Load Prediction Based Remaining Discharge Energy Estimation Using a Combined Online and Offline Prediction Framework

Ollie Hatherall*, James Marco†, Anup Barai‡, Mona Faraji Niri§
Warwick Manufacturing Group, University of Warwick, Coventry, United Kingdom, CV4 7AL
*Email: Ollie.Hatherall@warwick.ac.uk
†Email: James.Marco@warwick.ac.uk
‡Email: A.Barai@warwick.ac.uk
§Email: Mona.Faraji-Niri@warwick.ac.uk

***Abstract*— Remaining discharge energy (RDE) indicates how much useful energy can be extracted from a battery before reaching the discharge limit. Future current loading on vehicle battery systems can be predicted to increase the accuracy of RDE estimations. This is done by using clustering techniques to group load measurements into states, and then using a probability-based framework, along with real-world data, to calculate the transitional probabilities between states. Here, an adapted K-means clustering method is used to cluster load profile data. Markov modelling is used to produce state transition probabilities. Two methods for load prediction are used, which are referred to as the offline-training method and the moving window method, where the offline-training method has not been implemented for this application before. Additional control logic is implemented to combine the proposed load prediction methods to produce a new hybrid load prediction method. This hybrid method shows improved RDE accuracy for a generalised load case. The robustness of the proposed technique is assessed in the presence of model errors, still showing good accuracy when compared to state-of-charge based calculations.**

I. Introduction

Remaining discharge energy (RDE) refers to the amount of energy that can be discharged before a battery reaches its cut-off voltage, or a predetermined SoC value [1]. RDE can be calculated using the equation

$$RDE = \int_{t}^{EoDT} U_t(\tau) \cdot I(\tau) \cdot d\tau, \tag{1}$$

where t is the current time, $EoDT$ is the end of discharge time, U_t is the battery terminal voltage as a function of time τ, and I is the current as a function of time τ [1], [2]. For RDE estimation, three important features are outlined in [1], the calculation of the end-of-discharge time, which is defined as the time to reach the cut-off conditions; the real-time update of battery model parameters; and forecasting future these model parameters over a long prediction horizon.

To date, several methods for RDE estimation have been researched [2], which can be grouped into several categories: the direct calculation method, model-based estimation methods using filters, and prediction-based methods.

A. Direct Calculation Method

The direct calculation method uses an equation based on the current SoC value of the battery. This equation is

$$RDE = Q \cdot U_{nom} \cdot (SOC_t - SOC_{EoDT}), \tag{2}$$

where Q is the battery's rated energy capacity, U_{nom} is the terminal nominal voltage, SOC_t is the state-of-charge value at the current time, and SOC_{EoDT} is state-of-charge value at the end-of-discharge time [1]–[3]. Even though this method is straightforward to implement and has low computational intensity, it uses a constant voltage, which cannot represent the complex voltage responses of the system, thus will cause large errors [2].

B. Model-based Methods

Since battery state-of-energy (SoE) can be defined as the ratio of the battery's remaining energy to the total available energy, RDE can be estimated by calculating SOE. Model-based estimation uses algorithms to estimate SoE, in a similar manner to SoC estimation [2], [4]. Whereas SoE is a measure of the remaining energy, SoC is a measure of remaining capacity, so is not as useful for tasks such as remaining range estimation [5]. SoE is calculated using

$$SoE_t = SoE_{t_0} - \int_{t_0}^{t} P(\tau) d\tau / E_N \tag{3}$$

where SoE_t is the SoE value at the current time t, SoE_{t_0} is the SoE value at the initial time t_0, P is the power output as a function of time τ, and E_N is the total available energy [2]. This calculation relies on accurate estimations of the energy removed from the battery, which can be affected by measurement errors, and an accurate estimation of the maximum available energy, which changes based on the future conditions, such as voltage and temperature [3], [6]. Additionally, model-based estimations are often based on equivalent circuit models with OCV vs SoE curves [5], which cannot account for dynamic condition operating conditions [2].

