

Estimation of State-of-Charge and Capacity of Used Lithium-Ion Cells

Nenad G. Nenadic¹, Howard E. Bussey², Paul A. Ardis³, Michael G. Thurston¹

¹ *Golisano Institute of Sustainability, Rochester Institute of Technology, Rochester, NY 14623 USA*

nxnasp@rit.edu mgtasp@rit.edu

² *Tixlers Letters, PO BOX 831, Pittsford NY 14534*

howard@tixlers.com

³ *GE Global Research, Research Circle Bldg. K1-4A6, Niskayuna, NY 12309*

ardis.p@ge.com

ABSTRACT

We describe an approach to estimate state-of-charge and faded capacity of cobalt-based lithium-ion cell based on time-domain analysis of a short-term transient. This approach requires a relatively short-duration test and is suitable for repurposing cells for less demanding applications. The successful estimation requires previous characterization of the cells for the given family because lithium ion chemistries differ significantly. Two algorithms were considered for estimation of unknown state-of-charge and capacity: Bayesian inference and boosted regression trees. The achieved accuracy was 95 % of capacity estimations; estimations were within ± 2 % of the nominal cell capacity from the true value.

1. INTRODUCTION

Lithium-ion (Li-ion) batteries in high-end applications, such as communications systems, electric vehicles, and plug-in hybrid electric vehicles, are typically replaced at an early point in their lives, while they still have considerable capacity left. Typically this percentage is ~ 80 % of new because, by design, in the demanding applications, the remaining capacity determines an important system parameter, with low degradation tolerance. For example, the remaining capacity of electric vehicles translates directly into range. At the end of useful life in the demanding primary applications, the batteries can continue to be used for secondary, less demanding applications, such as storage in microgrids, power tools, and consumer electronics, which all exhibit considerably higher tolerance to the loss of capacity.

In practice many cells are discarded even before they reach ~ 80 %. The capacity of these cells can be higher than 90 %

because of general reluctance of the battery integrators to manage mismatched cells. If something, other than a cell, fails in the battery system, the cells and modules have a high probability of being discarded. Generally the maintenance of large battery packs faces considerable apprehension related to mixing cells of mismatched capacities, resulting in early retirement of the cells. In order to make sustainable decisions about pack maintenance, methods are needed to understand the relative health of each cell.

In both applications (reuse and maintenance), it is crucial to be able to efficiently assess Li-ion *state-of-charge* SOC and remaining capacity Q_c . While the most reliable estimation is obtained by cycling a cell from its fully charged state to its fully discharged state, this process is expensive and time-consuming.

Cell capacity may instead be estimated using data-driven techniques, where measurements of a cell's characteristics are used to estimate the cell capacity, based on a model. We are interested in data-driven approaches rather than electro-chemical models, as data driven modeling techniques can be applied to a wide variety of cell chemistries with little expert intervention. In contrast, electro-chemical models require considerable engineering, chemistry, and physics expertise to create a model for each cell chemistry. One of the commonly used models for data-driven regression is the *boosted regression tree* (BRT), because it has generally good predictive performance for non-linear functions and it is tolerant of imperfect data (Elith, Leathwick, & Hastie, 2008). We will explore BRT-based estimation for this problem, demonstrating its performance alongside another, more traditional data-driven approach: Bayesian inference.

Li-ion cells come from different families, determined by the cathode chemistries, the chief among these are: lithium cobalt (LiCoO_2), lithium iron-phosphate (LiFePO_4), lithium manganese spinel (LiMn_2O_4), and lithium mixed-metal (LiMO_2)

Nenad G. Nenadic et.al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

cathodes. This article is concerned with the lithium cobalt family. Figure 1 shows natural capacity degradation of eight lithium cobalt cells, labeled S1 through S8, subjected to full cycle charges and discharges vs. the number of cycles N . The circles indicate the in-class spread of the data in Q_c - N space. The spread slightly increases as the cells age.

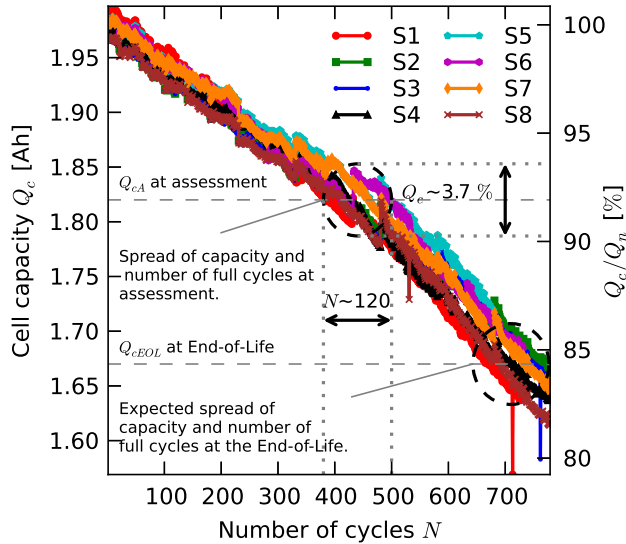


Figure 1. Capacity fade of eight lithium cobalt cells.

