6-10-2024

Comparing life-cycle dynamics of Li-ion batteries (LIBs) clustered by operating conditions with SINDy

Kristen L. Hallas
The University of Texas Rio Grande Valley

Md Shahriar Forhad
The University of Texas Rio Grande Valley

Tamer Oraby
The University of Texas Rio Grande Valley

Benjamin Peters
The University of Texas Rio Grande Valley

Jianzhi Li
The University of Texas Rio Grande Valley

Recommended Citation

Kristen L. Hallas, Md Shahriar Forhad, Tamer Oraby, Benjamin Peters, and Jianzhi Li "Comparing life-cycle dynamics of Li-ion batteries (LIBs) clustered by operating conditions with SINDy", Proc. SPIE 13036, Big Data VI: Learning, Analytics, and Applications, 1303607 (10 June 2024); https://doi.org/10.1117/12.3013519

Comparing Life-cycle Dynamics of Li-Ion Batteries (LIBs) Clustered by Operating Conditions with SINDy

Kristen L. Hallas[a], Md Shahriar Forhad[a], Tamer Oraby[a], Benjamin Peters[a], and Jianzhi Li[a]

[a]University of Texas Rio Grande Valley, 1201 W University Drive, Edinburg, United States

ABSTRACT

Lithium-ion batteries (LIBs) play a big part in the vision of a net-zero emission economy, yet it is commonly reported that only a small percentage of LIBs are recycled worldwide. An outstanding barrier to making recycling LIBs economical throughout the supply chain pertains to the uncertainty surrounding their remaining useful life (RUL). How do operating conditions impact initial useful life of the battery? We applied sparse identification of nonlinear dynamics method (SINDy) to understand the life-cycle dynamics of LIBs with respect to sensor data observed for current, voltage, internal resistance and temperature. A dataset of 124 commercial lithium iron phosphate/graphite (LFP) batteries have been charged and cycled to failure under 72 unique policies. Charging policies were standardized, reduced to PC scores, and clustered by a k-means algorithm. Sensor data from the first cycle was averaged within clusters, characterizing a "good as new" state. SINDy method was applied to discover dynamics of this state and compared amongst clusters. This work contributes to the effort of defining a model that can predict the remaining useful life (RUL) of LIBs during degradation.

Keywords: sparse regression, nonlinear dynamics, model discovery, degradation modeling, deep learning

1. INTRODUCTION

The Bipartisan Infrastructure Deal introduced by the White House in 2021, has the goal to reduce emissions by 50% by 2030 and become net-zero by 2050. Greenhouse gas emissions from transportation accounted for 28% of total U.S. greenhouse gas emissions in 2022 by economic sector, making it the largest contributor of U.S. greenhouse gas emissions.[1] The electric vehicle (EV) is one solution to lowering U.S. greenhouse gases. There is advancement in all areas of STEM to make EVs sustainable. Yet, what currently powers an EV, the lithium-ion (Li-Ion) battery, also contributes greatly to emissions.

An ideal model of treating retired end-of-life Li-Ion batteries (LIBs), proposed in 2021, suggests to reuse, recycle, then dispose of LIBs.[2] Reusing the LIBs for leftover energy capacity requires remanufacturing the LIBs via repairing and/or refurbishing or repurposing them for second-life purposes, such as in grid-connected storage, backup supplies, or electrical tools. Reusing LIBs involves tasks similar to the pre-treatment phase of the recycling process, such as inspection, disassembly and testing. Primary recycling methods for LIBs are ineffective at preserving lithium and alternatives are not yet effectively scalable for large-scale production. These factors contribute to the reason why, in practice, global recycling rates of LIBs are less than 5%. Compared to emissions from recycling techniques, the use of new lithium cobalt oxide (LCO) material for LIBs contributes significantly to carbon emissions.[3] Improper recycling of LIBs has other, severe environmental consequences, resulting in fires in solid waste systems, irreversible damage to residential recycling systems, and heavy metals like cobalt, copper, nickel and aluminum into soils, adversely affecting both the land and human health.[4] For EVs to be environmentally viable, alternative approaches to recycling LIBs must be developed and implemented.

Lead-acid batteries, one of the most successful examples of a circular economy, work because recycling is economical for the supply chain.[5] Incorporating recycling into the model of LIB manufacturing requires an understanding of the current state of recycling: how recycling takes place, how frequently it takes place, and how profitable it is. Different recycling methods may have different percent yields, depending on the LIB state-of-health (SOH). Consequently, the degradation of LIBs must be understood with more certainty.

Further author information: (Send correspondence to K.L.H.)
K.L.H.: Email: kristen.hallas01@utrgv.edu

Big Data VI: Learning, Analytics, and Applications, edited by Panos P. Markopoulos,
Proc. of SPIE Vol. 13036, 1303607 · © 2024 SPIE
0277-786X · doi: 10.1117/12.3013519

To contribute to this boundary of knowledge, a model of battery degradation under different charging conditions would clarify which performance indicators of LIBs correlate to higher percent yield of recovered lithium material. Moreover, governing equations that describe how different battery characteristics interact with each other over the life cycle of a battery, could greatly inform under what conditions such indicators could be maximized. With that said, determining the exact governing equations can be incredibly challenging. Likewise, obtaining the precise measurements for well-defined equations can be impractical, if not impossible to achieve, with traditional measurement instruments. Employing the sparse identification of nonlinear dynamics (SINDy) method,[6] we derive governing equations using the sequentially thresholded least squares (STLS) algorithm. We apply the algorithm to the measurement data of 124 lithium iron phosphate/graphite (LFP) batteries,[7] where each battery was charged and cycled to failure under 72 unique policies, defined by three variables. Cycle life ranged from 150 cycles to 2000+ cycles.