C. Prediction-based Methods

Prediction-based method attempt to predict future battery states to obtain an RDE estimation. These methods have the advantage of being suitable for all working conditions and can produce accurate RDE estimations by considering future voltage responses. In [1], a method that uses real-time battery model parameter estimation and future battery model parameter prediction for RDE estimation is proposed.

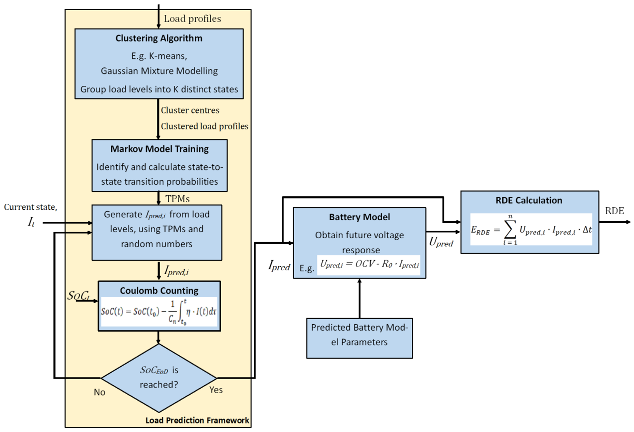

Fig. 1: A flowchart of how the described techniques can be used to estimate RDE.

In the literature, load prediction is often not applied to prediction-based methods, with a priori known load profiles used. This would not be viable for many real-world applications, and can only provide a limited insight to the effectiveness of its estimation accuracy. Therefore, load prediction is a valuable method for increasing the accuracy of RDE estimation, and since it has not been extensively investigated previously, there is a level of uncertainty which can be reduced.

Methods for RDE estimation using load prediction are proposed in [3], [7]. These involve using a moving window to collect load data, which is clustered into distinct states, which represent a sequence of load levels. From this, a Markov model can be produced representing the probabilities of each state transition occurring [3]. Two additional variables are also defined, these being the update interval (UI), which determines how often the RDE estimation is recalculated, and the historical window length (HW), which determines how much data is collected from the moving window [3]. The length of the prediction horizon used here is determined by the time taken to reach the cut-off voltage.

A diagram showing how the discussed techniques are used for RDE estimation is shown in figure 1, which gives an overview of the sequence of steps needed. Using the load data obtained, the future load can be predicted online. New load values are produced and the amount of SoC removed from the battery is predicted using coulomb counting. This process is repeated until the end-of-discharge voltage is reached. The predicted load profiles can be used with a battery model to predict the future voltage response and estimate RDE.

In this paper, an adapted K-means clustering technique is used to obtain relevant cluster information from the load data. Then, two methods for predicting the future load profiles are discussed and compared for defined scenarios. These are the offline-training method, which performs the clustering and calculates the state transition probabilities offline from a set of training load profiles, and uses this information to predict the future load profiles online; and the moving window method, which uses a historical window to gather

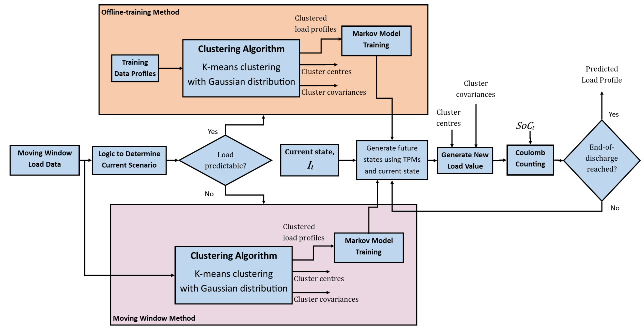

Fig. 2: A flowchart of how the proposed load prediction framework.

the load profile information and performs all the calculations online, which is the method used in [3], [7]. A load prediction method combining both these previous methods is proposed, which uses rule based logic to determine the most appropriate method to use, based on which will give the most accurate estimation. The contributions of this paper is the comparison between offline and online load prediction methods and the introduction of a hybrid method load prediction method, which uses the offline-training method and the moving window method to estimate RDE.

