An End-to-End Learning Framework for Early Prediction of Battery Capacity Trajectory

Jinqiang Liu[1], Adam Thelen[2], Chao Hu[1,2], and Xiao-Guang Yang[3]

[1] *Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, 50011, USA*
jinqiang@iastate.edu

[2] *Department of Mechanical Engineering, Iowa State University, Ames, IA, 50011, USA*
acthelen@iastate.edu
chaohu@iastate.edu

[3] *Department of Mechanical Engineering and Electrochemical Engine Center (ECEC), the Pennsylvania State University, University Park, PA 16802, USA*
xuy19@psu.edu

ABSTRACT

Accurately predicting the entire capacity trajectory using early-life data enables more efficient cell design, operation, maintenance, and evaluation for second-life use. To accomplish this challenging task, we propose an end-to-end learning framework combining empirical capacity fade models and data-driven machine learning models, in which the two types of models are closely coupled. First, we evaluate the accuracy of a library of relevant empirical models which have been shown to model the observed capacity fade of Li-ion cells with reasonable accuracy. After selecting a power-law model, we formulate an end-to-end learning problem that simultaneously fits the chosen power-law model to estimate the capacity fade curve and trains an elastic net to estimate the best-fit parameters of the empirical model. Our proposed end-to-end learning framework is evaluated using a publicly available battery dataset consisting of 124 lithium-iron-phosphate/graphite cells charged with various fast-charging protocols. This dataset was split into training, primary test, and secondary test datasets. Our method performs on par with existing early prediction methods in terms of cycle life prediction, attaining root-mean-square errors of 112 cycles and 165 cycles for primary and secondary test datasets, respectively. In addition to the cycle life prediction, our method possesses a unique ability to predict the entire capacity trajectory.

1. INTRODUCTION

Capacity-trajectory prediction is a critically important task given its broad utility throughout the battery product life cycle. Examples where capacity fade models prove useful are new material selection, manufacturing process validation, remaining useful life estimation for predictive maintenance, and charge/discharge protocol optimization [1]–[3]. Even more useful is estimating a battery cell's capacity trajectory when the cell has not exhibited any noticeable capacity degradation. Additionally, a new area of research aims to repurpose Li-ion batteries from electric vehicles (EVs). This research requires more understanding of the battery degradation profile than simply estimating the final cycle life of a cell [4]. Accurately predicting the entire capacity trajectory using early-life data not only provides accurate and early estimates of battery end-of-life (EoL), but also sheds light on the battery capacity degradation process over cycles. These tools will enable more efficient cell design, operation, maintenance, and evaluation for second-life use.

A different yet related area of research is battery lifetime prognostics. Many studies on battery lifetime prognostics attempted battery capacity-trajectory prediction. They can be divided into model-based methods and data-driven methods [5]. Model-based methods are developed based on a mechanistic model of degradation mechanisms [6], [7], an equivalent circuit model (ECM) of electrical performance [8], or an empirical model of capacity fade [1], [9]–[12]. The mechanistic model-based methods consider the internal electrochemical processes and generally achieve the highest accuracy and the greatest generalization ability. However, the high computation cost, expertise required, and difficulties in identifying model parameters inhibit the use of mechanistic

model-based methods. ECM-based methods consider battery aging mechanisms to some degree by modeling the growth of internal resistance. However, this technique often requires the use of electrochemical impedance spectroscopy (EIS), which necessitates expensive test equipment and may accelerate the battery aging process. Empirical model-based methods have been widely used because they are easy to establish and achieve acceptable accuracy. The most common implementation of empirical models is done by using a filtering algorithm to online estimate the empirical model parameters based on the latest available data measurements. Common filtering variants include particle filters [1], [11], extended Kalman filters [13], and unscented Kalman filters [14]. An inherent advantage to filtering techniques is that they can produce probabilistic predictions, making them more easily integrated into a prognostics-informed decision-making framework. The main drawback of empirical model-based methods is that they require a large amount of historical capacity fade data from the online cell in order to be accurate in estimating its future trajectory. This is also a common issue for most battery lifetime prognostic methods: they generally need to collect more than 40% of the entire life-cycle data from a cell to either estimate model parameters or train a data-driven model [5]. Another drawback of empirical model-based methods is the lack of shared information between cells. Empirical models are not able to be adjusted given extra information about previous best fit parameters from other cells in the fleet. This lack of shared information combined with the growing size of modern battery datasets is a main reason why data-driven models are an effective alternative.

Data-driven methods predict the capacity degradation trend based on historical data from a group of similar cells using various techniques including support vector machine [15], relevance vector machine [16], Gaussian process regression [17], and neural networks [18], [19]. Researchers in [20] developed a regression model using early-life statistical features from cell voltage vs. discharge capacity curves to estimate cell cycle life. The work showed for the first time that the cycle life for cells cycled under various conditions could be accurately estimated using data from only the first 100 cycles. Most related to our work is the early prediction study presented in [21]. Instead of only focusing on a point prediction like cell end-of-life (EoL) or remaining useful life (RUL), Li et al. proposed using a long short-term memory (LSTM) deep learning model to predict the battery capacity trajectory [21]. The input to the trained sequence-to-sequence LSTM model is nothing more than the previous capacity measurements of the online cell. The method achieved good accuracy on both capacity-trajectory prediction and cycle life prediction for the blink dataset. [21]. While all these new methods have found success, the techniques largely consist of machine/deep learning models that are trained entirely on statistical correlations found in training data. Researchers have opted for chemistry- and cycling profile-agnostic machine learning models to gain predictive accuracy but have in turn sacrificed model generalizability. The small sizes of battery aging test datasets tend to lead to overfit machine learning models as more accuracy is demanded of them. This is especially true for deep learning models and regression models which perform single point prediction, i.e. predicting cell EoL and RUL. It is critically important to evaluate data-driven model's generalization performance on more unseen test datasets including cells with different cycling conditions and storage conditions (e.g., storage time). There remains a significant opportunity for advancing the state-of-the-art in capacity-trajectory prediction to include higher accuracy, better generalization to new cycling conditions, and earlier prediction.