An outline of this work is as follows. Methodology related to dynamical systems will be presented as it relates to this application area, emphasizing the formulation of the SINDy method. An outline of data set details will be provided. Analysis was conducted in MATLAB, including the pre-processing, clustering of data policies and application of SINDy. Resulting models for various combination of LIB metrics with this methods will be shared and compared between clusters. The conclusion will remark on observations discovered during the parameter optimization as well as future research directions involving other types of methodology, including extensible versions of the SINDy method. A code appendix will be linked at the end, which is written in the MATLAB programming language, as well as relevant references.

2. REVIEW OF METHODS

Dynamical systems and the partial differential equations (PDE) that govern them describe physical phenomena in nearly every discipline.[8] Governing equations of dynamical systems specify exact interactions among each relevant quantity, as well as how these quantities and their interactions evolve over time. Many dynamics to systems are already well known, such as in classical mechanical systems, electrical circuits and turbulent fluids. In other fields, such as climate science, neuroscience, and epidemiology, the dynamics are difficult to determine.

Some dynamical systems with unknown governing equations, such as understanding cognition from neural recordings, or inferring patterns in climate, are inherently data-driven problems. With an excess of data and computational power, there have been several advances in automating the discovery of these governing equations for dynamical systems. Schmidt and Lipson successfully found an algorithm to identify Hamiltonians, Lagrangians, and other momentum conservation laws from observation data.[9] Given a set of experimental data points from a physical system, one may calculate the partial derivatives between variables from the data. Candidate symbolic functions are randomly generated in an exhaustive search to describe the physical invariance between observed variables. To measure how well a given equation describes an invariance, the same partial derivatives can be derived symbolically and compared with the observed data to select the most parsimonious equations.

Interestingly, this discovery process could determine the previous laws of simpler systems and employ them to explain more complex dynamics; however, this method had a computationally expensive solution convergence, even after parallelization. This algorithm for symbolic regression cost exponential time based on the complexity of the expression and quadratic time based on the number of variables in the system. Though this approach did not scale well to larger systems, it served to motivate another method of discovering system dynamics.

2.1 Finding dynamics via Spare Identification of Nonlinear Dynamics (SINDy)

In their work, Brunton and Kutz re-imagine the dynamical system discovery problem using recent advances in sparse regression and compressed sensing.[6] By considering the fact that most physical systems use only a few terms to define dynamics, one can assume that governing equations must be sparse in a high-dimensional nonlinear function space. Through a demonstration via the Lorenz system, the authors demonstrate how the governing equations that underlie a physical system can be derived, based on data that may be realistically collected in simulations or experiments. Sparse regression can determine which right hand side terms are nonzero, without performing a computationally intractable, combinatorial brute-force search. The resulting solution to identify a nonlinear model inherently balances the model complexity with accuracy. The sparse regression formulation requires a penalization of the number of terms in the dynamics that scales well to large problems.

This is a general process for SINDy, as outlined by Brunton and Kutz.[6] First, let the governing equation for a physical system be defined as:

$$\frac{d}{dt}x = \boldsymbol{f}(\boldsymbol{x})$$

For solving sparse regression formulation with the SINDy method, one approximates f in the above equation by the following generalized linear model:

$$y = \boldsymbol{f}(\boldsymbol{x}) \approx \sum_{k=1}^{p} \theta_k(\boldsymbol{x})\ \xi_k = \boldsymbol{\Theta}(\boldsymbol{x})\ \xi_k$$

Here, we assume measurements $y \in \mathbb{R}^m$ are a linear combination of columns from a feature library $\boldsymbol{\Theta}(\boldsymbol{x}) \in \mathbb{R}^{m \times p}$. A linear combination of active functions in the feature library can be expressed by vector $\boldsymbol{\xi} \in \mathbb{R}^p$. For a sparse solution, the objective is to solve for which terms of $\boldsymbol{\xi}$ are active in the dynamics via sparse regression, minimizing the terms ξ_k such that model parsimony is maintained.

First, to build the $\boldsymbol{x}$ matrix, construct the data matrix from observed, time series data such that:

$$X = \begin{bmatrix} x\,(t_1) & x\,(t_2) & \dots & x\,(t_m) \end{bmatrix}^T$$
$$\dot{X} = \begin{bmatrix} \dot{x}\,(t_1) & \dot{x}\,(t_2) & \dots & \dot{x}\,(t_m) \end{bmatrix}^T$$

With this construction, one can compute $\dot{x}$ directly from data in $\boldsymbol{x}$. The total-variation regularized derivative is numerically robust for noisy data. From this formulation, it follows that:

$$\frac{d}{dt}x = f(x) \text{ yields } \dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\ \boldsymbol{\Xi}$$

Here $\boldsymbol{\Theta}(\boldsymbol{X})$ represents the library of candidate nonlinear functions. These can be a combination f polynomial functions of the observed variables of $\mathbf{x}$, interactions between observed variables of $\mathbf{x}$, trigonometric functions, logarithm functions, and other nonlinear functions. To find a model with as few terms as possible in $\boldsymbol{\Xi}$, calculate:

$$\xi_k = \text{argmin of } \xi_k' \left(\left\| \dot{X}_k - \Theta(X)\ \xi_k' \right\|_2 + \lambda \left\| \xi_k' \right\|_1 \right)$$

In this expression, $\dot{X}_k$ is the kth column of $\dot{X}$, whereas λ is a hyperparameter to optimize sparsity in the model. Above, the Sequential Thresholded Least-Squares (STLS) algorithm has been applied to minimize ξ_k. Other convex convergence algorithms can be selected, such as the Least Absolute Shrinkage and Selection Operator (LASSO).[10] A solution to this optimization problem yields the solution, a set of expressions for the change in each given measurement variable. The one dimensional case may be expressed as:

$$\dot{x}_k = \boldsymbol{\Theta}(\boldsymbol{x})\ \xi_k$$

Here, $\dot{x}_k$ is the derivative of x_k which is k-th element of x, $\boldsymbol{\Theta}(\boldsymbol{x})$ is the row vector of symbolic functions of $\mathbf{x}$, and ξ_k is a vector of scalars. Selecting a convex convergence algorithm is critical to scale the process well with large data. Likewise, one must take discretion in considering a dataset for this method. Measurement error and numerical round-off can lead to every library term being included in the algorithm, resulting in an optimized expression with no sparsity.