The structure of this paper is as follows: the proposed load prediction framework is introduced and discussed in Section II; simulation results are shown and discussed in Section III; the paper is concluded in Section IV, with future work also outlined.

II. Proposed Load Prediction Framework

The framework of the proposed hybrid load prediction method is shown in figure 2. The framework's online inputs are the historical load data from the moving window; the current load value, I_t; and the current SoC value, SoC_t. The offline inputs are the training load profiles. The output is the predicted load profile.

A. K-means Clustering with Gaussian Distribution

Clustering algorithms refer to unsupervised methods for grouping unlabelled data into structures based on their similarity [8], relative to a set of defined criteria [9], [10]. Thus, a cluster is defined as a subset of the total data set which contains data that is comparable, in contrast with other data points from other clusters. Many clustering techniques are available, with K-means clustering and Gaussian mixture modelling (GMM) being two commonly used algorithms, which both have their distinct advantages for this application.

K-means clustering is a popular choice of algorithm [11], that uses distance measurements to partition a set of data points. It is easily implemented and has a high computational speed, which allows for large data sets to be processed [12], [13]. This is advantageous as it will allow for the data of many load profiles to be processed.

GMM, which is used in [3], describes each cluster as a Gaussian distribution [14]. Since the additional information of the variances are gained using this method, a range of values can be generated using this data. Compared to

K-means, which only produces values for cluster centres, GMM can more closely produce the load values experienced by electric vehicles (EVs). The disadvantage of GMM is the computational complexity of its algorithm, which takes significantly longer to cluster the data, compared with K-means clustering.

The proposed method uses K-means to find the cluster centres of the data set and a calculates the variance of the points in each cluster, with the cluster centre as the mean, where the cluster can be represented as a Gaussian distribution. The steps of the proposed method are as follows:

1. Choose the number of clusters (user-defined), k, with random starting locations.
2. Assign each data point to the closest cluster centre.
3. Relocate the cluster centres to the mean location of its assigned data points.
4. Repeat steps 2 and 3 until there is very little change in the cluster centre locations.
5. Calculate the variance of the data in each cluster.

B. Markov Modelling

Through clustering, load profiles can be described as a sequence of states. This information can be used to form a Markov model [15], with the purpose of forecasting future states [16]. The Markov model uses transitional probability matrices (TPMs) to provide the probabilities of certain state transitions occurring, where only the current state is needed to predict all future states [3], [17].

The TPM is an n-by-n matrix, where n is the number of clusters. The rows represent the current state, and the columns represent the possible future states, so the probability of transitioning from state i to state j is given by M_{ij}. Each TPM is calculated by counting the number of times each transition between specific values for i and j occurs, then dividing it by the total number of times i occurs.

In the context of the two load prediction methods used here, the input data from the moving window method contains a number of data points the same length as the moving window. The offline-training method uses an input of multiple load profiles. Since whole load profiles are considered when using this method, it can better detect how the load values change with time. Hence, a proposed improvement is to calculate multiple TPMs by segmenting the training data profiles, with the length of the segments being user-defined. For the simulations performed here, the length used is 700 seconds with 3 clusters used, which was decided as this was found to produced the most accurate RDE estimation results, although changing the number of clusters did not significantly affect the results.

C. Control Logic

To implement the hybrid load prediction method, different scenarios need to be defined based on load profile data. The data set used here contains around 4600 real world measurement data of EV load profiles, based on a mixture of urban diving patterns. For this method, two scenarios are considered, these being a predictable load profile and an unpredictable load profile. To categorise the data, definitions need to be made regarding predictable and unpredictable load profile segments and group them accordingly. A possible approach is to obtain key performance metrics from the segments and group them depending on how common their metrics are. Similar processes to this have been performed in [18] and [19] to define different driving cycles.