In this study, we aim to improve the information output of data-driven battery early life prediction machine learning models by augmenting them with empirical capacity fade models. Frist, we evaluate the accuracy of a library of relevant empirical models which have been shown to model the observed capacity fade of Li-ion cells with reasonable accuracy. We then formulate an end-to-end learning problem that simultaneously fits the chosen empirical model to estimate the capacity fade curve and trains a machine learning model to estimate the best-fit parameters of the empirical model. By solving this end-to-end learning problem, rather than sequentially executing the separate tasks of fitting the capacity fade model and training the machine learning model, we achieve a more optimal solution which is shown to better balance these two objectives. Our proposed end-to-end learning framework is evaluated using a publicly available battery dataset consisting of 124 lithium-iron-phosphate/graphite (LFP/graphite) cells charged with various fast-charging protocols [20], [22].

The rest of the paper is outlined as follows: Section 2 describes the methodology for the end-to-end learning framework and the implementation procedure for the end-to-end learning model. Section 3 presents the results and comparisons with other methods. Conclusions are drawn in Section 4.

2. METHODOLOGY

2.1. Empirical Capacity Fade Models

There are several empirical models reported in the literature that can be used to represent the Li-ion battery capacity fade behavior, including linear model [1], two-term exponential model [12], exponential/linear hybrid model [1], power-law model [10], exponential/power-law hybrid model [16], and two-term power-law model [10]. Each of the candidate empirical capacity fade models are evaluated for use in the end-to-end learning framework. The capacity data used for empirical model fitting were first normalized by dividing each capacity measurement by the value of the first capacity measurement. This ensures all cells start at a normalized

capacity of 1.0. Other than normalization, no preprocessing or denoising was performed. The models were evaluated for both overall fitting accuracy by comparing mean RMSE across all 41 training cells, and model simplicity indicated by the number of parameters needed to define the capacity fade curve. The power-law model was found to have acceptable accuracy (more accurate than the linear model and the exponential/linear hybrid model) and the least number of parameters, and so it was chosen as the candidate model for the end-to-end learning framework.

Power-law model [10]: This model assumes capacity fade can be represented as a single power law model, shown as

$$Q_k = 1 - a \cdot k^b \tag{1}$$

where a is the coefficient of the power law and b is the power-law exponent. The choice of empirical model is dataset dependent, and any of the aforementioned empirical models can be used in the proposed end-to-end learning framework without increasing implementation difficulty.

2.2. End-to-end Learning Framework

The end-to-end learning framework simultaneously fits an empirical model to estimate the capacity trajectory and trains a machine learning model to estimate the best-fit parameters of the empirical model. After selecting the power-law model as the empirical model and elastic net as the machine learning model, the problem is formulated as

$$\begin{aligned}
\min_{\mathbf{w}\in\mathbb{R}^{mn}} \quad & \frac{\lambda}{2mn}\left\|\log_{10}(1-\mathbf{Q}) - \mathbf{p}_1\mathbf{1}_{1\times m} - \mathbf{p}_2\mathbf{1}_{1\times m}\circ\log_{10}(\mathbf{K})\right\|_F^2 \\
& + \frac{1-\lambda}{2n}\left\|\log_{10}(1-\mathbf{Q}_{\mathrm{EoL}}) - \mathbf{p}_1 - \mathbf{p}_2\circ\log_{10}(\mathbf{K}_{\mathrm{EoL}})\right\|_2^2 \\
& + \alpha r\|\mathbf{w}\|_1 + \frac{\alpha}{2}(1-r)\|\mathbf{w}\|_2^2 \\
\text{subject to} \quad & \mathbf{p}_1 = \mathbf{X}\mathbf{w}_1 \\
& \mathbf{p}_2 = \mathbf{X}\mathbf{w}_2 \\
& \log_{10}(1-\mathbf{Q}_{\mathrm{EoL}}) - \mathbf{p}_1 - \mathbf{p}_2\log_{10}(l_{\max}) \le 0 \\
& \log_{10}(1-\mathbf{Q}_{\mathrm{EoL}}) - \mathbf{p}_1 - \mathbf{p}_2\log_{10}(l_{\min}) \ge 0 \\
& p_{1,\min} \le \mathbf{p}_1 \le p_{1,\max} \\
& p_{2,\min} \le \mathbf{p}_2 \le p_{2,\max} \\
& \mathbf{w} = \left[\mathbf{w}_1^{\mathrm{T}}, \mathbf{w}_2^{\mathrm{T}}\right]^{\mathrm{T}}
\end{aligned} \tag{2}$$

where m is the number of sample points for a capacity fade curve; n is the number of cells in the training dataset; λ is the ratio of the first term in modeling the capacity-trajectory prediction error, $\mathbf{Q}$ is an $n\times m$ capacity matrix for all sample points in the capacity fade curves of all cells and $\mathbf{K}$ is the corresponding $n\times m$ cycle matrix; $\mathbf{p}_1$ and $\mathbf{p}_2$ are respectively the vectors of best-fit values of two power-law capacity fade model parameters $\log_{10}(a)$ and b; $\mathbf{X}$ is the early-life feature matrix; $\mathbf{w}_1$ and $\mathbf{w}_2$ are the linear regression model coefficients for estimating best-fit empirical model parameters; α is the regularization magnitude; r is the ratio of the L_1 norm term in the regularization; $\mathbf{Q}_{\mathrm{EoL}}$ are the end-of-life capacity values of all training cells; $l_{\max}$ and $l_{\min}$ are the upper and lower limits for the cycle life and are empirically determined as 300 and 3000 in this study; and $p_{1,\min}$, $p_{2,\min}$, $p_{1,\max}$, and $p_{2,\max}$ are the upper and lower limits for the two empirical model parameters. Note that we use the same number of sample points m from the capacity fade curve of each cell, regardless of its cycle life. If we were to train the model using all the available capacity datapoints for each cell (i.e. m is equal to the cycle life of a given cell) in the training dataset, the model would be biased towards cells with longer cycle lives. A solution to this is to sample each cell's capacity fade curve at an evenly spaced interval of cycle number, so that every cell has the same impact during model training.

The objective is to minimize the capacity fade prediction error for all cells in the training dataset as well as the regularization penalties for the linear regression model coefficients. The first term in the objective function describes the $\log_{10}$-scale error of capacity-trajectory prediction for all cells. The $\log_{10}$ function is inherently more sensitive to the sample points at the early stage of the capacity trajectory. To increase model sensitivity to the capacity points near the EoL, we add a second term to the objective function which measures the capacity prediction error at the EoL. The third and fourth terms are the $L1$ and $L2$ regularization penalties.