Since λ is a sparsity-promoting knob, it can be modified to arrive at a more parsimonious solution. As $\lambda \to \infty$, a trivial solution $\dot{x} = 0$ is yielded, whereas $\lambda = 0$ is a simple least-squares solution, considering all terms in the candidate library. To find the best expression, one may consider different λ values via an optimization algorithm such as trial-and-error, grid or random search, gradient-based or Bayesian optimization, or metaheuristic algorithms.[11] Pareto-efficient solutions can be selected, evaluated against number of active terms of the solution and a model assessment statistic, such as Adjusted R^2 or Mallow's Cp, The latter metric is an alternative to the AIC (Akaike Information Criterion), suitable for Gaussian linear regression.[12]

For nonlinear SINDy methods, since general rational functions are not sparse linear combinations of basis functions, one can reformulate the dynamics as implicit ordinary differential equations. Though this formulation is nonconvex and sensitive to noise, recent algorithms have been proposed to scale solutions for high-dimensional systems,[13] as well as an application to partial differential equations (PDEs).[14] Likewise, there have been many advances to modify SINDy algorithms for control,[15] parallelism,[16] sensitivity analysis,[17] bagging,[18] mixed integer optimization,[14] and uncertainty quantification.[19]

2.2 Finding dynamics: alternative approaches

There are other methods to discover dynamics from measurement data, such as Neural Ordinary Differential Equation (ODE) and Physics Informed Neural Networks (PINNs). These will be briefly presented in connection to future work. Feed forward neural networks have input nodes, hidden layers, output nodes, and a loss function. Recurrent neural networks (RNN) have hidden states, instead of hidden layers, to better understand changes over time. Some examples of RNN include long-short term memory networks and residual networks.

$$h_{t+1} = h_t + f(h_t, \theta_t), \text{ where } t \in \{0, \ldots, T\} \text{ and } h_t \in \mathrm{R}^D$$

Neural ODE is an alternative approach to model discovery,[20] taking the architecture of a residual network and considering a Euler discretization of continuous transformation. By adding more layers with smaller step sizes, continuous dynamics can be parametrized with an ODE specified by a neural network. The output layer is a solution to the initial value problem (IVP) at some time T, in essence, creating a black box ODE solver.

$$\frac{d\boldsymbol{h}(t)}{dt} = f(\boldsymbol{h}(t),\ t,\ \theta)$$

Neural ODEs have been shown to fit and extrapolate time series data.[20] Likewise, PINNs[21] is another alternative approach to model discovery. Physics Informed Neural Networks (PINNs), which are rapidly growing in popularity, are another supervised learning method that exists to discover solutions of nonlinear PDEs. They can consider either continuous time or discrete time formulations, which approach a solution via spatio-temporal function approximators or implicit Runge-Kutta time stepper schemes,[20] respectively. To discover the dynamics from measurement data, one can embed the entirety of measurement data into the PINNs loss function, then employ regularization techniques with a threshold that drops out insigificant terms from the model.

PINNs works well by embed the residual of the PDE in the loss function of the network. With that said, this is not a one-size-fits-all model to solving all nonlinear PDEs. For example, the operators of fractional PDEs cannot be automatically differentiated. Likewise, symmetries are not easily discovered, such as in the 1D nonlinear Schrödinger (NLS) problem.[21] Since the boundary conditions of PDEs are difficult to implement and manage, especially over time, this leads to a higher error rate. Thus, for PDEs with solutions that are multiscale, chaotic, and turbulent, alternative approaches may be necessary.

One possible alternative is to create a hybrid neural network,[22] which combines one or more different types of deep learning structures. This hybrid topology is capable of achieving accurate results with reduced computational time when optimizing the parameters of PDEs, such as the diffusion equation. One motivation for the combination of neural network architecture lies in the fact that each neural network subtype has its own benefits. For example, the Autoencoder (AE) Neural Network is an effective nonlinear extension of principal component analysis (PCA). In contrast, the Convolutional Neural Network (CNN) has strengths in its construction consisting of sparse interactions, where the output units only interacts with a subset of inputs. Likewise, the parameter-sharing in the CNN means that same kernels used for different parts of the inputs, reducing computations. The Long Short-Term Memory (LSTM) Neural Network, a subset of the Recurrent Neural Network (RNN), can easily detect short and long-term dependencies in time series data. This strategy has been employed in similar engineering problems, such as to estimate remaining life of components for predictive maintenance.[22]

Ultimately, the CNN-LSTM architecture could optimize dynamical systems parameters for governing PDEs with a spatial-temporal representation, and particularly well for nonlinear PDEs. This architecture is one possibility beyond PINNs for applying the dynamics of a system to optimize a solution. By modelling some dynamical system as a stream of changing images, one could solve for the dynamical system parameters with hybrid neural network architecture. The most important factor is identifying the correct governing equation.

One more interesting idea is to specify temporal degradation as Brownian motion process to describe the degradation of lithium-ion batteries (LIBs),[23].[24] Identifying a theoretical governing equation and comparing it to equations discovered by SINDy, or a variant of SINDy with uncertainty quantification,[19] could given a specific band range in which data is obtained. Both these equations could be fed into some sort of hybrid neural network architecture and optimized in this way to determine a model of LIB salvagability. Furthermore, these equations can be validated through the neural ODE and PINNs approach.

This paper focuses on preliminary results of obtaining governing equations from a vanilla SINDy method. Due to the size and variability of the application data set, care must be taken to ensure that results adequately express the dynamics of the LIB life cycle, on average, with respect to differences in charging policies.