The data shown in Figure 1 are *full cycles*, the cells of a specific chemistry are fully charged and discharged at room temperature. This usage pattern does not immediately generalize to other cycling profiles, operating conditions, and environmental conditions because cell aging and capacity fade of the cells generally depends on several other important parameters (Broussely et al., 2005; Spotnitz, 2003; Vetter et al., 2005), including temperature, where higher temperature accelerates aging; charge and discharge current, where higher currents accelerate aging; depth of discharge, where shallower cycles decelerate aging; and rest time between charge and discharge, where low rests after discharge can greatly accelerate capacity loss. While the process does not directly apply to an arbitrary scenario, the conditions of the batteries in high end applications are generally carefully monitored and maintained. Thus, the entire process described here, including capturing training data, can be applied in full in a more realistic setting, after modification to accommodate the operating conditions are put in place.

High-end battery systems employ a *battery monitoring system* (BMS), with a sophisticated cell equalization, and a real-time *prognostics health monitoring* (PHM) system for estimation of *SOC* and *state-of-health* (SOH). There has been considerable work in the development of battery PHM, whose

full review is beyond the scope of this study (for early works see (Plett, 2004; Goebel, Saha, Saxena, Celaya, & Christophersen, 2008; Saha, Goebel, Poll, & Christophersen, 2009; Saha & Goebel, 2009); for a review, see (Zhang & Lee, 2011), and for more recent works (Liu, Pang, Zhou, Peng, & Pecht, 2013; Sankararaman, Daigle, Saxena, & Goebel, 2013)). Our approach complements real-time PHM with a specific use at the end of life. PHM provides the usage history of the pack and the state-of-health estimates as an essential input for the decision process at the end of life. However, PHM systems are concerned with the battery system as a whole and they often do not monitor individual cells. Instead, they typically use extrapolations from a finite number of measurements. Knowing attributes of individual cells is valuable when they are considered for secondary applications because the properties of individual cells diverge in time and use and this study aims of individual cell assessment at the end of life of their primary application.

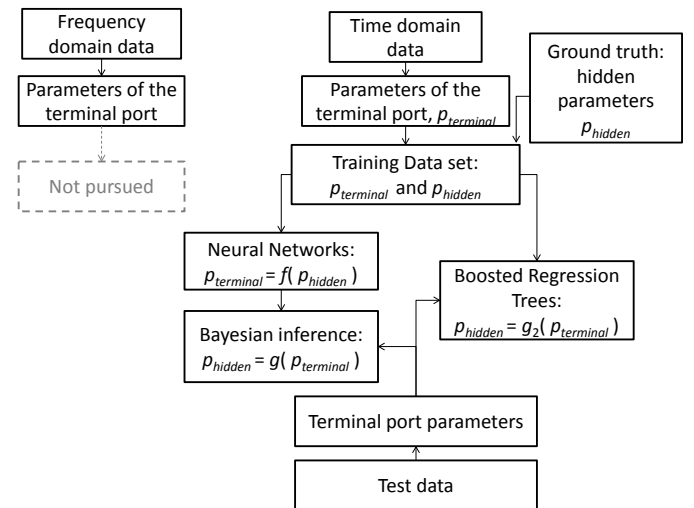


Figure 2. Outline of the approach.

2. APPROACH

Because direct capacity measurements are expensive, the age of cells is often estimated indirectly. In particular, the two methods considered here are based upon extensions to the well-known correlation between age and resistance. To estimate age and *SOC*, a characterization process first extracts a set of its parameters over a range of known *SOC* and capacity Q_c , using a battery model as described in Section 2.1, thereby estimating cell age. The data obtained during this characterization process is then used to estimate unknown values for capacity and *SOC*.

The outline of the approach is depicted in Figure 2. After introducing the battery model, with its two ports, terminal and hidden, we proceed to identify the components of the terminal port from the time-domain and frequency data. Because the

time domain approach yields a more accurate model, within our experimental and analytical setting, parameters extracted from the time-domain analysis are used for subsequent estimations of the hidden parameters.

