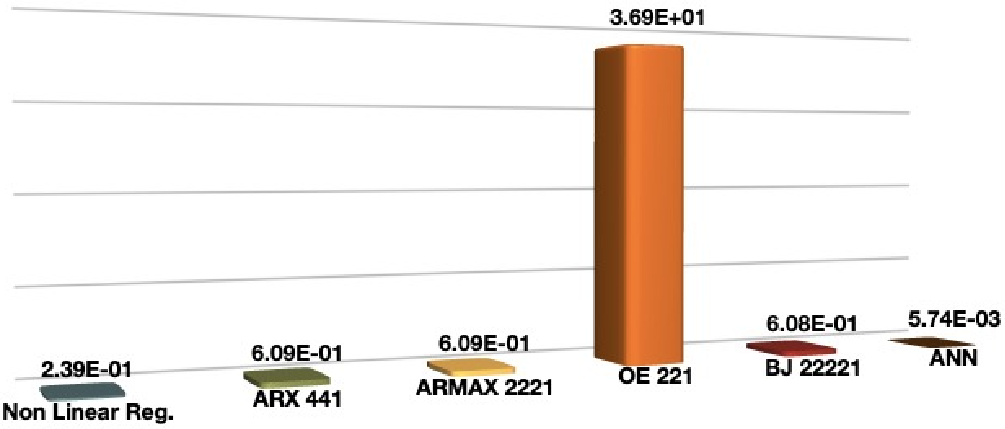

Figure 27. PL models mean squared errors (MSE).

4. Conclusions

In the current work, two-layer feed-forward neural networks were employed to develop soft sensors for estimations of SOC, SOE, and PL in an FSEV. The FSEV battery-pack system was first designed on the basis of the formula student competition guidelines. It included the construction of an FSEV powertrain model in MATLAB Simulink based on the designed vehicle and track parameters. The required input current profile for a single cell was obtained from the powertrain model and was utilised in conjunction with the cell parameters to derive the total PL over the circuit components. The total PL was used to yield the operating temperature and capacity; capacity was utilised to derive the SOC as per the current profile required by the powertrain model. Similarly, the SOE was determined using the input current profile and the terminal voltage (computed on the basis of the OCV). Lastly, the input current profile and derived SOC/SOE/PL datasets were loaded into separate two-layer feed-forward artificial neural-network-based soft-sensor designs. All developed soft sensors were tested for MSE and R-squared metrics of the dataset partitions; equations relating the derived and predicted outputs; error histograms of the training, validation and testing datasets; training state indicators such as gradient, mu and validation fails; validation performance curves over successive epochs and predicted versus derived plots over one lap time. Moreover, the performance of the proposed soft sensors was compared to predictions obtained by linear or nonlinear regression and the parametric models used in the system identification of complex systems. Below are the performance highlights of the proposed soft sensors.

1. Testing dataset accuracy of the proposed FSEV SOC, SOE, PL soft sensors was 99.96%, 99.96%, and 99.99%, respectively.
2. The MSEs of testing dataset partitions for the SOC, SOE, PL soft sensors were 8.34×10^{-8}, 7.4774 and 5.55×10^{-3}, respectively.
3. The R-squared prediction metrics of the proposed ANN-based soft sensors were superior to the R-squared values of the linear or nonlinear regression models and FIT% of the system identification-based parametric models.
4. The mean squared errors of the parametric model structures were lower than those of the ANN and linear or nonlinear regression models (except for the OE 221 PL parametric model).
5. The best validation performance (MSE) for the SOC, SOE, PL soft sensors was 8.3017×10^{-8} at epoch 25, 7.4901×10^{-8} at epoch 32, and 5.7628×10^{-3} at epoch 205, respectively.
6. The SOC and SOE dropped from 97% to 93.5% and 97% to 93.8%, respectively, during the FSEV running time of 118 s (one lap time).

Thus, two-layer feed-forward neural-network-based soft sensors can be effectively utilised to monitor and predict the SOC, SOE, and PL profile of an FSEV. The future scope of this work includes the modelling and estimation of other EV parameters such as regenerative power.

