

PHYSICS-INFORMED REPRESENTATION LEARNING FOR HYBRID ELECTRIC VEHICLE ENERGY MANAGEMENT

NOUREDDINE DJEMAI¹, ALI ARIF¹, ABDERRAZAK GUETTA¹ AND TAREK BERGHOUT²

Keywords: Deep learning; Energy management optimization; Fuel cell hybrid electric vehicles; Long-short term memory; Machine learning.

Integrating physics-based and learning systems enhances fuel cell hybrid electric vehicles (FCHEVs) for better performance control and efficient power source operation. Balancing this diverse mix is challenging, given the uncertainties and fluctuations in complex physics-based modeling. In this framework, our work has a dual purpose. Firstly, precise physics-based modeling enables the effective generation of data. This helps gather diverse data resembling real-world scenarios, aiding in drawing reliable conclusions. Several known driving cycles were utilized to generate sufficient data for the experiments and findings presented in this work. Secondly, the collected data undergoes an advanced representation learning process with adaptive functions, enhancing the interaction between learning models and the FCHEV system's physical phenomena. The effectiveness of the suggested approach is validated through a comprehensive evaluation of developed algorithms using various visual and numerical metrics. In a comparative analysis, the results illustrate the efficacy of the methodology in addressing energy management (EM) challenges in fuel cell hybrid electric vehicles (FCHEVs).

1. INTRODUCTION

Efficient EM in FCHEVs is essential for sustainable operation, improving vehicle efficiency, extending range, and optimizing power. Modeling plays a vital role in the design of FCHEVs, enabling the evaluation of performance, efficiency, and reliability, and ensuring sustainable transportation [1,2]. FCHEVs gain from physics-based modeling, though their dynamic complexity challenges traditional mathematical models and optimization. Capturing nonlinear component behavior with uncertainties, such as FC, batteries (BT), and ultracapacitors (UC) is challenging. Integrating machine learning (ML) significantly enhances EM in FCHEV by analyzing real-world data, improving predictions, and understanding vehicle performance across diverse scenarios [3,4]. Integrating the two enhances EM by accounting for electrochemical reactions, temperature, humidity, and energy source characteristics [3]. This section reviews recent literature, identifies research gaps, highlights key contributions, and provides the work's structure.

1.1 RELATED WORKS ANALYSIS

This section reviews EM in FCHEVs using ML methods, summarized in Table 1. For instance, in [5], the authors used a predictive learning system for FCHEVs, forecasting velocity (v), power (P), and state of charge (SOC_{BT}). This optimizes performance and reduces power fluctuations through the use of long-short term memory (LSTM), a wavelet transform algorithm (WTA), and a rule-based strategy (RBS) [6]. The

dataset is derived from 8 standard driving cycles, namely, New York City Cycle (NYCC), Economic Commission for Europe (ECE), Highway Cycle (US06-HWY), California Highway Vehicle Route (CAHVR), United States Emissions Integrated Modified 240 (IM240), Representative Test Procedure 2005 (REP05), West Virginia University City Cycle (WVUCITY), and high-temperature (HL07). In [7], an Interconnection and Damping Assignment Passivity-Based Control (IDA-PBC) is introduced, considering hydrogen levels and SOC_{BT} under various conditions. An Artificial Neural Network (ANN) allocates P demand from different sources to enable constrained power dispatching. Real Driving Cycles of Tramway (RDCT) are used to evaluate the proposed system (see [7], Table 1). In [8], the authors proposed a Fuzzy Control Strategy (FCS) with Genetic Algorithm (GA) and ANN for FCHEVs health monitoring, considering driving cycles of diverse conditions: Highway Fuel Economy Test Cycle (HWFET), New European Driving Cycle (NEDC), and Urban Dynamometer Driving Schedule (UDDS). In addition, P_{Load} , SOC_{BT} , and state of health (SOH_{fc}) are used as inputs and P_{fc} outputs of the learning system.

A Deep Neural Network (DNN) control framework is presented in [9] for EM in FCHEVs. It combines a pseudo-spectral optimal controller (PSOC) and a DNN predictive controller. Two datasets are used: the Argonne National Laboratory (ANL) dataset and the Mobile Century project dataset (MCPD) to train two DNNs. In [10], the authors proposed an EM with Fuzzy Logic (FLC) and GA.

Table 1
Cutting-Edge EM-based machine learning in FCHEVs.