The early-life features, $\mathbf{X}$, are fed as input to the linear regression model which predicts the empirical model parameters. Six early-life features are used as the input to the machine learning model. These six features are the same input features used in the "discharge model" introduced in [20], namely initial capacity, maximum capacity between cycle 10 and cycle 100, $\log_{10}|\mathrm{Min}(\Delta Q_{100-10})|$, $\log_{10}|\mathrm{Var}(\Delta Q_{100-10})|$, $\log_{10}|\mathrm{Skew}(\Delta Q_{100-10})|$, and $\log_{10}|\mathrm{Kurt}(\Delta Q_{100-10})|$. Here, the terminology ΔQ_{100-10} is referring to the difference in the discharge capacity vs voltage curves at cycle 100 and cycle 10. Researchers in [20] developed numerous features based on the difference in curves at cycle 100 and cycle 10. While many features were proposed and examined, the "discharge model" used four features from the ΔQ_{100-10} vs. voltage curve and two features from the capacity fade curve for the first 100 cycles, achieving the best accuracy for cycle life prediction [20], [22]. We chose to use the same six early-life features to demonstrate our proposed end-to-end learning framework's effectiveness as compared to similarly reported models. In our future work, feature selection will be performed for a more optimized and thorough discussion.

To better understand the advantages of the proposed end-to-end learning framework, we formulate a sequential optimization problem to be used as a baseline framework for comparison. The sequential optimization framework

executes the separate tasks of fitting a capacity fade model and training a machine learning model. First, we obtain best-fit empirical model parameters for all cells in the training set by solving the following unconstrained curve-fitting optimization problem

$$\min_{\mathbf{p}_1,\mathbf{p}_2} \frac{\lambda}{2mn}\left\|\log_{10}(1-\mathbf{Q})-\mathbf{p}_1\mathbf{1}_{1\times m}-\mathbf{p}_2\mathbf{1}_{1\times m}\circ\log_{10}(\mathbf{K})\right\|_F^2 + \frac{1-\lambda}{2n}\left\|\log_{10}(1-\mathbf{Q}_{\mathrm{EoL}})-\mathbf{p}_1-\mathbf{p}_2\circ\log_{10}(\mathbf{K}_{\mathrm{EoL}})\right\|_2^2 \tag{3}$$

The curve-fitting results for the training cells are used to set the upper and lower limits for the empirical model parameters. These limits are also used in the end-to-end framework. The limits are expressed as

$$\begin{aligned} p_{2,\min} &= 0.8\cdot\min(\mathbf{p}_2)\\ p_{2\max} &= 1.2\cdot\max(\mathbf{p}_2)\\ p_{1,\min} &= \log_{10}(1\text{-}Q_{\mathrm{EoL}})-p_{2,\max}\log_{10}(l_{\max})\\ p_{1,\max} &= \log_{10}(1\text{-}Q_{\mathrm{EoL}})-p_{2,\min}\log_{10}(l_{\min}) \end{aligned} \tag{4}$$

where $\mathbf{p}_{1,}$ is the fitted empirical model parameters obtained by solving the unconstrained curve-fitting optimization problem in Eq. (3). These empirical model parameter constraints are incorporated into the end-to-end learning model for two purposes. First, they incorporate relevant information about typical empirical model parameter values from the training dataset. The constraints loosely restrict the range of the empirical model parameters (and subsequently the range of predicted cycle lives) during the training process to improve the model's generalization capability. Second, the constraints ensure the predictions from the model remain physically meaningful, limiting the model to only predict capacity fade trajectories that decrease and eventually cross the EoL threshold.

For the second part of the sequential optimization method, we define two elastic net machine learning models that map the early-life features to the best-fit empirical model parameters, one for each empirical model parameter. The training processes of these two elastic net models are formulated as

$$\min_{\mathbf{w}_1} \frac{1}{2n}\left\|\mathbf{X}\mathbf{w}_1-\mathbf{p}_1\right\|_2^2+\alpha_1 r_1\left\|\mathbf{w}_1\right\|_1+\frac{\alpha_1}{2}(1-r_1)\left\|\mathbf{w}_1\right\|_2^2 \tag{5}$$

$$\min_{\mathbf{w}_2} \frac{1}{2n}\left\|\mathbf{X}\mathbf{w}_2-\mathbf{p}_2\right\|_2^2+\alpha_2 r_2\left\|\mathbf{w}_2\right\|_1+\frac{\alpha_2}{2}(1-r_2)\left\|\mathbf{w}_2\right\|_2^2 \tag{6}$$

2.3. Implementation of the End-to-end Learning Model

The optimization problem formulated for the end-to-end learning model in Eq. (2) can be solved by most off-the-shelf solvers. However, the heaviest computation burden is in determining the values for the three hyperparameters, λ, α, and r. To determine the optimal values for the hyperparameters, we use a random search technique combined with repeated cross validation (CV). The candidate hyperparameters are randomly and independently sampled from uniform distributions, denoted as

$$\begin{aligned} \lambda &\sim U(0.5,1)\\ r &\sim U(0,1)\\ \log_{10}\alpha &\sim U(-4,-0.5) \end{aligned} \tag{7}$$

A random search method was chosen because it was determined during early experimentation that the model was more sensitive to the regularization parameter α, but indifferent to changes in λ and r [23]. Knowing this and determining that the average evaluation time for a single set of hyperparameter values takes roughly 4 minutes, random search was used to better explore the large α parameter space in a shorter amount of time than a grid search method did. To evaluate the trained end-to-end model with a specific set of hyperparameters, a CV method is employed. The v-fold CV may produce an error estimate that is highly variable if the dataset is too small, which is exactly the case for this battery capacity-trajectory prediction problem. In order to enhance the stability of the v-fold CV estimate, we propose using repeated v-fold CV to determine the optimal hyperparameters for the end-to-end learning model.

The pseudo-code for determining the optimal hyperparameters and implementing an optimized end-to-end learning framework is summarized in Table 1. The procedure begins with the random generation of N hyperparameter tuples based on their distributions. Next, T training datasets are generated by randomly splitting the 41-cell training dataset into v-folds of nearly equal size. Then, v-fold cross validation is performed for all sets T, and the repeated cross-validation error (RCV) is recorded. Using T sets better accounts for the run-to-run variation experienced by randomly generating the v-folds. To select the optimal hyperparameters, the mean RCV value and standard deviation is calculated for all N tuples of hyperparameters. There will exist a few different sets of hyperparameters which are found to produce mean RCV less than the minimum mean RCV plus the corresponding standard deviation, and so we select the set of hyperparameters in which α is maximum. Selecting a set of hyperparameters which produce approximate minimum error and have a large regularization magnitude α ensures the end-to-end model is less likely to overfit the training data. In this study, N is set to 100, T is set to 100, and v is set to 4. consists of the following steps.