Author Contributions: Provided the EV cell data and reviewed the manuscript, S.P., R.F., C.S., M.F. and M.-K.T.; planned the study and wrote the manuscript, R.S.; worked on ANN based soft sensor design, P.S.; worked on Simulink modelling and battery pack design, K.P., S.S., V.N. and V.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data available for this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tran, M.K.; Akinsanya, M.; Panchal, S.; Fraser, R.; Fowler, M. Design of a Hybrid Electric Vehicle Powertrain for Performance Optimization Considering Various Powertrain Components and Configurations. *Vehicles* **2021**, *3*, 20–32. [CrossRef]
2. Pappalardo, C.; Lombardi, N.; Guida, D. A model-based system engineering approach for the virtual prototyping of an electric vehicle of class l7. *Eng. Lett.* **2020**, *28*, 215–234.
3. Vita, V.; Koumides, P. Electric vehicles and distribution networks: Analysis on vehicle to grid and renewable energy sources integration. In Proceedings of the 2019 11th Electrical Engineering Faculty Conference (BulEF), Varna, Bulgaria, 11–14 September 2019; pp. 1–4.
4. Sen, C.; Kar, N.C. Battery pack modeling for the analysis of battery management system of a hybrid electric vehicle. In Proceedings of the 2009 IEEE Vehicle Power and Propulsion Conference, Dearborn, MI, USA, 7–10 September 2009; pp. 207–212. [CrossRef]
5. Jaguemont, J.; Boulon, L.; Dubé, Y. Characterization and Modeling of a Hybrid-Electric-Vehicle Lithium-Ion Battery Pack at Low Temperatures. *IEEE Trans. Veh. Technol.* **2016**, *65*, 1–14. [CrossRef]
6. Cui, W.H.; Wang, J.S.; Chen, Y.Y. Equivalent circuit model of lead-acid battery in energy storage power station and its state-of-charge estimation based on extended kalman filtering method. *Eng. Lett.* **2018**, *26*, 504–517.