Ref.	Year	Methods	Learning features	Driving cycles
[5]	2020	LSTM, WTA, RBS.	Inputs: predicted velocity, SOC_{BT} , SOC_{UC} ; Outputs: P_{fc} , P_{BT} , P_{UC} , SOC_{BT} , SOC_{UC} ;	NYCC, ECE, US06 HWY, CAHVR, IM240, REP05, WVUCITY, HL07.
[7]	2021	IDA-PBC, ANN.	Inputs: SOC_{BT} , SOC_{UC} , FC n_{fc} ; Outputs: P_{fc} , P_{BT} , P_{UC} ;	RDCT
[8]	2021	FCS, GA, ANN.	Inputs: P_{Load} , SOC_{BT} , SOH_{fc} ; Outputs: P_{fc}	HWFET, NEDC, UDDS
[9]	2023	DNN, PSOC.	Inputs: v , SOC_{BT} , SOC_{UC} ; Outputs: P_{fc} , P_{BT} , P_{UC}	MCPD, ANL
[10]	2023	FCS, GA, K-means.	Inputs: P , SOC_{BT} , SOC_{UC} ; Outputs: P_{fc} .	NYCC, ECECOL, UDDS, HWY
This work	2024	LSTM, WDA. Outlier removals: Grubbs, Mahalanobisb, Euclidean, minkowski; Median filtering; Min-max scaling.	Inputs: P_m , P_{in} , P_{fc} , n_{fc} , v , V_{dc} , SOC_{UC} , SOC_{BT} ; Outputs: P_{BT} , P_{UC} , P_{load} , I_{fcconv} , I_{BTconv}	UDDS, NYCC, LA92, EUDCL, WLTC

¹ Department of Electrical Engineering, Laboratory of Modeling Energy Systems LMSE, University of Biskra, 07000 Biskra, Algeria.

² Laboratory of automation and manufacturing engineering, Batna2 university, 05000 Batna, Algeria.

Emails: noureddine.djemai@univ-biskra.dz, ali.arif@univ-biskra.dz, abderrazak.guettaf@univ-biskra.dz, t.berghout@univ-batna2.dz

The strategy reduces equivalent hydrogen consumption, enhancing fuel economy. The driving cycle clustering is based on the K-means method, utilizing the NYCC, ECECOL, UDDS, and HWY datasets.

1.2 RESEARCH GAPS

The literature review reveals data complexities and pattern variations in patterns within this field, suggesting potential research gaps.

Handling complexity and dynamics, investigating methods to enhance model adaptability to dynamic conditions, including vehicle changes and external performance factors, by developing algorithms that adjust to data variations.

Preprocessing, exploring advanced preprocessing methods beyond basic techniques like normalization, to improve learning model performance.

The evaluation utilizes comprehensive metrics to assess the accuracy and modeling ability, leveraging physics-based knowledge.

Addressing these gaps could contribute to the enhancement of EM in FCHEVs, offering valuable insights for researchers and practitioners.

1.3 KEY CONTRIBUTIONS AND OUTLINES

This work focuses on data preprocessing and advanced deep learning techniques as key strategies to address FCHEV complexities in EM. The main contributions are outlined below:

Data Preprocessing: A structured algorithmic pipeline is implemented to reduce complexity and enhance data quality through filtration, scaling, and outlier removal. This ensures that models are trained on high-relevance features, improving predictive accuracy and overall performance.

Deep Learning Framework: The integration of LSTM provides a powerful mechanism for capturing temporal dependencies. Its multi-layered architecture effectively identifies complex sequential patterns, making it well-suited for modeling dynamic EM optimization and accurately representing real-world driving scenarios.

Multi-Input and Multi-Output Learning: unlike previous studies, this work employs a multi-input, multi-output learning approach, as shown in Table 1. By incorporating a more extensive set of inputs and outputs, the model gains an enhanced ability to decode intricate relationships, significantly improving predictive performance and robustness.

This paper is structured into five sections: Section 2 details dataset generation, including driving cycles, physics-based models, and preprocessing techniques. Section 3 presents the deep learning methodologies used. Section 4 discusses the experimental setup, execution, and results, highlighting insights into EM optimization. Finally, section 5 summarizes key findings, implications, and future research directions.

2. MATERIALS

This section discusses the physics-based modeling process of FCHEVs, with an emphasis on dataset generation. It describes inputs and outputs for training the LSTM model and highlights data processing methods.