3. RESULTS

We seek to understand the dynamics of LIBs during their life cycle, characterizing the initial state and studying how it evolves over time. The dataset[7] used consists of 124 LIBs that have been cycled to failure under various fast-charging conditions. Experiments were done in 3 batches, which took place from May 2017 to April 2018. The lithium-ion phosphate (LFP)/graphite cells were commercially available, and manufactured by A123 Systems (APR18650M1A). Cells have a nominal capacity of 1.1 Ah and a nominal voltage of 3.3 V.

All LIBs were charged with a one-step or two-step fast-charging policy. Data owners provided the formatting "C1(Q1)-C2", which denotes C1 and C2 as first and second constant-current steps respectively, and Q1 value serving as the state-of-charge percentage for which the currents switch. The second current step ends at 80% state-of-charge (SOC), and then the cells charge at 1C constant current, constant voltage. To stay consistent with manufacturer, upper and lower cutoff potentials are 3.6 V and 2.0 V, respectively. All cells discharge at 4C. The method of CC-CV charging regulates the current and voltage for charging, supplying constant electric current at a specified ampere, and charging at a set voltage to maintain constancy.

Batteries were cycled in horizontal cylindrical fixtures on a 48-channel Arbin LBT potentiostat. The temperature chamber with forced convection was set to 86°F. Internal resistance measurements were collected during charging at 80% state-of-charge by averaging 10 pulses of ±3.6C with a pulse width of 30 ms or 33 ms, depending on the time of the experiment. Temperature metrics were captured by stripping a portion of the battery plastic insulation and attaching a Type-T thermocouple with thermal epoxy and Kapton tape to the exposed cell can. Since contact between the thermocouple and the cell can vary, and since the thermocouple may lose contact during cycling, temperature measurements cannot be considered perfectly reliable.[7]

For each battery, the following measurement data was observed:

- **Voltage**: difference in charge between two points
- **Current**: rate at which charge is flowing
- **Internal Resistance**: value of how current is resisted
- **Temperature**: temperature of the cell
- **Discharge Capacity**: the charge that is capable of coming out of the cell
- **Charge Capacity**: the charge that is capable of going into the cell

Each of these sensors sampled data roughly 1000 times per cycle, though an exact sampling rate varied between observations. Cycles for batteries ranged from 150 to over 2000 cycles. In applying SINDy to this data, we begin by importing the measurement data for the variables. Clustering of the various policies was performed, followed by an application of SINDy, which is presented in the remaining sections of this work.

3.1 Clustering

In this data set,[7] cells were charged and discharged with 72 unique fast-charging policies. The manufacturer provided a recommended protocol of 3.6C, constant current-constant voltage (CC-CV). The first step applied C1 over the given state-of-charge rate specified by Q1. Once the state-of-charge Q1 was reached, the second step C2 was applied to a constant state-of-charge 80% for all battery cells. Likewise, all cells were discharged with a CC-CV discharge at 4C to 2.0V. To give an example, a policy of C1=4.65, Q1=69%, and C2=6 means that the cell was a 4.65C charging step from 0% to 69% state-of-charge (SOC), then a 6C charging step from 69% to 80& SOC, then a 1C charging step to 100% SOC.

Principal component analysis (PCA)[25] was performed on three variables, the C1, Q1, and C2 values, after standardization, to better understand the variance between charging policies. The sum of the sample variances of all individual variables was used to obtain a fraction of variance explained by each principal component. The fraction of variance explained by the first PC score was 52.88%, and by the second PC score was 35.93%. Thus, the first two PC scores explained 88.81% of the variance of the charging policies.

The first two PC scores were utilized to cluster policies into groups by a k-means algorithm. Optimal k values were scanned from $k = 2$ to $k = \text{ceil}(\sqrt{140}) = 12$, where 140 is the number of observations in the data set. The elbow method[26] and gap statistic[27] verified an optimal $k = 10$ clusters. The evaluation metrics were plotted for each k and can be seen in Figure 1 for each method.

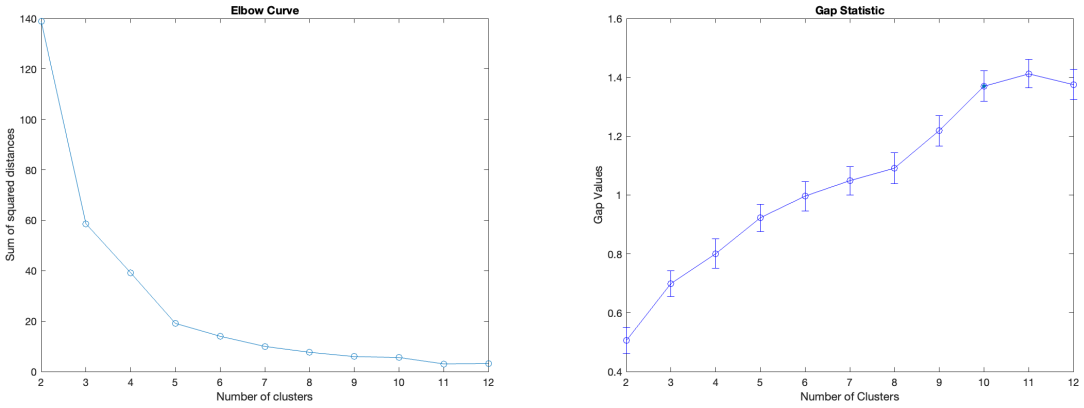

Figure 1. Combining the elbow method and gap statistic, an optimal $k = 10$ clusters was identified.

Plotting the PC scores on a scatter plot in Figure 2, one can see a clear boundary between each cluster.

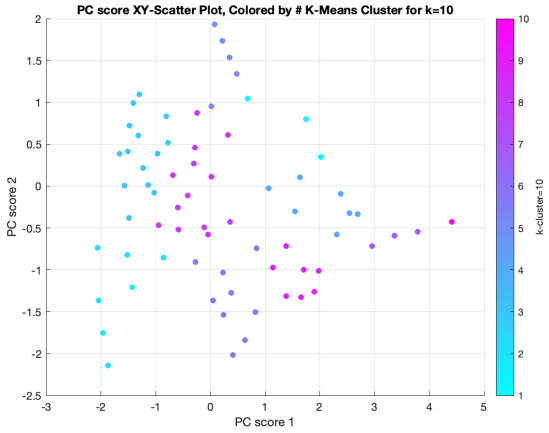

Figure 2. Each set of policies are contained within a boundary, identified with a k-means algorithm for $k = 10$.