The method used to group the load segments is performed by choosing relevant features from the data set, then clustering the data to find the number of points assigned to each cluster, for an arbitrarily large number of clusters. The largest clusters represent dense regions of points with similar characteristics, and hence, can be considered predictable. Various features can be defined from the data of the load segments. The load profile behaviour that is of most interest for RDE estimation is the size of the current and how transient the profile is. Therefore, the features used here are the mean current and the mean of the absolute rate of change of current.

To choose an appropriate clustering method, it is important to assess the context in which it is needed. Non-hierarchical clustering algorithms, such as K-means, use iterative calculations to assign all data points to cluster centres, whereas hierarchical clustering algorithms, such as an agglomerative method, initialise all points as their own cluster and iteratively combines them [20], [21]. This is done based on various pairwise dissimilarity metrics between two data points in separate clusters, which include maximum distance, minimum distance, and average distance [20]. Since the goal of this clustering is to create large groups of data points with similar values, and smaller groups of outliers, agglomerative clustering with an average distance dissimilarity metric is used, which follows the methods and toolboxes outlined in [22].

The results from the clustering are shown in figure 3. This method is effective at identifying dense regions of points that can be considered predictable load segments. Using each type of load segment, longer load profiles can be created and labelled predictable and unpredictable. For this paper, 70 of the generated predictable load profiles are used as the initial training data set, as adding more than this did not affect the RDE accuracy.

From the cluster centres, the control logic rules can be defined. This is achieved by calculating the mean current and the mean of the absolute rate of change of current of the load in a moving window, then comparing these values of the cluster centres. If the data is closer to a predictable cluster centre, the current state can be considered predictable, otherwise it can be considered unpredictable.

D. Battery Model

The battery model used to obtain future voltage responses is a thermally coupled equivalent circuit model with diffusion (TECMD), from [23]. A detailed description of the model, in terms of its assumptions, parametrisation and dynamic performance is provided in [23] and will not be repeated here. For the simulations, a battery model is used to represent

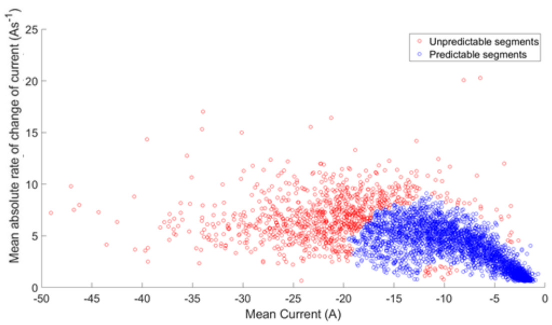

Fig. 3: Distribution of data points from load profile segments, discharge currents are negative values.

the real battery. The states (SoC, temperature etc.) will be taken from this model and used for the voltage prediction model. Therefore, additional robustness simulations must be performed in the presence of model parameter errors for the voltage prediction model to show how the proposed framework performs in a scenario closer to a real-world application. Additionally, for these simulation the initial SoC is set at 85%, and the simulation is ended when the battery voltage reaches the cut-off value.

III. Simulation and Discussion

To test the accuracy of the proposed framework, both defined load profiles will be tested using the hybrid method, as well as using purely the offline-training method and the moving window method, against SoC direct calculation as a reference. The purpose of this is to compare the hybrid method's overall accuracy against the moving-window and offline-training methods for the whole prediction horizon, so the RMSE will be the main metric for comparison. The lower the RMSE, the closer the estimation is to the reference value of RDE.

A. RDE Estimation Using Generated Load Profiles

Firstly, a preliminary test using different clustering methods was performed, so the advantage of using a Gaussian distribution can be assessed. For this test, the direct calculation is compared against the offline-training method using GMM, K-means and Gaussian K-means. The data is shown in Table I, where the results show that the use of a Gaussian distribution gives a more accurate result. Additionally, there is no significant difference between the GMM results and the Gaussian K-means results, and since K-means has much greater efficiency than GMM, it is the optimal choice.