2.1 SYSTEM DESCRIPTION

In [11], a meticulously designed hybrid emergency power system supports aircraft during emergency landings, including a 12.5 kW liquid-cooled proton exchange

membrane fuel cell system, enhanced with battery and supercapacitor modules for better energy storage and management. The system features essential auxiliaries, a controller for steady power, a battery management system for optimizing performance and safety, and supercapacitor modules for rapid energy discharge. Specialized DC/DC converters manage power flow. An inverter converts DC to AC power, and programmable loads simulate various electrical demands. The system is monitored and controlled via sensors and signal conditioning units, providing real-time data acquisition and stability, offering a practical framework for studying FCHEV. The physics-based model focuses on proton exchange membrane cells suitable for automotive applications. Implemented in MATLAB/Simulink, the model simplifies by neglecting reactant flow dynamics and calculating voltage by considering various losses. The model's accuracy is confirmed through comparisons with actual fuel cell output voltages, showing a low error margin.

2.2 DATA GENERATION

This study uses a systematic approach to generate a comprehensive database for representation learning. Various driving cycles (UDDS, NYCC, LA92, EUDCL, WLTC) are collected to cover real-world scenarios (Fig. 1). These cycles precisely simulate common commuter behavior, traffic conditions, and peak power demand situations, testing the vehicle's EM system under high-load conditions. These individual cycles are then concatenated to form a singular, extensive database that captures the richness and variability inherent in different driving conditions.

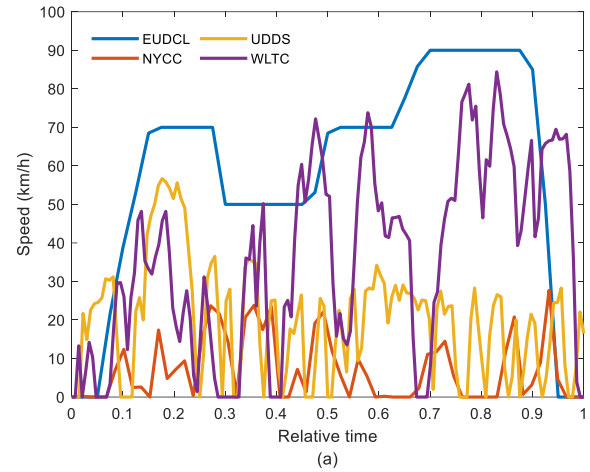


Fig.1 – Utilizing diverse driving cycles for comprehensive data generation.

The dataset inputs depicted in Fig. 2(a) encompass motor power P_m , input power P_{in} , SOC_{BT} , UC state of charge SOC_{UC} , FC power P_{fc} , hydrogen gas quantities n_{fc} , vehicle speed v DC-link voltage V_{dc} , DC-link reference voltage $V_{dc_{ref}}$. The outputs depicted in Fig. 2(b) include, P_{BT} , P_{UC} , P_{load} , FC converter current $I_{fc_{conv}}$, and $I_{BT_{conv}}$. By integrating diverse inputs and outputs, data generation methodology ensures a robust and representative dataset. Fig. 2(a, b) illustrates the complex interplay between components, showcasing the system's adaptive response and highlighting data drift issues. The dataset underscores challenges and opportunities in optimizing FCHEV for real-world applications, emphasizing complexity and variability in hybrid systems.

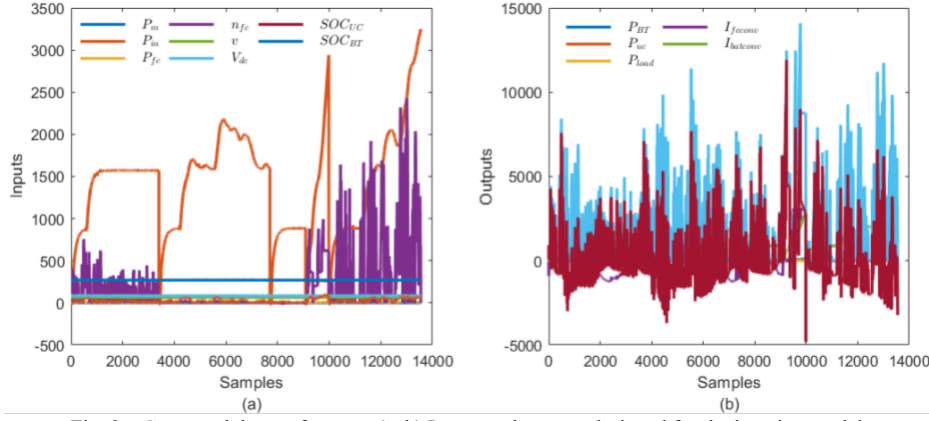


Fig. 2 – Generated dataset features; (a, b) Inputs and outputs designed for the learning model.