A distribution of the cycle life of the cells in each cluster can be seen in Figure 3.

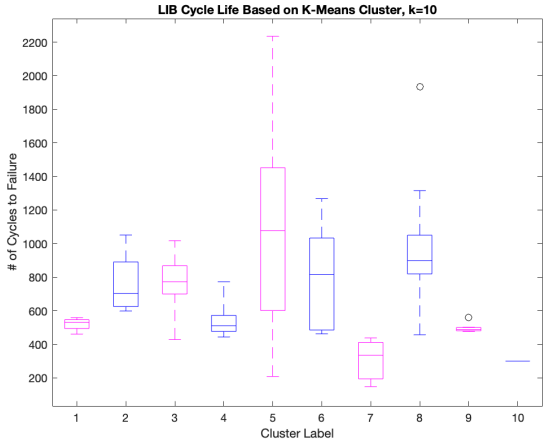

Figure 3. Differences between LIB life cycle can be seen across clusters, ranging from median cycle length 300 to 1078.

Clustering was further validated by plotting the initial current for all cells in the cluster - see Figure 4. Each cell was sampled approximately 1000 times, without a standardized sampling rate. For this reason, there was a lack of perfect overlap in some clusters. Though phenomenon was more notable in the clusters with larger sample sizes, other clusters, such as cluster 3, had a near perfect overlap of current behavior.

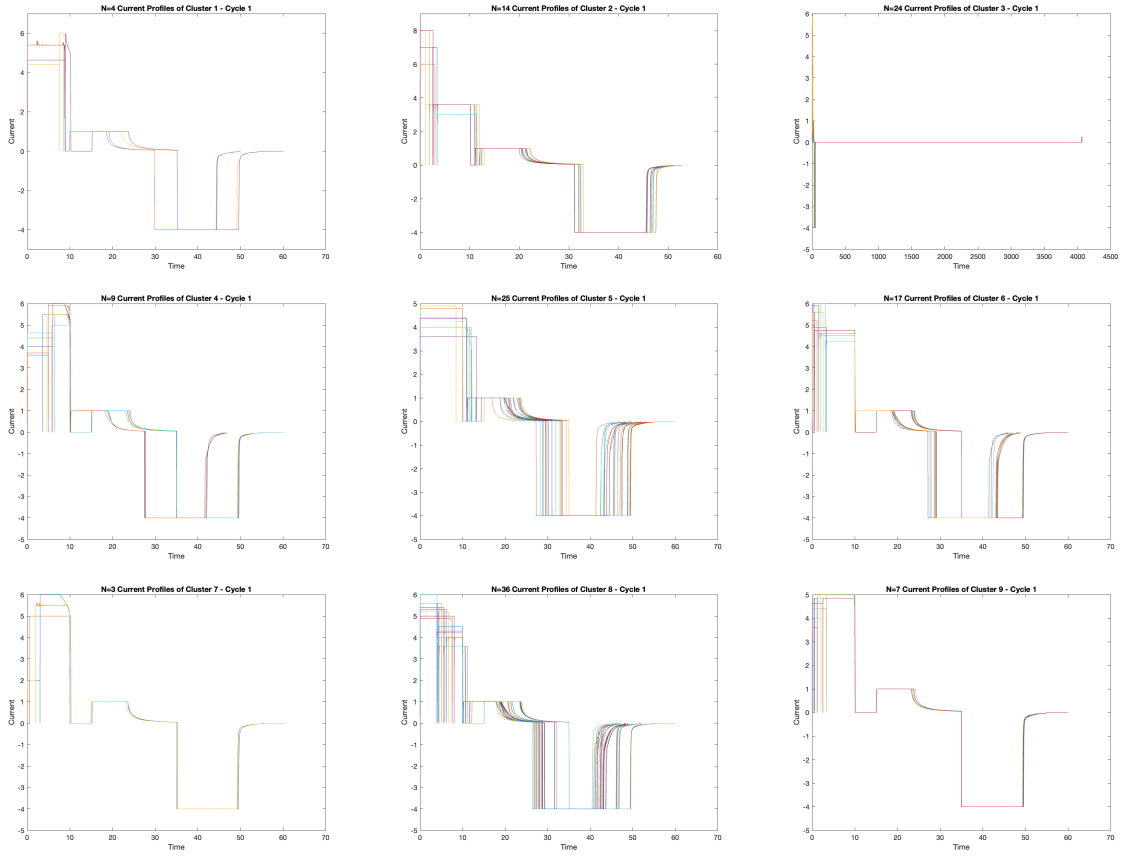

Figure 4. Some clusters have a nearly perfect overlap of current behavior. Cluster 10 had just one cell, thus was excluded.

From each cluster, we found the cell closest to the median as given by the box plot in Figure 3. We note the cell representing the policy cluster in Table 1, presented below.

Cluster #	C1 Value	Q1 Value	C2 Value	Cycle Length	Batch #
1	4.65	69%	6	527	2
2	8	25%	3.6	702	1
3	6	60%	3	757	1
4	3.6	30%	6	511	2
5	4.8	80%	4.8	1078	3
6	5.6	19%	4.6	816	3
7	2	7%	5.5	335	2
8	5.4	50%	3.6	902	1
9	4.65	19%	4.85	489	3
10	1	4%	6	300	2

Table 1. One battery was selected from each cluster to serve as a representative of the charging policy and median cycle length within that cluster. Batch 1 of the experimental data was represented in 3 clusters, batch 2 was represented in 4 clusters, and batch 3 was represented in 3 clusters. Note that cluster 10 consists of a single observation.