7. Li, J.; Mazzola, M.S. Accurate battery pack modeling for automotive applications. *J. Power Source* **2013**, *237*, 215–228. [CrossRef]
8. Liu, C.; Liu, W.; Wang, L.; Hu, G.; Ma, L.; Ren, B. A new method of modeling and state of charge estimation of the battery. *J. Power Sources* **2016**, *320*, 1–12. [CrossRef]
9. Ting, T.; Man, K.; Lei, C.U.; Lu, C. State-of-charge for battery management system via Kalman filter. *Eng. Lett.* **2014**, *22*, 75–82.
10. Shah, P.; Sekhar, R. Closed Loop System Identification of a DC Motor using Fractional Order Model. In Proceedings of the 2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE), Bali, Indonesia, 4–6 December 2019; pp. 69–74.
11. Sekhar, R.; Singh, T.; Shah, P. Complex Order PI$^{\alpha+j\beta}$D$^{\gamma+j\theta}$ Design for Surface Roughness Control in Machining CNT Al-Mg Hybrid Composites. *Adv. Sci. Technol. Eng. Syst. J. (ASTESJ)* **2020**, *5*, 299–306. [CrossRef]
12. Sekhar, R.; Singh, T. Determination of Johnson Cook Parameters in Turning of Micro and Nano Reinforced Aluminum Composites using Trust Region Reflective Algorithm. *Int. J. Innov. Technol. Explor. Eng.* **2019**, *8*, 1712–1716. [CrossRef]
13. Sekhar, R.; Shah, P. Predictive Modeling of a Flexible Robotic Arm using Cohort Intelligence Socio-Inspired Optimization. In Proceedings of the 2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE), Yogyakarta, Indonesia, 13–14 October 2020; pp. 193–198. [CrossRef]
14. Sekhar, R.; Singh, T.; Shah, P. System identification of tool chip interface friction while machining CNT-Mg-Al composites. *AIP Conf. Proc.* **2021**, *2317*, 1–9. [CrossRef]
15. Sekhar, R.; Singh, T.; Shah, P. ARX/ARMAX Modeling and Fractional Order Control of Surface Roughness in Turning Nano-Composites. In Proceedings of the 2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE), Bali, Indonesia, 4–6 December 2019; pp. 97–102. [CrossRef]
16. Sekhar, R.; Singh, T.; Shah, P. Micro and Nano Particle Composite Machining: Fractional Order Control of Surface Roughness. In Proceedings of the Third International Conference on Powder, Granule and Bulk Solids: Innovations and Applications PGBSIA 2020, Patiala, India, 26–28 February 2020; pp. 35–42.
17. Shah, P.; Sekhar, R. Predictive Modeling and Control of Clamp Load Loss in Bolted Joints Based on Fractional Calculus. In *Advances in Computing and Network Communications*; Thampi, S.M., Gelenbe, E., Atiquzzaman, M., Chaudhary, V., Li, K.C., Eds.; Springer: Singapore, 2021; pp. 15–32.
18. Shah, P.; Sekhar, R.; Singh, P. Predictive Modeling of A Bio-Fuelled Diesel Engine Using System Identification Approach. In Proceedings of the International Conference on Renewable Energy: Generation and Applications (ICREGA), Al Ain, United Arab Emirates, 2–4 February 2021; pp. 95–100. [CrossRef]
19. Sekhar, R.; Singh, T.; Shah, P. Machine learning based predictive modeling and control of surface roughness generation while machining micro boron carbide and carbon nanotube particle reinforced Al-Mg matrix composites. *Part. Sci. Technol.* **2021**, 1–18. [CrossRef]
20. Jatti, V.S.; Sekhar, R.; Shah, P. Machine Learning Based Predictive Modeling of Ball Nose End Milling using Exogeneous Autoregressive Moving Average Approach. In Proceedings of the 2021 IEEE 12th International Conference on Mechanical and Intelligent Manufacturing Technologies (ICMIMT), Cape Town, South Africa, 13–15 May 2021; pp. 68–72.
21. Shah, P.; Sekhar, R.; Kulkarni, A.J.; Siarry, P. *Metaheuristic Algorithms in Industry 4.0*; CRC Press: Boca Raton, FL, USA, 2021.
22. Li, C.; Xiao, F.; Fan, Y. An Approach to State of Charge Estimation of Lithium-Ion Batteries Based on Recurrent Neural Networks with Gated Recurrent Unit. *Energies* **2019**, *12*, 1592. [CrossRef]
23. Castano, S.; Gauchia, L.; Voncila, E.; Sanz, J. Dynamical modeling procedure of a Li-ion battery pack suitable for real-time applications. *Energy Convers. Manag.* **2015**, *92*, 396–405. [CrossRef]
24. Zhang, L.; Hu, X.; Wang, Z.; Sun, F.; Dorrell, D.G. Fractional-order modeling and State-of-Charge estimation for ultracapacitors. *J. Power Sources* **2016**, *314*, 28–34. [CrossRef]
25. Xiao, R.; Shen, J.; Li, X.; Yan, W.; Pan, E.; Chen, Z. Comparisons of Modeling and State of Charge Estimation for Lithium-Ion Battery Based on Fractional Order and Integral Order Methods. *Energies* **2016**, *9*, 184. [CrossRef]
26. Meng, J.; Luo, G.; Ricco, M.; Swierczynski, M.; Stroe, D.I.; Teodorescu, R. Overview of Lithium-Ion Battery Modeling Methods for State-of-Charge Estimation in Electrical Vehicles. *Appl. Sci.* **2018**, *8*, 659. [CrossRef]
27. He, W.; Williard, N.; Chen, C.; Pecht, M. State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation. *Int. J. Electr. Power Energy Syst.* **2014**, *62*, 783–791. [CrossRef]
28. Charkhgard, M.; Farrokhi, M. State-of-Charge Estimation for Lithium-Ion Batteries Using Neural Networks and EKF. *IEEE Trans. Ind. Electron.* **2010**, *57*, 4178–4187. [CrossRef]
29. Chemali, E.; Kollmeyer, P.J.; Preindl, M.; Emadi, A. State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach. *J. Power Source* **2018**, *400*, 242–255. [CrossRef]
30. Li, I.; Wang, W.; Su, S.; Lee, Y. A Merged Fuzzy Neural Network and Its Applications in Battery State-of-Charge Estimation. *IEEE Trans. Energy Convers.* **2007**, *22*, 697–708. [CrossRef]
31. Weigert, T.; Tian, Q.; Lian, K. State-of-charge prediction of batteries and battery–supercapacitor hybrids using artificial neural networks. *J. Power Source* **2011**, *196*, 4061–4066. [CrossRef]
32. Offer, G.J.; Yufit, V.; Howey, D.A.; Wu, B.; Brandon, N.P. Module design and fault diagnosis in electric vehicle batteries. *J. Power Source* **2012**, *206*, 383–392. [CrossRef]
33. Carraro, E.; Degano, M.; Morandin, M.; Bianchi, N. Formula SAE electric competition: Electrical motor design. In Proceedings of the 2013 International Electric Machines Drives Conference, Chicago, IL, USA, 12–15 May 2013; pp. 1142–1148. [CrossRef]