Next, a predictable load profile is tested. The results of this are shown in figure 4. Figure 4(a) shows the specific

TABLE I: A table showing RDE RMSE of the offline-training method, for different clustering techniques, compared against SoC direct calculation.

	SoC direct calculation	K-means	GMM	Gaussian K-means
RDE RMSE (Ws)	20.1	15.3	12.4	12.4

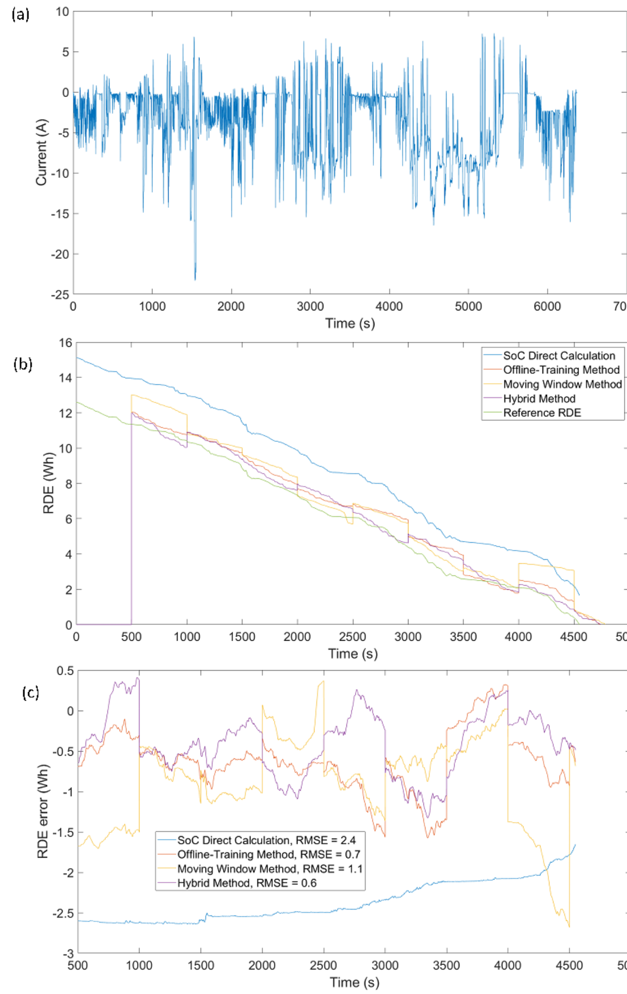

Fig. 4: RDE results from the proposed hybrid load prediction method, using a predictable load profile. (a) The predictable load profile used for the RDE estimation. (b) The calculated RDE using the above load profile, using direct calculation, and offline-training, moving window and hybrid methods. (c) The RDE error from the RDE estimation. Values of UI = 500 and HW = 500 were used.

load profile used. Figure 4(b) shows the RDE estimates using the direct calculation method, as well as the offline-training method, the moving window method, and the hybrid method. The reference RDE is calculated by multiplying the output voltage of the model representing the actual battery and the current values of the load profile together. The direct calculation values are calculated from equation 2. Figure 4(c) shows the RDE error of the methods used. This is calculated by subtracting the RDE estimate from the reference RDE, and the RMSE can be calculated using this. As expected, the SoC direct calculation method performs the worst. When comparing the load prediction methods, the offline-training method produces a more accurate result than the moving window method. The hybrid method performs well, meaning that the implemented logic can correctly identify the scenario.