Moreover, we observe that in the data set,[7] there were three batches of experimental data. The experiment number is also displayed in Table 1, showing an adequate representation of all experimental data. Each batch has various notes associated with the experimental procedure data collection. Most notably, batch 1 did not have data for cycle 1, so any current profiles from that batch were for cycle 2, which is another potential reason for lack of overlap within some clusters. For example, cluster 5 with 25 cells had 8 cells from batch 1, whereas cluster 3 had a majority of cells, 16 out of 24, from batch 1 of the experiment.

For each cluster, we will now model multivariate sensor profiles using the equations discovered by SINDy. We applied SINDy to a state dimension = 4, consisting of the measurements for capacity discharge, voltage, current, and temperature. In the candidate library, we considered polynomial terms of measurements and interaction terms between measurements. Results are presented in the next section, first considering only the 1st cycle of data, then considering the data over all cycles of the representative cell.

3.2 Sparse regression: 1st cycle

To assess battery state-of-health, we seek to first characterize the initial state of the battery, as degradation can be viewed as deviation from this state. However, this is not trivial since batteries that are identically manufactured may still be subjected to various charging conditions. Therefore, we begin by investigating the effect of charging conditions on the initial state of LIBs. Again, time steps varied for each cluster based on the variation in sensor measurement sampling rates for each observation.

For each variable, derivative was calculated using the gradient function in MATLAB. SINDy is applied as a sparse regression, applying the Sequential Thresholded Least-Squares (STLS) algorithm. SINDy was performed in 4 state dimensions, and inter-cluster analysis is provided below. For notation sake, we will use the following

- Q_d: capacity discharge
- v: voltage
- c: current
- T: temp

Because of Ohm's law,[28] voltage = current $\cdot$ internal resistance, internal resistance was excluded from the model to prevent confounding effects. Polynomial terms and interactions went up to degree 2. For the initial analysis, additional functions in the candidate library were excluded from the model. To consider the initial state, sparse regression was performed for only the 1st cycle of each cell. The hyper-parameter, sparsity-promoting knob λ, was initiated at 0.025 and modified on an as-needed basis to arrive at the most parsimonious solution.

We now present the dynamics of the first life cycle of each representative battery:

$$\dot{c}_1 = 2.4927 - 1.3485Q_d - 1.4084v - 0.1568Q_d^2 - 0.391Q_dv - 0.1977v^2 - 0.0390vT$$

$$\dot{c}_2 = -7.9467 + 1.3948Q_d + 2.3285v + 0.2571T + 0.2896Q_d^2 - 0.2999Q_dv - 0.0767v^2 - 0.0754vT$$

$$\dot{c}_3 = 0.3139 - 0.0674Q_d - 0.0940v$$

$$\dot{c}_4 = 1.9641 - 1.0403Q_d - 1.1003v - 0.1090Q_d^2 - 0.3029Q_dv + 0.1528v^2$$

$$\dot{c}_5 = -44.5046 + 10.9731Q_d - 0.3594c + 13.333v + 1.4053T + 1.6127Q_d^2 + 0.1393Q_dc + 1.3933Q_dv - 0.4982Q_dT$$
$$+0.1082cv - 0.0989v^2 - 0.4105vT$$

$$\dot{c}_6 = -7.0566 + 1.0322Q_d + 1.5413v + 0.2709T + 0.2467Q_d^2 + 0.4537Q_dv - 0.078Q_dT + 0.1392v^2 - 0.078vT$$

$$\dot{c}_7 = 0.2405 - 0.071v - 0.058_d^2, \ \dot{T}_7 = 0.5830 - 0.3923Q_d - 0.3926v + 0.1533Q_dv + 0.0646v^2$$

$$\dot{c}_8 = 0.342 - 0.1965Q_d - 0.1029v^2 + 0.0496Q_dv,$$

$$\dot{T}_8 = 4.5569 - 1.6739Q_d - 1.4309v - 0.1285T - 0.1285T - 0.2159Q_d^2 + 0.0556Q_dT + 0.0351v^2 + 0.0369vT$$

$$\dot{c}_9 = -4.0168 + 0.6953v + 0.1874T + 0.2927Q_d^2 + 0.4668Q_dv - 0.052Q_dT + 0.1377v^2 - 0.0547vT,$$

$$\dot{T}_9 = -0.1568Q_d + 0.0638Q_dv$$

$$\dot{c}_{10} = -4.5651 + 0.3491Q_d + 1.5677v + 0.1367T - 0.0941Q_d^2 - 0.1012Q_dv - 0.0766v^2 - 0.0390vT$$

Equations discovered by SINDy render an expression with respect to the change in current expressed as a change in discharge capacity, voltage, and temperature. Subscript of the PDE represent the cluster number associated with the expression. Most expressions were expressed simply as $\dot{c}$ though active terms and signs/magnitudes to the coefficient varied for each policy. This indicated a difference in the dynamics across clusters. Moreover, some policies were expressed as a combination of $\dot{c}$ and $\dot{T}$.

We can plot the gradient of actual measurement data against the solution of the differential equation generated by SINDy, to check the results of the discovered equation. An example of this can be seen in Figure 5.