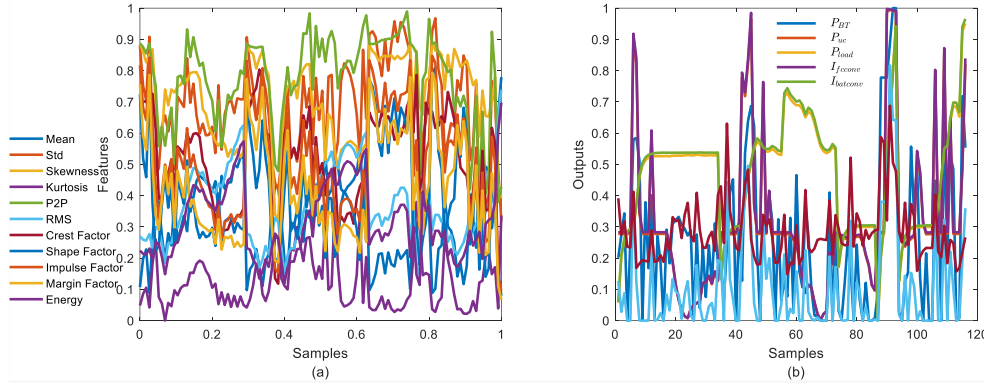


Fig. 3 – Data preprocessing results: a) inputs (extracted features); b) outputs.

Table 2
Parameters of data preprocessing.

Method	Parameters
Data scaling	Interval: [0,1].
Feature extraction	Mean, standard deviation (std), Skewness, Kurtosis, Peak two Peak (P2P), Root Mean Squares (RMS), Crest Factor, Shape Factor, Impulse Factor, Margin Factor, Energy.
Denoising	Type of wavelet: symlets; Denoising method: Bayes method; Level of wavelet decomposition: $\log_2 N$;
Outliers	Method for detecting outliers: Grubbs statistics; Moving method: Moving average median; Operating dimension: 1; Minimum outlier count: 10%.

2.3 DATA PROCESSING

Inspired by [12], this subsection employs a systematic four-layer approach to enhance driving cycles data for EM in FCHEV. The process includes scaling, feature extraction, denoising, and outlier removal. Scaling mitigates noise and normalizes data using moving average, median filtering, and min-max normalization. Feature extraction derives critical time and frequency domain features to distinguish operational states/patterns. Denoising uses a wavelet-based approach, specifically an empirical Bayesian method with a Cauchy prior, to refine signal quality and eliminate noise. Outlier removal incorporates statistical tests and distance measurements to identify and remove anomalies in the dynamic FCHEV environment. Table 2 summarizes key parameters of the employed algorithms, ensuring comprehensive and precise data processing. It is noteworthy to mention that preprocessing steps are also applied to outputs depicted in Fig. 2(b). Outlier instances and their associated ground truths are excluded in this process,

enhancing overall data quality. The results in Fig. 3 demonstrate marked smoothness and discernible patterns compared to the initial representation in Fig. 2(a, b), which had numerous outliers, noise, and randomness. This refined data contributes to the effectiveness of deep learning algorithms.

3. METHODS

LSTM is a type of deep recurrent neural network specifically designed to handle sequential data, where the concept of time is crucial. The hidden layer of LSTM includes three key gates: the input gate g_t^i , the output gate g_t^o , and the forget gate g_t^f , as shown in (1-4). These gates allow the network to effectively adapt and learn over time sequences. The hidden state h_t and cell state C_t are computed using (5-6), which take into account the inputs x_t , weights (w_f, w_i, w_h, w_c) and biases (b_i, b_o, b_h, b_c). These components work together to capture and retain relevant information throughout the sequence. The output O_t , as defined in equation (7), is calculated by applying output weights w_{oh} , output biases b_o , and an activation function f . The activation function f combines a sigmoid function and the hyperbolic tangent ($\tanh$), introducing non-linearity and aiding in feature extraction, which is critical for the model's ability to learn complex patterns.