3. RESULTS

The open-source battery aging dataset in [20] was used to validate the proposed end-to-end learning framework. The dataset consists of 124 LFP/graphite A123 APR18650M1A cells cycled under different fast-charging protocols. The

Table 1. Procedure for implementing the end-to-end learning model

Algorithm: Implementing the end-to-end learning model

Inputs: Training dataset: **X**, **Q**, **K**
Hyperparameter distributions:
$\lambda \sim U(0.5,1); r \sim U(0,1); \log_{10}\alpha \sim U(-4,-0.5)$
Sample number for the random search: N
Number of repeated v-fold CV evaluations: T
CV folds: v

Output: Trained machine learning model: $\mathbf{w}_1$, $\mathbf{w}_2$

for i=1: N
  Generate a tuple of random hyperparameters: (α_i, r_i, $\eta_{1,i}$) based on their distributions.
  for j=1: T
    Generate a CV dataset by randomly splitting the 42-cell training dataset into v subsets with nearly equal size.
    for k=1: v
      Train the end-to-end learning model using the CV subsets excluding the kth subset.
      Validate the trained model on the kth subset and calculate the validation error, CV_k
    end for
    Calculate the jth v-fold CV error:

$$\mathrm{CV}^{(j)} = \sum_{j=1}^{k} \mathrm{CV}_k / v$$

  end for
  Save the repeated v-fold CV error vector:

$$\mathbf{RCV}_i = \{\mathrm{CV}^{(j)}\}_{j=1:T}$$

end for
Calculate the mean $\mathrm{RCV}_{i,\mathrm{mean}}$ and standard deviation $\mathrm{RCV}_{i,\mathrm{std}}$ of the repeated v-fold CV error vector $\mathbf{RCV}_i$ corresponding to the hyperparameters $\lambda_{1,i}$, α_i, and r_i.
Obtain the sample indices of hyperparameter tuples which have minimum $\mathrm{RCV}_{i,\mathrm{mean}}$:

$$i^- = \arg\min_i \mathrm{RCV}_{i,\mathrm{mean}}$$

Determine the optimal sample index:

$$i^* = \arg\max_i \left(\alpha_i \mid \mathrm{RCV}_{i,\mathrm{mean}} \le \mathrm{RCV}_{i^-,\mathrm{mean}} + \mathrm{RCV}_{i^-,\mathrm{std}} \right)$$

Train the end-to-end model on the training dataset using the optimal hyperparameters λ_{i^*}, α_{i^*}, and r_{i^*}.

original authors split the dataset into three subsets, namely a training dataset (41 cells), a primary test dataset (43 cells), and a secondary test dataset (40 cells). To ensure the proposed end-to-end learning framework is directly comparable to similar models for cycle life prediction, the same dataset partitioning used in [20] is also used in this paper. The capacity fade trajectories of all the cells are plotted in Fig. 1. One cell in the primary test dataset experienced extremely fast capacity degradation and was recognized as an outlier in the original dataset description. Therefore, we have removed this cell from the primary test dataset. The end-to-end learning framework is trained on the training dataset and evaluated on both the primary and secondary test datasets. Figure 2 presents the predicted capacity trajectories for some sample cells in the primary and secondary test datasets.

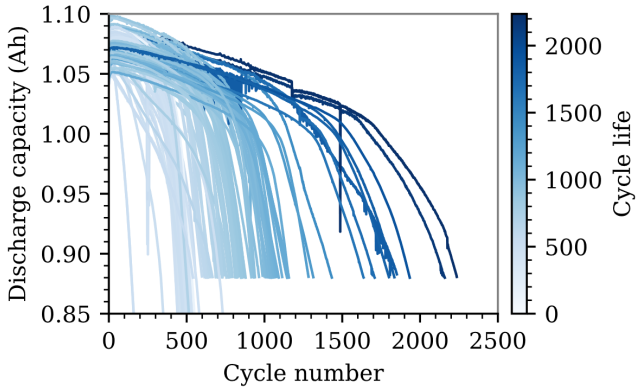
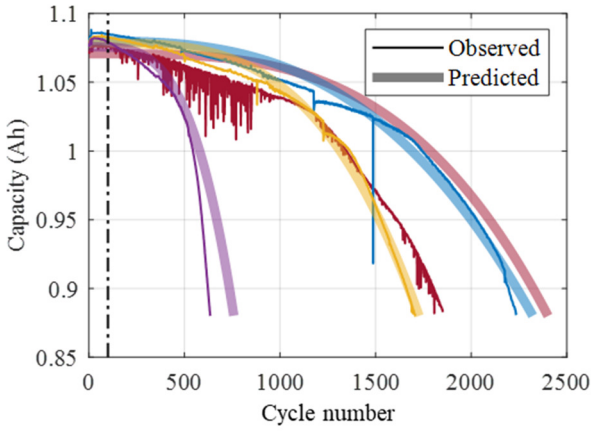
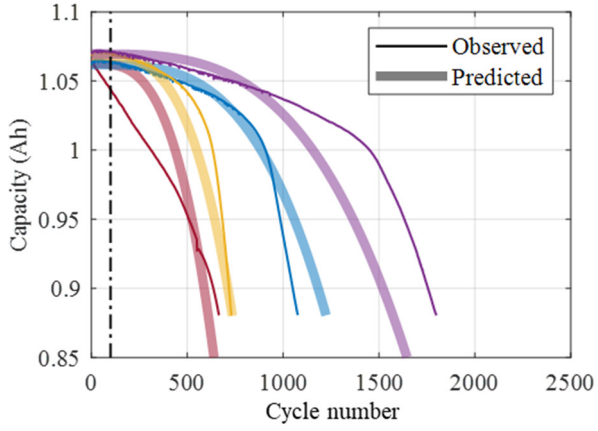


Figure 1. Capacity fade curves of all 124 cells in the open-source dataset.



(a)



(b)

Figure 2. Example capacity-trajectory predictions by the end-to-end learning model. (a) Four primary test cells. (b) Four secondary test cells.