34. Munaro, M.; Bianchi, N.; Meneghetti, G. The Formula SAE Electric Vehicle Competition: A High-Torque Density Permanent Magnet Motor. *IEEE Ind. Appl. Mag.* **2020**, *26*, 76–86. [CrossRef]
35. Zhang, Y.; Xiong, R.; He, H.; Shen, W. Lithium-Ion Battery Pack State of Charge and State of Energy Estimation Algorithms Using a Hardware-in-the-Loop Validation. *IEEE Trans. Power Electron.* **2017**, *32*, 4421–4431. [CrossRef]
36. Zhang, W.; Shi, W.; Ma, Z. Adaptive unscented Kalman filter based state of energy and power capability estimation approach for lithium-ion battery. *J. Power Source* **2015**, *289*, 50–62. [CrossRef]
37. He, H.; Zhang, Y.; Xiong, R.; Wang, C. A novel Gaussian model based battery state estimation approach: State-of-Energy. *Appl. Energy* **2015**, *151*, 41–48. [CrossRef]
38. Zheng, L.; Zhu, J.; Wang, G.; He, T.; Wei, Y. Novel methods for estimating lithium-ion battery state of energy and maximum available energy. *Appl. Energy* **2016**, *178*, 1–8. [CrossRef]
39. Li, K.; Wei, F.; Tseng, K.J.; Soong, B.H. A Practical Lithium-Ion Battery Model for State of Energy and Voltage Responses Prediction Incorporating Temperature and Ageing Effects. *IEEE Trans. Ind. Electron.* **2018**, *65*, 6696–6708. [CrossRef]
40. Lin, C.; Mu, H.; Xiong, R.; Cao, J. Multi-model probabilities based state fusion estimation method of lithium-ion battery for electric vehicles: State-of-energy. *Appl. Energy* **2017**, *194*, 560–568. [CrossRef]
41. Zhang, X.; Wang, Y.; Wu, J.; Chen, Z. A novel method for lithium-ion battery state of energy and state of power estimation based on multi-time-scale filter. *Appl. Energy* **2018**, *216*, 442–451. [CrossRef]
42. Li, X.; Pan, K.; Fan, G.; Lu, R.; Zhu, C.; Rizzoni, G.; Canova, M. A physics-based fractional order model and state of energy estimation for lithium ion batteries. Part II: Parameter identification and state of energy estimation for LiFePO4 battery. *J. Power Source* **2017**, *367*, 202–213. [CrossRef]
43. Apostolaki-Iosifidou, E.; Codani, P.; Kempton, W. Measurement of power loss during electric vehicle charging and discharging. *Energy* **2017**, *127*, 730–742. [CrossRef]
44. Category Archives: Rules. Available online: https://www.formulabharat.com/category/rules/ (accessed on 6 August 2021).
45. Technical Data and Manual for EMRAX Motors/Generators. Available online: https://emrax.com/wp-content/uploads/2017/10/user_manual_for_emrax_motors.pdf (accessed on 6 August 2021).
46. Battery Pack Design Validation and Assembly Guide A123 Systems AMP20M1HD-A Nanophosphate Cell. Available online: https://www.buya123products.com/uploads/vipcase/b24d4f5b63934c59d43e93b3bb4db60a.pdf (accessed on 6 August 2021).
47. Panchal, S.; Mcgrory, J.; Kong, J.; Fraser, R.; Fowler, M.; Dincer, I.; Agelin-Chaab, M. Cycling degradation testing and analysis of a $LiFePO_4$ battery at actual conditions. *Int. J. Energy Res.* **2017**, *41*, 2565–2575. [CrossRef]
48. Saidani, F.; Hutter, F.X.; Scurtu, R.G.; Braunwarth, W.; Burghartz, J.N. Lithium-ion battery models: A comparative study and a model-based powerline communication. *Adv. Radio Sci.* **2017**, *15*, 83–91. [CrossRef]
49. Weber, T.; Sossenheimer, J.; Schäfer, S.; Ott, M.; Walther, J.; Abele, E. Machine Learning based System Identification Tool for data-based Energy and Resource Modeling and Simulation. In Proceedings of the Procedia CIRP, 26th CIRP Conference on Life Cycle Engineering (LCE) Purdue University, West Lafayette, IN, USA, 7–9 May 2019. [CrossRef]
50. Nguyen-Truong, H.T.; Le, H.M. An implementation of the Levenberg–Marquardt algorithm for simultaneous-energy-gradient fitting using two-layer feed-forward neural networks. *Chem. Phys. Lett.* **2015**, *629*, 40–45. [CrossRef]
51. El-Banbi, A.; Alzahabi, A.; El-Maraghi, A. *Artificial Neural Network Models for PVT Properties*; Gulf Professional Publishing: Oxford, UK, 2018; pp. 225–247. [CrossRef]
52. Jenkal, S.; Kourchi, M.; Rachdy, A.; Oussalem, O.; Ajaamoum, M.; Oubella, M. Modeling a photovoltaic emulator using four methods and buck-boost converter. *Eng. Lett.* **2021**, *29*, 408–415.
53. Dahunsi, O.; Pedro, J. Neural network-based identification and approximate predictive control of a servo-hydraulic vehicle suspension system. *Eng. Lett.* **2010**, *18*, 4.
54. Quesada, A. 5 Algorithms to Train a Neural Network. Available online: https://www.neuraldesigner.com/blog/5_algorithms_to_train_a_neural_network#Levenberg-Marquardt (accessed on 15 March 2021).
55. Mathworks, L. Levenberg-Marquardt Backpropagation. Available online: https://in.mathworks.com/help/deeplearning/ref/trainlm.html (accessed on 16 March 2021).
56. Paolucci, R. Linear Regression v.s. Neural Networks. 2020. Available online: https://towardsdatascience.com/linear-regression-v-s-neural-networks-cd03b29386d4 (accessed on 16 March 2021).