Two approaches for estimation of are considered: Bayesian inference and BRTs. In the first approach, the direct relationships of the terminal ports are first learned as a function of the hidden parameters, and then inverted using the Bayesian inference. In the second approach, the estimations of hidden parameters are learned directly from the data.

2.1. Battery model

A popular battery model adopted in time-domain analyses, described in (Chen & Rincon-Mora, 2006), consists of two electrical ports as shown in Figure 3. The parameters of the terminal port are: open-circuit voltage V_{oc} , resistance R_0 , and two parallel RC circuits R_1-C_1 and R_2-C_2 . We refer to this model as *the double exponential model*¹. In this model, the circuit elements are both functions of SOC and consumed life. For example, if the consumed life is expressed in terms of the number of full charge/discharge cycles N , and expressed in remaining capacity Q_c (see Figure 1) a circuit element is the function of SOC and Q_c , e.g. $R_0 = R_0(SOC, Q_c)$.

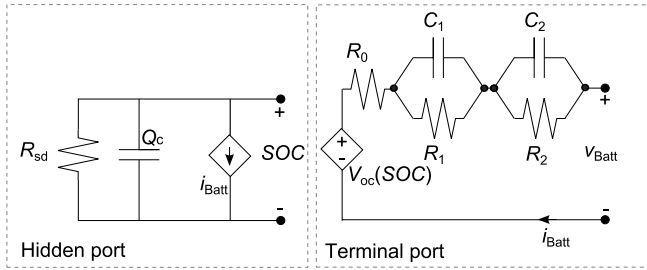


Figure 3. Double-exponential model.

The hidden port of the model consists of cell capacity Q_c , represented by a capacitor, self-discharge resistor R_{sd} , and controlled-current source SOC . The loss of charge when the battery is in open circuit condition is typically negligible for most commercial lithium ion batteries, and R_{sd} can be safely ignored; that is, we assume that $R_{sd} \rightarrow \infty$. Because cell capacity Q_c fades as the cell ages, it can be used as a direct measure of consumed life. As a result, assessment consists of two steps: first, the terminal voltage v_{Batt} and terminal current i_{Batt} are used to estimate the circuit components of the terminal port of Figure 3: V_{oc} , R_0 , R_1 , C_1 , R_2 , and C_2 ; and second, the parameters of the terminal port are used to estimate the components of the hidden port, SOC and Q_c .

¹A simpler model, with only one parallel RC circuits ($R_2 = 0$ and $C_2 \rightarrow \infty$) is referred to as *the single-exponential model*. The simple model was also considered, but was outperformed by the double-exponential not only in fit accuracy, but also in generalization.

2.2. Parameter extraction

The model parameters can be extracted from measurements in the time or frequency domains. In our work, reduced-error fits were achieved in the time domain, although parameter extraction from impedance spectroscopy (frequency domain) is also discussed.

2.2.1. Time domain

Time-domain analysis is based upon a short-duration step change in current during charge and discharge; a single-step test takes a few minutes (compared to several hours for full charge). Because the trajectories during charge and discharge are slightly different, including both charge and discharge adds to the variance of the data. Here we present just the results associated with charge cycles. The model parameters are extracted from the transient waveform of the terminal battery voltage v_{Batt} and current i_{Batt} .

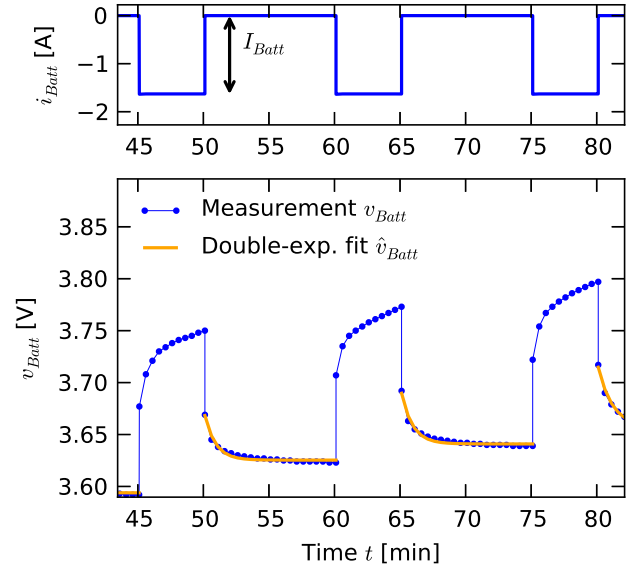


Figure 4. A charge transient: measurements and the fit.