3.1. Model Performance Evaluation Criteria

The proposed end-to-end learning framework not only provides cycle life estimates, but also provides an estimate of the capacity trajectory. Thus, it is important that a predicted capacity fade curve for a cell accurately models the observed capacity trajectory, and so we define the capacity-based root-mean-square error (RMSE) metric as follows

$$\mathrm{RMSE}_Q = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{m}\sum_{j=1}^{m}\left(Q_{ij}-\hat{Q}_{ij}\right)^2\right]} \tag{8}$$

where n is the cell number, m is the number of points sampled from the capacity trajectory, Q_{ij} is the true capacity value in mAh for cell i at the jth sample point of the capacity trajectory, and $\hat{Q}_{ij}$ is the predicted capacity value for Q_{ij}. In this study, m is set to 100 to strike a balance between computation time and modeling accuracy.

While RMSE_Q is used to evaluate the model's predictive performance on estimating the capacity values at specific cycles, we define another RMSE metric to evaluate the model's predictive performance on estimating the cycle number at a specific capacity fade level. This metric is denoted as

$$\mathrm{RMSE}_{\mathrm{Cycle}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{m}\sum_{j=1}^{m}\left(k_{ij}-\hat{k}_{ij}\right)^2\right]} \tag{9}$$

where k_{ij} is the true cycle value for cell i at the capacity level of the jth sample point, and $\hat{k}_{ij}$ is the predicted cycle for k_{ij}.

In addition to the metrics used to evaluate the capacity-trajectory predictions, we also define the RMSE and mean absolute percentage error (MAPE) to evaluate the end-to-end learning model's performance on cycle life prediction. In this study, and similar to the work in [20], a cell's cycle life is defined as the cycle number at which the cell's capacity reaches 80% of the nominal capacity. These two metrics are defined as

$$\mathrm{RMSE}_{\mathrm{EoL}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\left(k_{\mathrm{EoL},i}-\hat{k}_{\mathrm{EoL},i}\right)^2\right]} \tag{10}$$

$$\mathrm{MAPE}_{\mathrm{EoL}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|k_{\mathrm{EoL},i}-\hat{k}_{\mathrm{EoL},i}\right|}{k_{\mathrm{EoL},i}}\times 100 \tag{11}$$

where $k_{\mathrm{EoL},i}$ and $\hat{k}_{\mathrm{EoL},i}$ are the observed and predicted cycle lives of the ith cell, respectively.

3.2. Model Performance

Model performance is evaluated from two perspectives. First, we evaluated the end-to-end framework and the sequential optimization framework in terms of capacity-trajectory prediction accuracy. This analysis provides insight into the apparent advantage of the end-to-end learning framework in its ability to accurately predict a cell's capacity trajectory. Second, we compared the end-to-end learning framework with a state-of-the-art model (i.e., "discharge model" in [20]) on cycle life prediction.

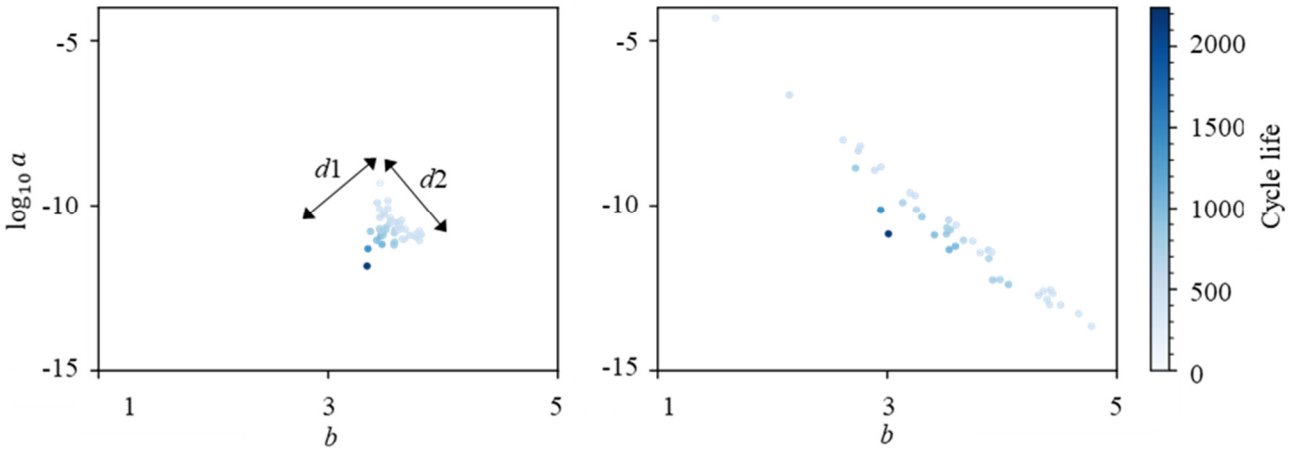
3.2.1. Capacity-Trajectory Prediction Performance

The proposed end-to-end learning framework was developed to enable early capacity-trajectory prediction. Similar early prediction models are only trained to estimate cell EoL at a specific capacity threshold, and therefore cannot be directly compared to the end-to-end learning framework in this area. The sequential optimization framework outlined in Section 2.2 is proposed as a baseline comparison model for the end-to-end framework. The first step in training the sequential optimization framework is fitting an empirical model to each cell's capacity fade curve. To show how accurate the curve-fitting procedure is, we included the curve-fitting RMSEs in Table 2 (see the "fitted empirical model"). Next, the elastic net machine learning model is trained to predict the fitted empirical model parameters by mapping from early-life features. However, for all but one of the test datasets, this two-step sequential process was found to introduce large errors, evident by the largest RMSE_Q and $\mathrm{RMSE}_{\mathrm{Cycle}}$ values in Table 2.

The poor performance of the sequential model is due to the two separately executed tasks used to train the model. First fitting the empirical model to the cells biases the sequential model to preferring a more accurate capacity-trajectory prediction. This overfitting of the empirical model parameters in the sequential model is evident when the empirical model parameters for the training dataset are plotted in the parameter space (Fig. 3b). The predicted empirical model parameters from the end-to-end learning model for the training dataset are shown in Fig. 3a for comparison. Specifically, we can see that the sequential optimization method overfit both parameters *a* and *b*, evident by the larger dispersion of the empirical model parameters on the less informative axis *d2* in Fig. 3b. We know this to be overfitting because the end-to-end learning framework achieved greater accuracy on all but one dataset and error metric in Table 2 while having a much smaller dispersion of fitted parameter values along the *d2* axis. The same observation can be made about the *d1* axis. The *d1* axis is shown to correlate with cell cycle life more strongly, and the end-to-end model was found to slightly increase parameter dispersion by about 8% in this direction. The overall smaller parameter value variance in the *d2* direction, and increased variance in the *d1* direction in Fig. 3a is evidence that the end-to-end model is more appropriately balancing the two objectives of fitting empirical model parameters and minimizing the elastic net regression weights. This improved balance between objectives was found to be the main reason why the end-to-end learning framework achieved much more consistent accuracy across the three datasets in Table 2.