$$g_t^f = f(w_f [h_{t-1} + x_t] + b_f), \quad (1)$$

$$g_t^i = f(w_i [h_{t-1} + x_t] + b_i), \quad (2)$$

$$g_t^o = f(w_o [h_{t-1} + x_t] + b_o), \quad (3)$$

$$h_t = f(w_h [h_{t-1} + x_t] + b_h), \quad (4)$$

$$\check{C}_t = \tanh(w_c [h_{t-1} + x_t] + b_c), \quad (5)$$

$$C_t = g_t^f C_{t-1} + g_t^i \check{C}_{t-1}, \quad (6)$$

$$O_t = f(w_{oh} h_t + b_o). \quad (7)$$

LSTM layer has been configured with specific parameters to optimize its performance in this study. The LSTM layer consists of 20 neurons, a configuration that allows for capturing temporal dependencies in the data effectively. Training of this network is conducted using the Adaptive Moment Estimation algorithm (*i.e.*, Adam optimization), chosen for its efficiency in handling sparse gradients and adapting the learning rate during training. The training process is configured with a maximum of 150 epochs and a mini-batch size of 20, balancing the need for computational efficiency with the accuracy of gradient estimation. The initial learning rate is set to 0.01, a standard value for starting the training process, and it's adjusted as needed during training. A gradient threshold of 1 is applied to prevent the gradients from exploding, ensuring stable training progress. The L_2 regularization parameter is set to 0.0001, introducing a small amount of regularization to avoid overfitting while maintaining the flexibility of the model. The network employs a 3-fold cross-validation technique, enhancing the reliability of its performance evaluation by splitting the dataset into three parts, using each in turn for validation while training on the remaining data. This approach ensures a comprehensive assessment of the model's generalization capabilities. Moreover, the training is conducted solely on a CPU, which, while potentially slower than GPU-based training, offers broader compatibility with different computational environments. Validation data is explicitly provided for assessing the model's performance during training, ensuring that the learning process is guided by not just training data performance but also by how well the model generalizes to unseen data. Notably, the order of the data is preserved during training, which could be crucial for time-series data where temporal sequence integrity is important. Streamlining the output for efficiency or integration purposes. Overall, this LSTM neural network architecture, with its specific configuration and the use of 3-fold cross-validation, is tailored to capture complex temporal patterns in data while ensuring robustness and generalization capabilities, making it well-suited for tasks requiring a nuanced understanding of time-series data.

4. RESULTS AND DISCUSSION

In this work, in addition to the single LSTM layer neural network, the study also involves a comparative analysis with several other neural network architectures to evaluate their respective performances. These architectures include a Gated Recurrent Unit (GRU) network, a Bidirectional LSTM (BiLSTM) network, and a Single Hidden Layer Feedforward Network (SLFN) [13] as per the contribution “*Integration of conventional machine learning methods*” in section 1.3. GRU combines the properties of GRUs and LSTMs, aiming to leverage the strengths of both in capturing temporal dependencies, potentially offering improved efficiency in learning long-range dependencies. The BiLSTM, on the other hand, extends the traditional LSTM by processing the data in both forward and backward directions, thus providing a more comprehensive understanding of the context in sequence data. This bidirectional approach is particularly beneficial for tasks where the context from both past and future data points is

crucial for accurate predictions [13]. Lastly, the SLFN, with its simpler architecture consisting of a single hidden layer, serves as a baseline for comparison. While less complex and possibly less powerful in capturing complex patterns compared to LSTM-based models, the SLFN's performance in this study provides valuable insights into the necessity and efficacy of more complex recurrent neural networks for the specific application at hand. This comparative study aims to assess the trade-offs between these different architectures in terms of learning capabilities, computational efficiency, and overall effectiveness in the given context.