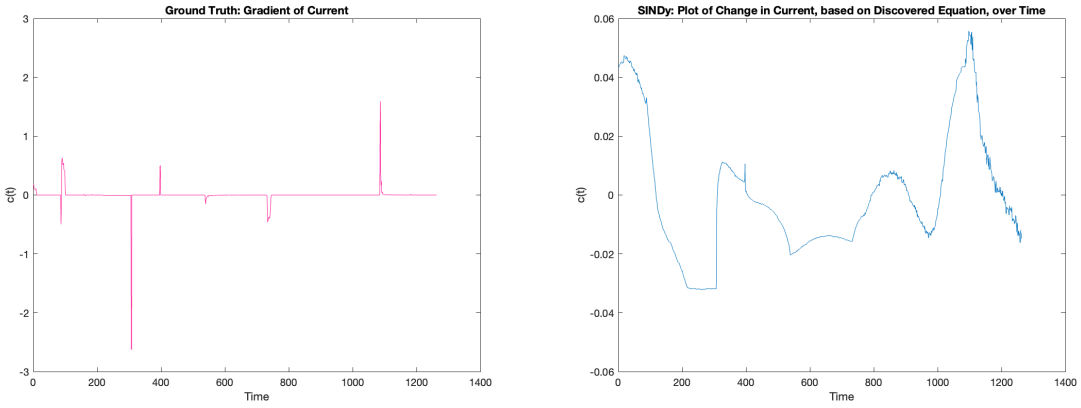

Figure 5. Results of SINDy applied to the single observation in cluster 10 to express the current behavior for cycle 1. On the left is the ground truth, based on the computed gradient for current. On the right is a plot of the expression derived by SINDy. Though magnitudes differ, peaks and valleys of ground truth mirror the SINDy plot in early time steps.

This indicates that the λ value could need some tweaking in order to arrive at a parsimonious model with respect to the overall accuracy of understanding the change in the observed variable.

3.3 Sparse regression: all cycles

Using the MATLAB gradient() function, we compute the derivative for each variable. From there, we build the library of candidate functions, considering polynomials up to various orders. Again, the hyper-parameter lambda was primarily set to 0.025 and modified on an as needed basis, depending on the sparsity of the solution.

To process a sparse regression for all cycle data, one must observe the profile data for the entire cycle life dynamics of the battery representative of each cluster. To start, we flatten the observations per cycle to a vector consisting of the entirety of cycle data, each sampling the sensor data about 1000 times per cycle. For each cycle, points were sampled a certain amount of time, and this time is the same for each sensor for that cycle. For the next cycle, more points may have been sampled, but the number of sampling points are the same for each sensor within the cycle.

We now present the dynamics of the entire life cycle of each representative battery.

$$\dot{c}_1 = 0.3543 - 0.2112Q_d - 0.1038v + 0.0453Q_dv,\ \dot{T}_1 = -0.1649Q_d + 0.0685Q_dv$$

$$\dot{c}_2 = 0.4903 - 0.466Q_d - 0.1461v + +0.0796Q_d^2 0.1086Q_dv,\ \dot{T}_2 = -0.0855Q_d - 0.0517Q_d^2 + 0.0586Q_dv$$

$$\dot{c}_3 = 0.6969 - 0.42Q_d - 0.3317v + 0.044Q_d^2 + 0.1144Q_dv + 0.0363v^2,\ \dot{T}_3 = -0.0899Q_d^2 + 0.0378Q_dv$$

$$\dot{c}_4 = 1.1438 - 0.6596Q_d - 0.6336v + 0.0908Q_d^2 + 0.1852Q_dv + 0.087v^2,$$

$$\dot{Q}_{d4} = -0.2886 + 0.2135 + 0.1721v - 0.0568Q_d^2 - 0.0561Q_dv - 0.0256v^2$$

$$\dot{c}_5 = 1.2816 - 0.5354Q_d - 0.7046v + 0.1686Q_dv + 0.0947v^2,\ \dot{T}_5 = -0.1453Q_d + 0.0617Q_dv$$

$$\dot{c}_6 = 2.0138 - 1.0774Q_d - 1.1256v + 0.1101Q_d^2 + 0.3178Q_dv + 0.1551v^2,\ \dot{T}_6 = -0.1994Q_d + 0.0853Q_dv$$

$$\dot{c}_7 = 0.1949 - 0.0563Q_d - 0.0569v,\ \dot{T}_7 = -0.1335Q_d^2 + 0.0503Q_dv$$

$$\dot{c}_8 = 0.6822 - 0.3097Q_d - 0.3396v + 0.0919Q_dv + 0.0399v^2\ \dot{T}_7 = -0.1058Q_d - 0.0337Q_d^2 + 0.0589Q_dv$$

$$\dot{c}_9 = 0.4089 - 0.3757Q_d - 0.1199v + 0.0688Q_d^2 + 0.081Q_dv,$$

$$\dot{T}_9 = 0.5461 - 0.3913Q_d - 0.3457v + 0.149Q_dv + 0.054v^2$$

$$\dot{c}_{10} = -0.1183 + 0.0494Q_d + 0.1464v - 0.0366Q_dv - 0.0325v^2,\ \dot{T}_{10} = -0.0697Q_d + 0.0290Q_dv$$

Again, these equations discovered by SINDy render an expression with respect to the change in current expressed as a change in discharge capacity, voltage, and temperature. A second governing equation was yielded as either a change in the discharge capacity Q_d or temperature T. There are distinct differences between clusters.

4. FUTURE WORK

This paper has investigated the effect of charging conditions on the initial state and life cycle of LIBs. By clustering features from the battery current profile, we identified 10 core clusters of policy types. Selecting a representative for each cluster, we modeled multivariate sensor profiles using differential equations, discovered by the SINDy method. The preliminary results show potential statistically significant differences between charging conditions. With that said, the sparse equations yielded can be made more accurate. Additional tuning of the hyper-parameter λ, which can be done in a myriad of ways,[11] may afford a more accurate model. Moreover, it may be necessary to consider additional functions within the candidate library, such as trigonometric or nonlinear functions. Another possibility for converging to a more parsimonious solution may be to change convergence algorithms for the sparse regression, such as to LASSO.[10] Once an adequate expression has been discovered, additional methods can be utilized to compare results, such as neural ODE or PINNs. discovering a representative equation of the dynamics.

Though there are extreme resource costs for LIB recycling, the benefits will greatly outweigh them over time, due to the improved environmental quality, the resulting economic prosperity, and the continual striving for social equity. LIB manufacturing requires the mining of metals like cobalt and nickel, which requires scare resources and presents environmental concerns. A barrier to the economic viability of LIB recycling is the uncertainty surrounding the lifetime of LIBs; recycling an LIB too early results in a loss of useful battery life whereas a completely failed LIB may not be salvageable. Thus, an accurate and parsimonious characterization of the state-of-health of an LIB is paramount for making decisions regarding LIB recyclability.