Table 2. Capacity-trajectory prediction accuracy

Method	$RMSE_Q$			$RMSE_{Cycle}$		
	Train	Primary	Secondary	Train	Primary	Secondary
End-to-End	**0.041**	**0.037**	0.069	**67**	**104**	**135**
Sequential optimization	0.047	0.057	**0.047**	82	212	534
Fitted empirical model	0.013	0.012	0.015	29	44	70



(a) end-to-end framework (b) sequential optimization framework

Figure 3. Predicted empirical model parameters for training dataset

3.2.2. Cycle-Life Prediction Performance

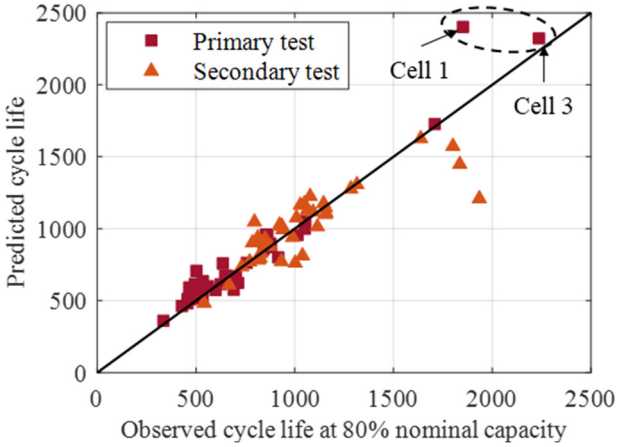
Neither the proposed end-to-end learning framework nor the sequential optimization framework directly estimates cell cycle life. However, based on the predicted capacity trajectory, we can derive the predicted cycle life at any capacity value by evaluating the predicted capacity fade curve. To be consistent with prior literature, the cycle life of a cell is defined as the cycle at which the cell reaches 80% nominal capacity. The nominal capacity for the cells in this study was reported to be 1.1 Ah, and therefore the EoL threshold is 0.88 Ah. For comparison, the derived cycle life prediction results are presented alongside an elastic net model that directly maps the early-life features to cell cycle life. The elastic net model is equivalent to the "discharge model" reported in [20].

As shown in Table 3, the end-to-end learning framework achieved accuracies on par with those of the elastic net for both the training and secondary test datasets. However, the error of the end-to-end model was much greater than that of the elastic net model for the primary test dataset. It is reasonable to expect that the elastic net model would achieve higher accuracies than the end-to-end model because the former does not have to consider fitting an empirical model, but this is shown to not necessarily be the case given the better performance by the end-to-end model on the other two datasets.

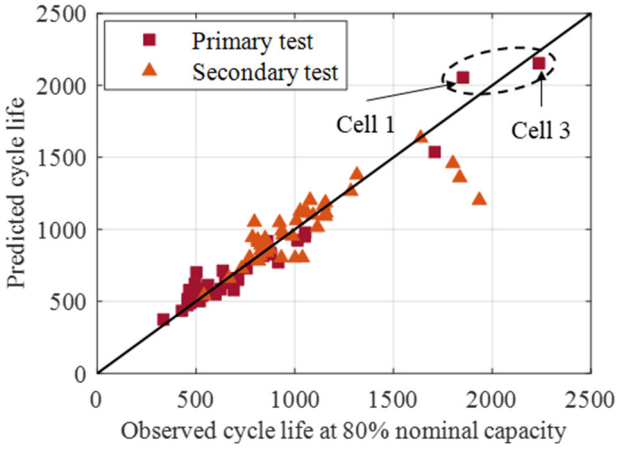
Table 3. Model errors for cycle life prediction

Method	$RMSE_{EoL}$			$MAPE_{EoL}$		
	Train	Primary	Secondary	Train	Primary	Secondary
End-to-End	**69**	112	**165**	10.0	10.4	9.2
Elastic net	75	**82**	173	**9.6**	**9.8**	**8.7**
Sequential optimization	88	327	769	12.5	16.9	55.3

To further investigate the accuracy discrepancy between the proposed end-to-end learning model and the elastic net model, we retrained the elastic net model to estimate cell cycle life at 90% nominal capacity and plotted the predicted cycle lives at both 80% and 90% nominal capacity in Fig. 4 and Fig. 5, respectively, and present the errors for the 90% nominal capacity prediction problem in Table 4. By analyzing the cycle life errors at different levels of degradation, we can get a better understanding of whether the large errors are a result of a larger modeling issue or a result of a few outlier datapoints in the dataset. Figures 4 and 5 show that the large accuracy difference between the models

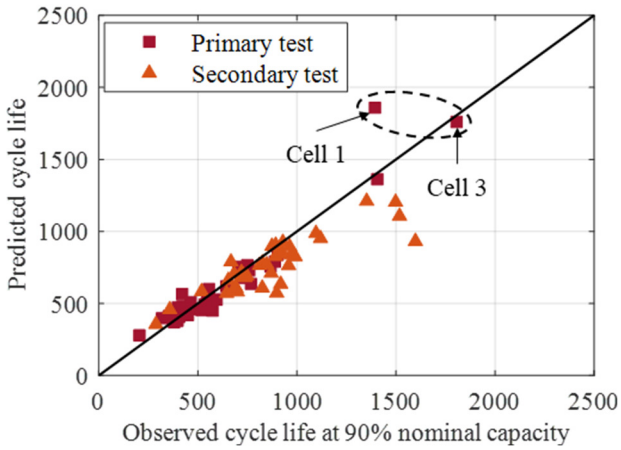


(a) End-to-end leaning framework

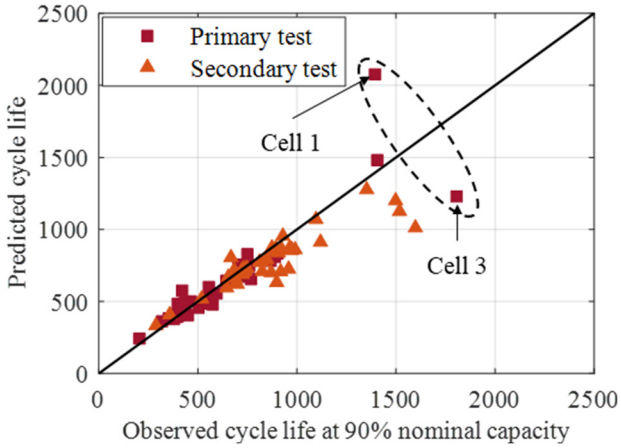


(b) Elastic net

Figure 4. Cycle life predictions for primary and secondary datasets (at 80% nominal capacity)