In the comparative analysis, a comprehensive set of metrics is utilized to evaluate and contrast the performance of each model. These metrics include Root Mean Square Error (RMSE), Mean Squared Error (MSE), Mean Absolute Error (MAE), and the coefficient of determination, denoted as R^2 . RMSE provides a measure of the differences between values predicted by the models and the actual values, offering insights into the models' prediction accuracy. MSE, similar to RMSE, quantifies the average squared differences between the predicted and actual values, while MAE measures the average magnitude of errors in a set of predictions, without considering their direction. R^2 , on the flip side, signifies the proportion of the variance in the dependent variable that can be predicted from the independent variables. It serves as a measure of how well the model replicates the observed outcomes. To further enhance the robustness of the comparison, the standard deviation of RMSE, MSE, and MAE across each fold of the 3-fold cross-validation process (σ_m) was calculated. This approach offers valuable insights into the models' consistency and reliability across various data subsets. Furthermore, the standard deviation of R^2 for each of the 3-fold models (σ_R) was calculated to highlight the variability in the models' capacity to explain data variance across folds. Including these standard deviations enhances the analysis, providing a more thorough understanding of the models' performance stability and robustness, leading to a more detailed and nuanced comparison.

Table 3

3-fold cross-validation results for the LSTM network.

Folds	RMSE	MSE	MAE	R^2
1	0.1086	0.0117	0.0801	0.7679
2	0.1117	0.0124	0.0805	0.7639
3	0.1089	0.0118	0.08275	0.7816
μ	0.1097	0.01205	0.0811	0.7711
σ	0.0435	0.0435	0.0435	0.0092

Table 4

Comparison results.

Methods	RMSE	MSE	MAE	R^2
LSTM	0.1097	0.01205	0.0811	0.7711
Bi-LSTM	0.1123	0.0127	0.0833	0.7560
GRU	0.1261	0.0160	0.0950	0.6968
SLFN	0.1740	0.0307	0.1349	0.4197

This thorough evaluation strategy allows for a well-rounded assessment of each architecture's predictive capabilities and effectiveness in the context of the study.

Table 3 introduces first 3-fold cross validations results related to LSTM network. The results show a consistent performance of LSTM model across the folds, with relatively close values for RMSE, MSE, and MAE, indicating stability in the model's error metrics. The R^2 values, which measure the proportion of variance explained by the model, are also consistent across the folds, with a mean value of $\mu = 0.77$ suggesting a good fit to the data. Interestingly, the standard

deviation for RMSE, MSE, and MAE is the same 0.43×10^{-1} , indicating uniform variability in these error metrics across the folds. The standard deviation for R^2 is relatively low 0.9×10^{-2} , suggesting less variability in the model's explanatory power across different subsets of the data. This uniformity in the standard deviations of RMSE, MSE, and MAE is somewhat unusual and might warrant further investigation to understand the underlying factors contributing to this pattern.

Table 4 demonstrates the averaged results of a 3-fold cross-validation related to the compared models. In the comparative analysis of neural network models, the LSTM demonstrated superior performance with the lowest RMSE (0.1098), MSE (0.0121), and MAE (0.0812), and the highest R^2 value (0.7712), indicating its high accuracy and predictive capability. The BiLSTM followed closely, showing slightly higher errors (RMSE: 0.1123, MSE: 0.0128, MAE: 0.0833) and a marginally lower R^2 (0.7560), still maintaining commendable performance. GRU performance dropped notably, with higher error rates (RMSE: 0.1261, MSE: 0.0160, MAE: 0.0950) and a significantly reduced ability to predict variance in the data, as reflected by its R^2 value (0.6968). The SLFN lagged behind the others, registering the highest errors (RMSE: 0.1741, MSE: 0.0307, MAE: 0.1349) and the lowest R^2 (0.4198), indicating its relatively poorer fit and predictive accuracy. This table underscores LSTM's effectiveness in handling the dataset's complexities, with BiLSTM also showing good potential, while GRU and especially SLFN exhibited lower performance metrics.

Finally, the LSTM performance, as indicated by the provided loss values in Fig. 4, shows a positive trend. In the training phase depicted by Fig. 4(a), the loss values show a notable downward trend. For instance, the initial losses in the first fold start at 0.073