The results in figure 5 show the RDE accuracy of the proposed method, using an unpredictable profile. Once again,

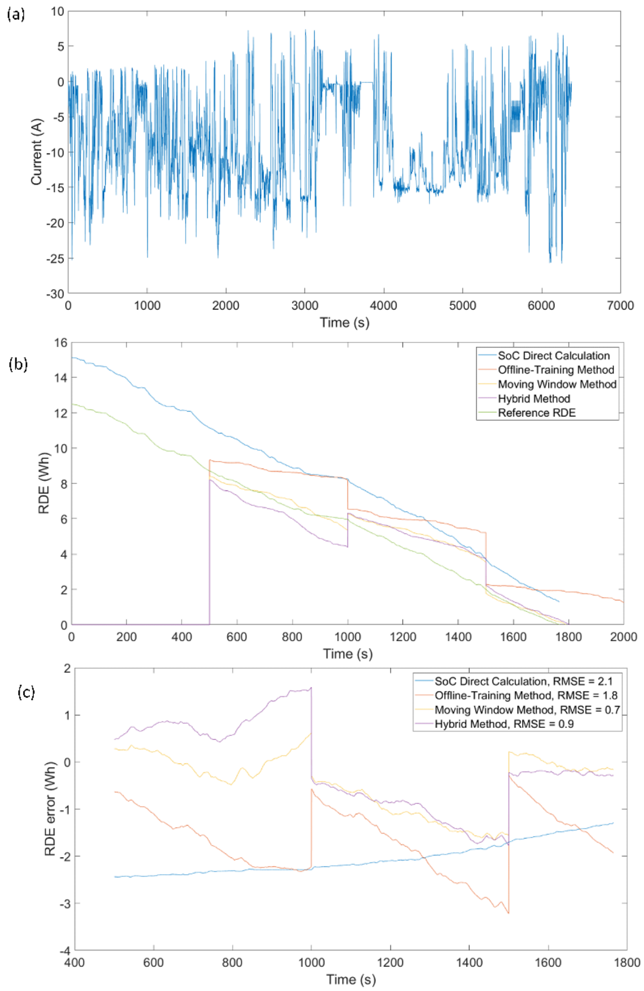

Fig. 5: RDE results from the proposed hybrid load prediction method, using an unpredictable load profile. (a) The unpredictable load profile used for the RDE estimation. (b) The calculated RDE using the above load profile, using direct calculation, and offline-training, moving window and hybrid methods. (c) The RDE error from the RDE estimation. Values of UI = 500 and HW = 500 were used.

figure 5(a) show the load profile used, and figures 5(b) and (c) show the RDE results and RDE error results respectively. For the profile shown, the SoC direct calculation method shows the worst accuracy. Unlike the predictable profile, the moving window method performs better than the offline-training method for this type of profile. The proposed hybrid method performs well at distinguishing unpredictable scenarios.

To assess the accuracy of each method for a general case, the RDE RMSE is averaged over 40 of each profile, with each profile being performed five times (to account for RDE changes from the random numbers generated). The results of this are shown in table II. For the predictable profiles, the offline-training method provides the smallest RDE RMSE, followed by the hybrid method, and then the moving window method. The difference between the offline-training results and the hybrid method is $\sim$18%. This is a significant difference, which can be attributed to the control logic, since it decides what load prediction method is being used. If the logic was improved further, by more rigorously defining the clusters for the different scenarios and improving the method used for identifying the current scenario, the accuracy would increase. For the unpredictable profiles, the moving window method provides the greatest accuracy. The difference between this result and the results from the hybrid method, shows the control logic performs better for unpredictable profiles. Overall, the accuracy of the hybrid method performs significantly better generally, than the other two method.

B. *Robustness Evaluation of the Proposed Hybrid Method*

For a real-world application of the proposed RDE estimation method, uncertainties in battery model voltage response and states will be present. To test for these, values in the model used to obtain the future voltage are changed. These include SoC, capacity, temperature, and internal resistance. The results from this are shown in table III. The results were averaged over five different load profiles. From the table, it can be seen that SoC and capacity errors have the largest effect on the RDE accuracy when compared to temperature and internal resistance. This can be attributed to the fact that these are both measurements of capacity, which is related to how much energy can be extracted from the battery. Additionally, it will also affect the load prediction and voltage prediction values. The direct calculation results are shown with no change in parameters, meaning the true values are used. Even though this is the case, the RMSE of the prediction-based methods are still significantly lower than the direct calculation method.