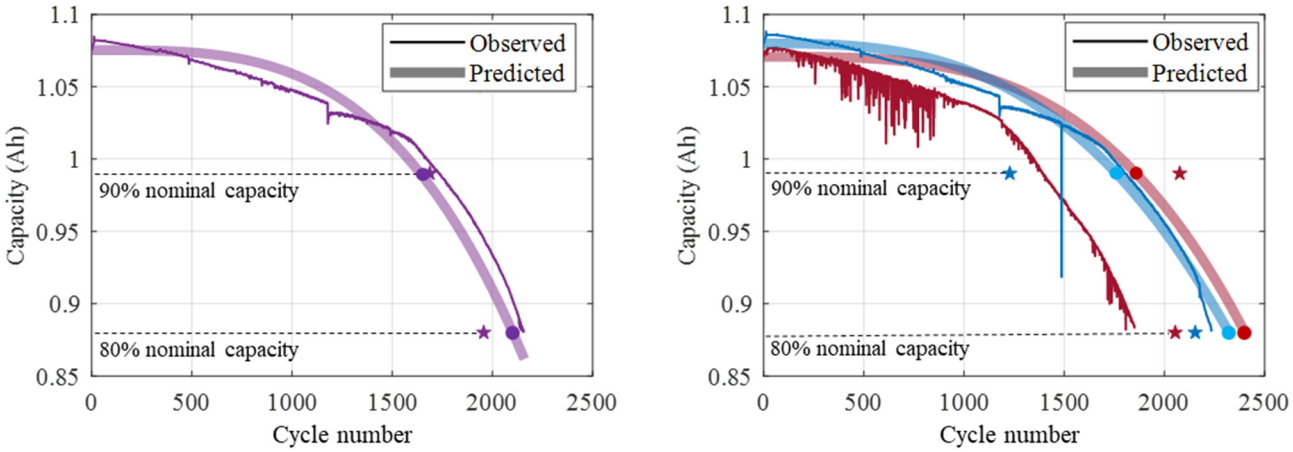
(a) End-to-end leaning framework



(b) Elastic net

Figure 5. Cycle life prediction for primary and secondary datasets (at 90% nominal capacity)

for the primary test dataset in Table 3 mainly comes from the predictions on just two cells: cell 1 and cell 3. These two cells have 80% normalized capacity cycle lives upwards of 2000 cycles. Both the elastic net model and the proposed end-to-end model struggled to predict the cycle lives of cells 1 and 3, regardless of the EoL criteria used. The high and sporadic errors observed for cells 1 and 3 in the primary test dataset are due to two main reasons. Firstly, the training dataset only contains a single cell (cell 2) with a similarly high cycle life to cells 1 and 3 in the primary test dataset. The lack of training data in the high cycle life region suggests that the trained models will have lower accuracy and less confidence in predicting cells in the high cycle life region, as seen in Figs. 4, 5 and 6. Second, cells 1 and 3 of the primary test dataset are observed to have a high level of noise in their capacity measurements, as shown in Fig. 6. It is for these reasons that we saw sporadic behavior in both models on the primary test dataset. Also, it can be observed that the end-to-end model was more consistent in its cycle life prediction for the high cycle life cells shown in Fig. 6. This is because the end-to-end framework was trained a single time to learn and predict the capacity trajectory. So, when it was evaluated at a higher EoL cut-off than 80%, it did not need to be retrained, and its cycle life prediction will always be found to decrease following the capacity fade model. The same cannot be said about the elastic net model. A good example of this issue is the predictions for cell 1 in Fig. 6b. The elastic net model was found to increase its 90% nominal capacity cycle life prediction relative to its 80% nominal capacity prediction. This sort of prediction violates the commonly reported physical phenomenon where the aging of a Li-ion cell causes the cell capacity to decrease monotonically with cycle and time.



(a) Cell 2 from the training dataset

(b) Cell 1 (red) and 3 (blue) in primary test dataset

Figure 6. Prediction for cells in the high cycle life region. Pentagrams represent the predicted cycle lives by elastic net model.

Table 4. Accuracy for cycle number prediction corresponding to 90% nominal capacity

Method	$RMSE_{EoL,90}$			$MAPE_{EoL,90}$		
	Train	Primary	Secondary	Train	Primary	Secondary
End-to-End	62	**93**	175	9.8	**10.5**	12.9
Elastic net	**55**	149	**155**	**8.6**	10.7	**10.3**

4. Conclusion

An end-to-end learning framework has been proposed for Li-ion battery early life prediction. The proposed method simultaneously fits an empirical model to estimate the capacity trajectory and trains a machine learning model to estimate the best-fit parameters of the empirical model using early-life features. The end-to-end learning framework advances the state-of-the-art in early life prediction by having the unique ability to predict a cell's entire capacity trajectory using only data from the first 100 cycles. Using the predicted capacity trajectory, the cycle life of a cell can be derived for any capacity threshold. The proposed end-to-end learning framework is shown to better balance the tradeoff between empirical model fitting accuracy and machine learning prediction accuracy than a baseline comparison framework which sequentially executes the separate tasks of fitting an empirical model and training a machine learning model. A comparative study is conducted on a publicly available battery dataset consisting of 124 lithium-iron-phosphate/graphite cells charged with various fast-charging protocols. The end-to-end learning framework is shown to achieve cycle life prediction accuracies on par with, or better than current state-of-the-art models, while also providing more physically compliant results. The proposed method shows promise in applications like Li-ion battery second-life use where predicting the entire capacity trajectory of a cell gives more insight into the current internal degradation of the cell and enables more accurate and timely retirement of cells from the field so that second-life use is maximized.