Figure 4 illustrates example parametric fits for the double-exponential model during the characterization phase. The battery was charged with pulsed current: active charging lasted five minutes, followed by ten minutes of rest. There were 18-20 such pulses during a charge, initially, and the number of pulses dropped as the cell aged. The current was held constant during the active period. The data waveforms were sampled every thirty seconds, $T_s = 30$ s. Note that the current is negative during charge, according to the convention that i_{Batt} is positive during discharge (see Figure 3). To determine parametrization, we consider transients when the battery current charge drops to zero, after being constant sufficiently long that the transient had died out, and choose the

beginning of the transient as the time reference, i.e., $t = 0$. The terminal battery voltage is then given by

$$v_{Batt}(t) = v_{Batt}(0^-) - [R_0 + R_1 + R_2 - R_1 e^{-t/(R_1 C_1)} - R_2 e^{-t/(R_2 C_2)}] I_{Batt} \quad (1)$$

where I_{Batt} is the magnitude of the charge current before the transient started. Eq. (1) was used to estimate the terminal voltage. The unknown parameters are fitted using a non-linear optimization function such as *fminsearch()* in Matlab or *fmin()*, used here, from the *scipy* Python library (Jones, Oliphant, Peterson, et al., 2001–). The cost function J covers the squared error of the measured voltage v_{Batt} and the estimate \hat{v}_{Batt} , summed over the instances of time, with an additional term controlled by a weight that penalizes negative parameter values:

$$J(\mathbf{y}|v_{Batt}) = \sum_{t_k} (v_{Batt}(t_k) - \hat{v}_{Batt}(t_k|\mathbf{y}))^2 + \lambda \sum_{y_i < 0} |y_i| \quad (2)$$

In this equation, \mathbf{y} is the vector of the unknown parameters $\mathbf{y} = [V_{oc}, R_0, R_1, C_1, R_2, C_2]^T$, y_i are its components, λ is the weight for penalizing negative values, and $\hat{v}_{Batt}(\mathbf{y})$ is the estimate of the terminal battery voltage for the given test set of the terminal circuit parameters. Finally, the terminal port parameters of Figure 3 can be estimated simultaneously using an optimization library

$$\hat{\mathbf{y}} = \underset{\mathbf{y}}{\operatorname{argmin}} [J(\mathbf{y}|v_{Batt})]. \quad (3)$$

Slightly different approach is to first estimate \hat{R}_0 , and then separately estimate \hat{V}_{oc} , \hat{R}_1 , \hat{C}_1 , \hat{R}_2 and \hat{C}_2 using the estimated \hat{R}_0 . A step charge (or discharge) follows a period where $i_{Batt} = 0$ for a long enough time that C_1 and C_2 have effectively discharged, so Δv_{Batt} and Δi_{Batt} , from the first two points of the step current change, can be used to estimate R_0 :

$$\hat{R}_0 = \frac{\Delta v_{Batt}}{\Delta i_{Batt}} = \frac{v_{Batt}(0+) - v_{Batt}(0-)}{I_{Batt}} \quad (4)$$

Comparison of the two approaches showed that first estimating R_0 and then estimating the other parameters yielded smaller mean-squared estimation error, so the two-phase parameter estimation method was used. Note that Eqs. (1-2) remain the same, but \mathbf{y} in Eq. (2) does not contain R_0 , that is the parameter vector here is $\mathbf{y} = [V_{oc}, R_1, C_1, R_2, C_2]^T$.

2.2.2. Frequency domain

A common health indicator of the battery aging is impedance (Hawkins, 1994; Lasia, 1999; Amine et al., 2001; Brousely et al., 2005; Dees, Gunen, Abraham, Jansen, & Prakash, 2005; Tröltzsch, Kanoun, & Tränkler, 2006). According to the double-exponential model of Figure 3, the impedance of

the terminal port is given by

$$\hat{\underline{Z}}_{Batt}(\omega) = \hat{R}_0 + \frac{\hat{R}_1}{1 + j\omega \hat{R}_1 \hat{C}_1} + \frac{\hat{R}_2}{1 + j\omega \hat{R}_2 \hat{C}_2} \quad (5)$$

where ω is the angular frequency; $\omega = 2\pi f$.

Figure 5(a) shows the average impedance spectra of fully charged cells when new (with at most a few cycles passed) and after their capacity dropped to 90 % of new. Each plot contains seventy one data points in 0.01 Hz – 100 kHz range. The points are logarithmically spaced over the frequency range.

The double-exponential model of Figure 3 and Eq. (5) do not account for the inductive part of the curve, but the impedance spectra becomes inductive only for frequencies $f \geq 1$ kHz, which is of lesser importance than $f < 1$ kHz because batteries normally operate at low frequency. The standard aging feature employed in the industry is the real part of the impedance evaluated at $f = 1$ kHz, which closely approximates R_0 of Figure 3, because of high correlation between resistance and capacity was expected (see e.g. (Saha & Goebel, 2008)).