ACKNOWLEDGMENTS

Many thanks are extended to the UTRGV Center for Advanced Manufacturing Innovation and Cyber Systems (CAMICS) and the Consortium of Advanced Additive Manufacturing Research and Education for Energy Related Systems (CA2REERs) for sponsoring this work. The domain-level expertise, provided by Dr. Jianzhi (James) Li and Dr. Benjamin Peters has been critical in shaping the direction of data analysis. Additional appreciation is expressed towards Dr. Tamer Oraby from the UTRGV School of Mathematical and Statistical Sciences (SMSS) who has demonstrated excellence in creating a classroom environment that promotes applied research with modern methods on the boundaries of new knowledge.

REFERENCES

[1] "Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2021." U.S. EPA, 2024. https://www.epa.gov/ghgemissions/inventory-us-greenhouse-gas-emissions-and-sinks. (Accessed: 3 April 2024).

[2] Neumann, J., Petranikova, M., Meeus, M., Gamarra, J. D., Younesi, R., Winter, M., and Nowak, S., "Recycling of lithium-ion batteries—current state of the art, circular economy, and next generation recycling," *Advanced Energy Materials* **12**(17), 2102917 (2022).

[3] Gaines, L., Zhang, J., He, X., Bouchard, J., and Melin, H. E., "Tracking flows of end-of-life battery materials and manufacturing scrap," *Batteries* **9**(7) (2023).

[4] "Used Lithium-Ion Batteries." U.S. EPA, 2024. https://www.epa.gov/recycle/used-lithium-ion-batteries. (Accessed: 3 April 2024).

[5] Gaines, L., Richa, K., and Spangenberger, J., "Key issues for li-ion battery recycling," *MRS Energy & Sustainability* **5**, E14 (2018).

[6] Brunton, S. L., Proctor, J. L., and Kutz, J. N., "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences* **113**(15), 3932–3937 (2016).

[7] Severson, K. A., Attia, P. M., Jin, N., Perkins, N., Jiang, B., Yang, Z., Chen, M. H., Aykol, M., Herring, P. K., Fraggedakis, D., et al., "Data-driven prediction of battery cycle life before capacity degradation," *Nature Energy* **4**(5), 383–391 (2019).

[8] Brunton, S. L. and Kutz, J. N., [*Data-driven science and engineering*], Cambridge University Press, Cambridge, England (Feb. 2019).

[9] Schmidt, M. and Lipson, H., "Distilling free-form natural laws from experimental data," *science* **324**(5923), 81–85 (2009).

[10] Tibshirani, R., "Regression Shrinkage and Selection via The Lasso: A Retrospective," *Journal of the Royal Statistical Society Series B: Statistical Methodology* **73**, 273–282 (04 2011).

[11] Yang, L. and Shami, A., "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing* **415**, 295–316 (2020).

[12] Boisbunon, A., Canu, S., Fourdrinier, D., Strawderman, W., and Wells, M. T., "Aic, cp and estimators of loss for elliptically symmetric distributions," (2014).

[13] Champion, K., Lusch, B., Kutz, J. N., and Brunton, S. L., "Data-driven discovery of coordinates and governing equations," *Proceedings of the National Academy of Sciences* **116**(45), 22445–22451 (2019).

[14] Bertsimas, D. and Gurnee, W., "Learning sparse nonlinear dynamics via mixed-integer optimization," (2022).

[15] Fasel, U., Kaiser, E., Kutz, J. N., Brunton, B. W., and Brunton, S. L., "Sindy with control: A tutorial," in [*2021 60th IEEE Conference on Decision and Control (CDC)*], 16–21, IEEE (2021).

[16] Kaheman, K., Kutz, J. N., and Brunton, S. L., "Sindy-pi: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics," *Proceedings of the Royal Society A* **476**(2242), 20200279 (2020).

[17] Naozuka, G. T., Rocha, H. L., Silva, R. S., and Almeida, R. C., "Sindy-sa framework: enhancing nonlinear system identification with sensitivity analysis," *Nonlinear Dynamics* **110**(3), 2589–2609 (2022).

[18] Fasel, U., Kutz, J. N., Brunton, B. W., and Brunton, S. L., "Ensemble-sindy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control," *Proceedings of the Royal Society A* **478**(2260), 20210904 (2022).

[19] Hirsh, S. M., Barajas-Solano, D. A., and Kutz, J. N., "Sparsifying priors for bayesian uncertainty quantification in model discovery," *Royal Society Open Science* **9**(2), 211823 (2022).

[20] Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K., "Neural ordinary differential equations," *Advances in neural information processing systems* **31** (2018).

[21] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics* **378**, 686–707 (2019).

[22] Aydemir, G. and Paynabar, K., "Image-based prognostics using deep learning approach," *IEEE Transactions on Industrial Informatics* **16**(9), 5956–5964 (2019).

[23] Liu, X., Yeo, K., and Kalagnanam, J., "Statistical modeling for spatio-temporal degradation data," *arXiv preprint arXiv:1609.07217* (2016).

[24] Gebraeel, N. Z., Lawley, M. A., Li, R., and Ryan, J. K., "Residual-life distributions from component degradation signals: A bayesian approach," *IiE Transactions* **37**(6), 543–557 (2005).

[25] Abdi, H. and Williams, L. J., "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics* **2**(4), 433–459 (2010).

[26] Thorndike, R. L., "Who belongs in the family?," *Psychometrika* **18**(4), 267–276 (1953).

[27] Tibshirani, R., Walther, G., and Hastie, T., "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63**(2), 411–423 (2001).

[28] Millikan, R. A. and Bishop, E. S., [*Elements of electricity: a practical discussion of the fundamental laws and phenomena of electricity and their practical applications in the business and industrial world*], American Technical Society (1917).