Acknowledgement

This work was supported in part by Iowa Economic Development Authority under the Iowa Energy Center Grant Number 20-IEC-018 and in part by the U.S. National Science Foundation under Grant ECCS-2015710. Any opinions, findings, or conclusions in this paper are those of the authors and do not necessarily reflect the views of the sponsors.

References

[1] C. Hu, H. Ye, G. Jain, and C. Schmidt, "Remaining useful life assessment of lithium-ion batteries in implantable medical devices," *J. Power Sources*, vol. 375, no. November 2017, pp. 118–130, 2018, doi: 10.1016/j.jpowsour.2017.11.056.

[2] Y. H. Lui *et al.*, "Physics-based prognostics of implantable-grade lithium-ion battery for remaining useful life prediction," *J. Power Sources*, vol. 485, no. November 2020, p. 229327, 2021, doi: 10.1016/j.jpowsour.2020.229327.

[3] J. Guo, Z. Li, and M. Pecht, "A Bayesian approach for Li-Ion battery capacity fade modeling and cycles to failure prognostics," *J. Power Sources*, vol. 281, pp. 173–184, 2015, doi: 10.1016/j.jpowsour.2015.01.164.

[4] I. Mathews, B. Xu, W. He, V. Barreto, T. Buonassisi, and I. M. Peters, "Technoeconomic model of second-life batteries for utility-scale solar considering calendar and cycle aging," *Appl. Energy*, vol. 269, no. January, p. 115127, 2020, doi: 10.1016/j.apenergy.2020.115127.

[5] X. Hu, L. Xu, X. Lin, and M. Pecht, "Battery Lifetime Prognostics," *Joule*, vol. 4, no. 2, pp. 310–346, 2020, doi: 10.1016/j.joule.2019.11.018.

[6] G. Ning, R. E. White, and B. N. Popov, "A generalized cycle life model of rechargeable Li-ion batteries," *Electrochim. Acta*, vol. 51, no. 10, pp. 2012–2022, 2006, doi: 10.1016/j.electacta.2005.06.033.

[7] V. Ramadesigan *et al.*, "Parameter Estimation and Capacity Fade Analysis of Lithium-Ion Batteries Using First-Principles-Based Efficient Reformulated Models Venkatasailanathan Ramadesigan," *Electrochem. Soc. Trans.*, vol. 19, no. 16, pp. 11–19, 2009.

[8] A. Guha and A. Patra, "Online Estimation of the Electrochemical Impedance Spectrum and Remaining Useful Life of Lithium-Ion Batteries," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 8, pp. 1836–1849, 2018, doi: 10.1109/TIM.2018.2809138.

[9] M. Broussely, S. Herreyre, P. Biensan, P. Kasztejna, K. Nechev, and R. J. Staniewicz, "Aging mechanism in Li ion cells and calendar life predictions," *J. Power Sources*, vol. 97–98, pp. 13–21, 2001, doi: 10.1016/S0378-7753(01)00722-4.

[10] W. Diao, S. Saxena, and M. Pecht, "Accelerated cycle life testing and capacity degradation modeling of LiCoO2-graphite cells," *J. Power Sources*, vol. 435, no. July, p. 226830, 2019, doi: 10.1016/j.jpowsour.2019.226830.

[11] B. Saha, K. Goebel, S. Poll, and J. Christophersen, "Prognostics methods for battery health monitoring using a Bayesian framework," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 2, pp. 291–296, 2009, doi: 10.1109/TIM.2008.2005965.

[12] W. He, N. Williard, M. Osterman, and M. Pecht, "Prognostics of lithium-ion batteries based on Dempster-Shafer theory and the Bayesian Monte Carlo method," *J. Power Sources*, vol. 196, no. 23, pp. 10314–10321, 2011, doi: 10.1016/j.jpowsour.2011.08.040.

[13] A. Pradeep Lall, B. Hao Zhang, and C. Rahul Lall, "PHM of state-of-charge for flexible power sources in wearable electronics with EKF," *IEEE Int. Reliab. Phys. Symp. Proc.*, vol. 2018-March, p. PSR.21-PSR.26, 2018, doi: 10.1109/IRPS.2018.8353695.

[14] E. Walker, S. Rayman, and R. E. White, "Comparison of a particle filter and other state estimation methods for prognostics of lithium-ion batteries," *J. Power Sources*, vol. 287, pp. 1–12, 2015, doi: 10.1016/j.jpowsour.2015.04.020.

[15] A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, and K. Dietmayer, "Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods," *J. Power Sources*, vol. 239, pp. 680–688, 2013, doi: 10.1016/j.jpowsour.2012.11.146.

[16] D. Wang, Q. Miao, and M. Pecht, "Prognostics of lithium-ion batteries based on relevance vectors and a conditional three-parameter capacity degradation model," *J. Power Sources*, vol. 239, pp. 253–264, 2013, doi: 10.1016/j.jpowsour.2013.03.129.

[17] R. R. Richardson, M. A. Osborne, and D. A. Howey, "Gaussian process regression for forecasting battery state of health," *J. Power Sources*, vol. 357, pp. 209–219, 2017, doi: 10.1016/j.jpowsour.2017.05.004.

[18] J. Liu, A. Saxena, K. Goebel, B. Saha, and W. Wang, "An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries," *Annu. Conf. Progn. Heal. Manag. Soc. PHM 2010*, pp. 1–8, 2010.

[19] Y. Zhang, R. Xiong, H. He, and M. G. Pecht, "Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5695–5705, 2018, doi: 10.1109/TVT.2018.2805189.

[20] K. A. Severson *et al.*, "Data-driven prediction of battery cycle life before capacity degradation," *Nat. Energy*, vol. 4, no. 5, pp. 383–391, 2019, doi: 10.1038/s41560-019-0356-8.

[21] W. Li, N. Sengupta, P. Dechent, D. Howey, A. Annaswamy, and D. U. Sauer, "One-shot battery degradation trajectory prediction with deep learning," *J. Power Sources*, no. xxxx, p. 230024, 2021, doi: 10.1016/j.jpowsour.2021.230024.

[22] P. M. Attia *et al.*, "Closed-loop optimization of fast-charging protocols for batteries with machine learning," *Nature*, vol. 578, no. 7795, pp. 397–402, 2020, doi: 10.1038/s41586-020-1994-5.

[23] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.