and decrease steadily to 0.0040 by the final iteration. This significant reduction reflects the model's capacity to learn from the training data effectively. Similarly, in the second and third folds, we observe a decrease from initial values of 0.0295 and 0.0319 to 0.0037 and 0.0035, respectively. These patterns are indicative of consistent learning across all folds. The testing phase, as depicted in Fig. 4(b), evaluates the model's efficacy on previously unseen data. This evaluation also indicates a reduction in loss values, though this decrease is less pronounced compared to that observed during the training phase. For example, in the first fold, the loss decreases from an initial value of 0.0423 to 0.0060 towards the end. In the second and third folds, the initial losses of 0.09643 and 0.0714 reduce to 0.00660 and 0.0060, respectively, by the end of the process. While these decreases are encouraging, the generally higher loss values in testing compared to training are typical and highlight the challenges models face when generalizing to new data. The consistency in the decreasing trend of loss values across all folds in both training and testing phases is a positive indicator. However, the fact that the testing losses do not converge as closely to the training losses as one might hope could suggest areas for model improvement, possibly through more sophisticated regularization techniques or additional training data. Overall, the model demonstrates effective learning and a reasonable degree of generalization. The gradual reduction in loss values, both in training and testing, underscores the model's ability to adapt to the data it is trained on and its competence in handling new, unseen data. Overall, LSTM's performance in this study demonstrates its robustness in handling complex data sequences. Its ability to capture temporal dependencies is evident from its superior metrics, particularly its low error rates (RMSE, MSE, MAE) and high R^2 value, indicating high predictive accuracy and model fit.

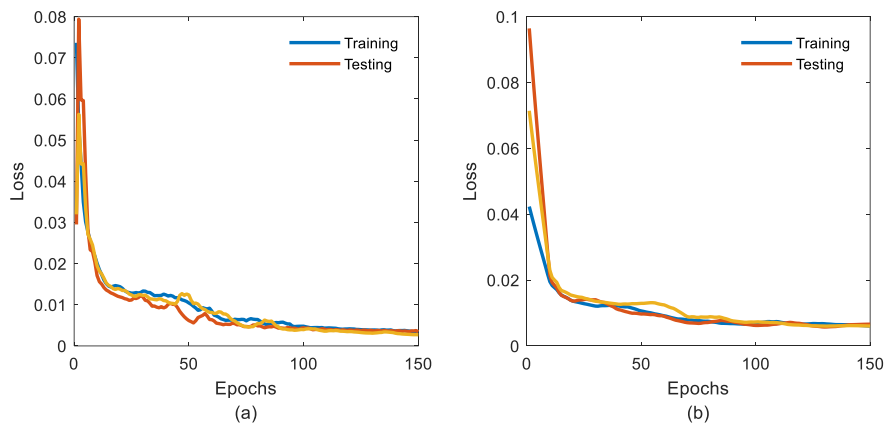


Fig. 4 – (a, b) Loss function behaviors during training and testing, respectively.

The results are consistent across cross-validation and suggest that LSTM is not prone to overfitting, maintaining its performance across different data subsets. This stability is a crucial aspect of reliable predictive modeling. Compared to other architectures, LSTM stands out for its ability to effectively process and learn from data. It underscores its suitability for complex sequence modeling tasks, making it a valuable tool. However, while reported error values provide a broad assessment of model performance, it is important to analyze error behavior under different operational scenarios. A more detailed investigation into specific conditions, such as high-speed driving, varying load conditions, and transient acceleration phases, could reveal potential weaknesses in prediction capability.

Additionally, while cross-validation ensures consistency across different data subsets, it does not fully capture how generalization applies to entirely new driving cycles that were not included in the training. Future work should explore testing the model on unseen driving conditions to assess its adaptability to novel scenarios, ensuring robustness beyond the evaluated dataset. For instance, evaluations on diverse real-world cycles, including aggressive or heavy conditions, could provide deeper insights into potential limitations. Extending this comparison to state-of-the-art methods (refer to Table 1), this study is distinguished by considering a significantly larger number of outputs (five), unlike most others, except for the method outlined in references [7], [9], which considered three outputs. This ability to handle multiple outputs is a challenge and a

testament to sophistication and adaptability. The ability to process and predict multiple outcomes offers an understanding of data dynamics, which is beneficial in complex predictive scenarios. This aspect further demonstrates LSTM's versatility and its edge over traditional methods that are typically limited in the scope of output handling [14].