IV. Conclusion

Prediction-based methods have been previously shown to be a promising technique for increasing RDE accuracy. The framework outlined here builds upon existing load prediction techniques by proposing an offline-training method, which obtains the information needed to predict future load profiles offline. Additionally, the two load prediction methods are combined with supervisory control logic to determine the best method to use depending on the current scenario. Firstly, the results from the simulations show that the SoC direct calculation method is inaccurate when compared to the prediction-based methods. Next, the results show that for predictable load profiles, the offline-training method provides the greatest accuracy, but for unpredictable load profiles, the moving window method provides the greatest accuracy. Overall, when considering both types of profile, the proposed hybrid method shows the greatest accuracy. In addition, the difference between the accuracy of the offline-training method and the hybrid method, for the predictable load profiles, is significant, meaning the implemented control logic could be further improved to provide even greater accuracy. The proposed method's accuracy was also tested in the presence of parameter errors. From this, it was found that capacity and SoC errors have the largest effects on the RDE accuracy. It was also found that even when errors are

TABLE II: A table showing RDE RMSE for the proposed methods for predictable and unpredictable profiles.

	Offline-training Method	Moving Window Method	Hybrid Method
Predictable	0.54	1.01	0.65
Unpredictable	1.37	0.99	1.03
Overall	0.95	1.00	0.84

TABLE III: A table showing RDE RMSE for robustness testing using predictable (top entries) and unpredictable profiles (bottom entries).

	RDE RMSE (Wh)			
	No scaling	Direct calculation	Scaling = 0.9	Scaling = 1.1
SoC	0.63 — 0.80	1.83	0.98 — 1.08	1.01 — 1.21
Capacity (4.9 Ah)	0.63 — 0.80	1.83	0.91 — 1.02	0.95 — 1.17
Temperature (°C)	0.63 — 0.80	1.83	0.64 — 0.81	0.63 — 0.80
Internal resistance (12.5 mΩ)	0.63 — 0.80	1.83	0.62 — 0.83	0.63 — 0.82

present, the proposed method still achieves more accurate results than the direct calculation method.

Two categories exist for further work, these being experimental work and extensions to the proposed method. The experimental work would consist of using real cells to validate the proposed algorithm, along with an improved battery model and an SoC estimation algorithm. Extensions of this method would seek to improve the RDE accuracy. This could be achieved by more rigorously defining additional scenarios (only two were considered here), each with their own TPMs. This will also require using different clustering techniques for high dimensional data. Improvements in the accuracy could also be achieved by improving the control logic and its ability to identify the current scenario.

References

[1] G. Liu, M. Ouyang, L. Lu, J. Li, and J. Hua, "A highly accurate predictive-adaptive method for lithium-ion battery remaining discharge energy prediction in electric vehicle applications," *Applied Energy*, vol. 149, pp. 297–314, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2015.03.110

[2] D. Ren, L. Lu, P. Shen, X. Feng, X. Han, and M. Ouyang, "Battery remaining discharge energy estimation based on prediction of future operating conditions," *Journal of Energy Storage*, vol. 25, no. July, p. 100836, 2019. [Online]. Available: https://doi.org/10.1016/j.est.2019.100836

[3] M. F. Niri, T. M. Bui, T. Q. Dinh, E. Hosseinzadeh, T. F. Yu, and J. Marco, "Remaining energy estimation for lithium-ion batteries via Gaussian mixture and Markov models for future load prediction," *Journal of Energy Storage*, vol. 28, no. January, p. 101271, 2020. [Online]. Available: https://doi.org/10.1016/j.est.2020.101271

[4] K. Mamadou, E. Lemaire, A. Delaille, D. Riu, S. E. Hing, and Y. Bultel, "Definition of a State-of-Energy Indicator (SoE) for Electrochemical Storage Devices: Application for Energetic Availability Forecasting," *Journal of The Electrochemical Society*, vol. 159, no. 8, pp. A1298–A1307, 2012.