4. CONCLUSION

The paper presents an approach to EM optimization in FCHEV through representation learning. It begins with a robust data generation methodology, creating a representative dataset capturing the dynamic nature of FCHEVs under various driving conditions. Afterward, systematic data processing is employed to refine driving cycle data, tackling challenges in FCHEV energy management. A significant part of the study is the use of LSTM. Their careful architecture and a 3-fold cross-validation technique effectively capture complex temporal patterns in data. The paper also includes a comparative analysis with other neural network architectures, underscoring the LSTM model's superior performance in predictive accuracy. Lower RMSE, MSE, and MAE values and a higher R^2 value evidence this. The BiLSTM also shows commendable performance, while the GRU and SLFN lag behind, highlighting the importance of sophisticated recurrent neural networks for this specific application. An essential aspect of the study is the consistency in the standard deviations across the folds for LSTM metrics, which adds to the model's stability and reliability. Additionally, LSTM handles a significantly larger number of outputs, showcasing the model's advanced adaptability and sophistication. In conclusion, this research provides valuable insights into using LSTM networks for FCHEV energy optimization, particularly in handling complex datasets, and paves the way for further advancements in representation learning.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Noureddine Djemai: conceptualization, methodology, investigation, writing – original draft.
 Ali Arif: Methodology, data curation, writing – review & editing.
 Abderrazak Guettaf: writing – review & editing, validation.
 Tarek Berghout: conceptualization, methodology, investigation, writing – original draft – review & editing.

Received on 30 April 2024

REFERENCES

1. M. Rasool, M.A. Khan, R. Zou, *A comprehensive analysis of online and offline energy management approaches for optimal performance of fuel cell hybrid electric vehicles*, *Energies*, **16**, 8, pp. 1–33 (2023).
2. Y. Cao, M. Yao, X. Sun, *An overview of modelling and energy management strategies for hybrid electric vehicles*, *Applied Sciences*, **13**, 10, pp. 1–23 (2023).
3. M. Fayyazi et al., *Artificial intelligence/machine learning in energy management systems, control, and optimization of hydrogen fuel cell vehicles*, *Sustainability*, **15**, 6, pp. 1–38 (2023).
4. A.F. Glavan, V. Croitoru, *Incremental learning for edge network intrusion detection*, *Rev. Roum. Sci. Techn. – Électrotechn. Et Énerg.*, **8**, 3, pp. 301–306 (2023).
5. J. Wang, J. Zhou, D. Xu, *A real-time predictive energy management strategy of fuel cell/battery/ultra-capacitor hybrid energy storage system in electric vehicle*, Chinese Automation Congress, Institute of Electrical and Electronics Engineers Inc (2020).
6. V. Hraniak, *Using discrete wavelet analysis of vibration signal for detection of electrical machines' defects*, *Rev. Roum. Sci. Techn. – Électrotechn. Et Énerg.*, **68**, 4, pp. 357–362 (2023).
7. A. Benmouna, M. Becherif, L. Boulon, C. Dépature, H.S. Ramadan, *Efficient experimental energy management operating for FC/battery/SC vehicles via hybrid artificial neural networks-passivity based control*, *Renew. Energy*, **178**, pp. 1291–1302 (2021).
8. X. Hu, S. Liu, K. Song, Y. Gao, T. Zhang, *Novel fuzzy control energy management strategy for fuel cell hybrid electric vehicles considering state of health*, *Energies*, **14**, 20, pp. 1–20 (2021).
9. S.M. Sotoudeh, B. HomChaudhuri, *A deep-learning-based approach to eco-driving-based energy management of hybrid electric vehicles*, *IEEE Trans. Transp. Electrification*, **9**, 3, pp. 3742–3752 (2023).
10. Y. Wang et al., *Genetic algorithm-based Fuzzy optimization of energy management strategy for fuel cell vehicles considering driving cycles recognition*, *Energy*, **263**, pp. 126112 (2023).
11. S.N. Motapon, L.A. Dessaint, K. Al-Haddad, *A comparative study of energy management schemes for a fuel-cell hybrid emergency power system of more-electric aircraft*, *IEEE Trans. Ind. Electron.*, **61**, 3, pp. 1320–1334 (2014).
12. T. Berghout, T. Bentrucia, W.H. Lim, M. Benbouzid, *A neural network weights initialization approach for diagnosing real aircraft engine inter-shaft bearing faults*, *Machines*, pp. 1–13 (2023).
13. Y. Yu, X. Si, C. Hu, J. Zhang, *A review of recurrent neural networks: LSTM cells and network architectures*, *Neural Comput.*, **31**, 7, pp. 1235–1270 (2019).
14. B. Ponnuswamy, C. Columbus, S.R. Lakshmi, J. Chithambaram, *Wind turbine fault modeling and classification using cuckoo-optimized modular neural networks*, *Rev. Roum. Sci. Techn. – Électrotechn. Et Énerg.*, **68**, 4, pp. 369–374 (2023).