[5] Y. Z. Zhang, H. W. He, and R. Xiong, "A Data-Driven Based State of Energy Estimator of Lithium-ion Batteries Used to Supply Electric Vehicles," *Energy Procedia*, vol. 75, pp. 1944–1949, 2015.

[6] R. Xiong, Y. Zhang, H. He, X. Zhou, and M. G. Pecht, "A double-scale, particle-filtering, energy state prediction algorithm for lithium-ion batteries," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1526–1538, 2017.

[7] X. Lai, Y. Huang, H. Gu, X. Han, X. Feng, H. Dai, Y. Zheng, and M. Ouyang, "Remaining discharge energy estimation for lithium-ion batteries based on future load prediction considering temperature and ageing effects," *Energy*, vol. 238, p. 121754, 2022. [Online]. Available: https://doi.org/10.1016/j.energy.2021.121754

[8] B. Basaran and F. Günes, "Data clustering," *Intelligent multidimensional data clustering and analysis*, vol. 31, no. 3, pp. 28–72, 2016.

[9] O. Nasraoui and C.-E. Ben N'Cir, *Clustering Methods for Big Data Analytics*, 2019. [Online]. Available: https://www.springer.com/gp/book/9783319978635%0Ahttp://link.springer.com/10.1007/978-3-319-97864-2

[10] B. Mirkin, *Mathematical Classification and Clustering*, 1996.

[11] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, "Constrained K-means Clustering with Background Knowledge," *International Conference on Machine Learning ICML*, vol. pages, pp. 577–584, 2001. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.4624&rep=rep1&type=pdf

[12] N. Dhanachandra, K. Manglem, and Y. J. Chanu, "Image Segmentation Using K-means Clustering Algorithm and Subtractive Clustering Algorithm," *Procedia Computer Science*, vol. 54, pp. 764–771, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.procs.2015.06.090

[13] T. Kodinariya and P. Makwana, "Review on determining number of Cluster in K-Means Clustering," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 1, no. 6, pp. 90–95, 2013.

[14] X. He, D. Cai, Y. Shao, H. Bao, and J. Han, "Laplacian regularized Gaussian mixture model for data clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 9, pp. 1406–1418, 2011.

[15] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *PROCEEDINGS OF THE IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[16] T. W. Anderson and L. A. Goodman, "Statistical Inference About Markov Chains," *The Annals of Mathematical Statistic*, vol. vol. 28, n, no. 03, pp. pp. 89–110, 1957.

[17] J. R. Beck and S. G. Pauker, "The Markov Process in Medical Prognosis," *Med Decis Making*, vol. 3, no. 4, 1983.

[18] C. Pan, W. Dai, L. Chen, L. Chen, and L. Wang, "Driving range estimation for electric vehicles based on driving condition identification and forecast," *AIP Advances*, vol. 7, no. 10, 2017. [Online]. Available: http://dx.doi.org/10.1063/1.4993945

[19] X. Lin, G. Zhang, and S. Wei, "Velocity prediction using Markov Chain combined with driving pattern recognition and applied to Dual-Motor Electric Vehicle energy consumption evaluation," *Applied Soft Computing*, vol. 101, p. 106998, 2021. [Online]. Available: https://doi.org/10.1016/j.asoc.2020.106998

[20] U. Braga-Neto, *Fundamentals of Pattern Recognition and Machine Learning*, 2020.

[21] F. K. Gulagiz and S. Suhap, "Comparison of Hierarchical and Non-Hierarchical Clustering Algorithms," *International Journal of Computer Engineering and Information Technology*, vol. 9, no. 1, pp. 6–14, 2017. [Online]. Available: www.ijceit.org

[22] "Hierarchical Clustering - MATLAB & Simulink - MathWorks United Kingdom." [Online]. Available: https://uk.mathworks.com/help/stats/hierarchical-clustering.html

[23] W. D. Widanage, "A Simscape-Battery-Library," 2021. [Online]. Available: https://github.com/WDWidanage/Simscape-Battery-Library/releases/tag/v1.0.0