The impedance data shows that more resolution is available at lower frequencies (1 Hz or less), in the so-called Warburg region. At lower frequencies, the change in the real part of the impedance is $\Delta \Re\{\underline{Z}_{Batt}\} \approx \Delta(R_0 + R_1 + R_2)$, whereas at $f = 1$ kHz, $\Delta \Re\{\underline{Z}_{Batt}\} \approx \Delta R_0$.

The model parameters are grouped in parameter vector \mathbf{z} , which does not contain the open circuit voltage V_{oc}

$$\mathbf{z} = [R_0, R_1, C_1, R_2, C_2]^T \quad (6)$$

Fitting the model in the frequency domain can be done by minimizing the cost function J_Z , defined as the square error in the impedance (the difference between the measured impedance \underline{Z}_{Batt} and the estimate $\hat{\underline{Z}}_{Batt}$ given by Eq. (5)), summed over the frequencies and with an additional term for penalizing negative values of the parameters, with respect to the model parameter vector.

$$\begin{aligned} \hat{\mathbf{z}} &= \underset{\mathbf{z}}{\operatorname{argmin}} \left[J_Z \left(\mathbf{z} \mid \Re\{\underline{Z}_{Batt}(\omega)\}, \Im\{\underline{Z}_{Batt}(\omega)\} \right) \right] \\ &= \underset{\mathbf{z}}{\operatorname{argmin}} \left\{ \sum_{\omega_k} \left[\left(\Re\{\underline{Z}_{Batt}(\omega_k)\} - \Re\{\hat{\underline{Z}}_{Batt}(\omega_k)\} \right)^2 \right. \right. \\ &\quad \left. \left. + \left(\Im\{\underline{Z}_{Batt}(\omega_k)\} - \Im\{\hat{\underline{Z}}_{Batt}(\omega_k)\} \right)^2 \right] \right. \\ &\quad \left. + \lambda \sum_{\hat{z}_i < 0} |\hat{z}_i| \right\} \quad (7) \end{aligned}$$

While the extracted values of the model parameters are similar whether transient domain or frequency domain data is used, the double-exponential model fits the transient data

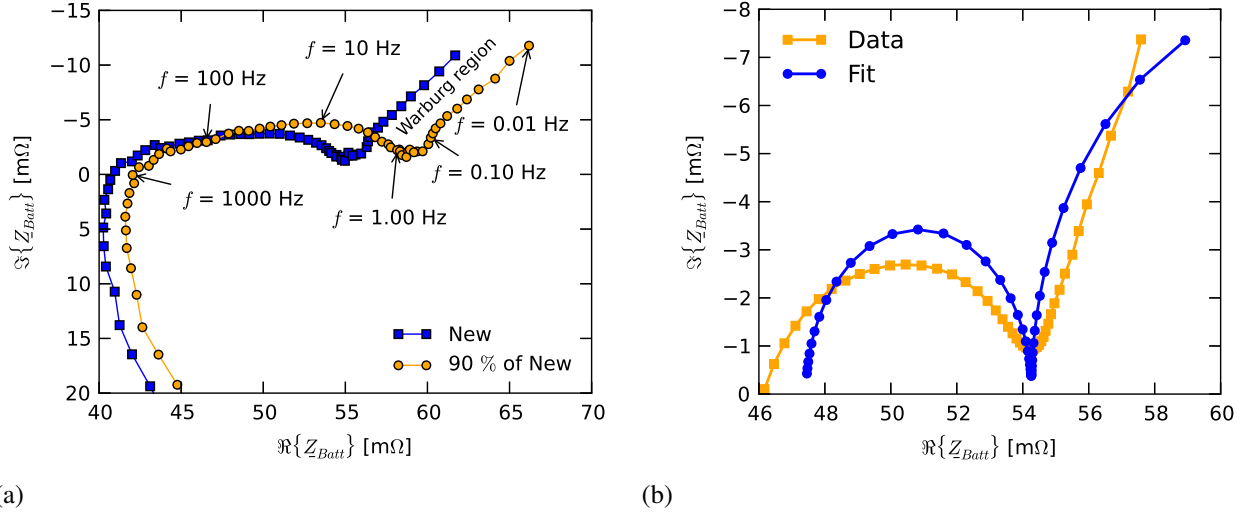


Figure 5. Impedance spectra. (a) Charged and discharge with frequency annotations. (b) Least square fit.

better than the impedance spectroscopy. However, the impedance spectroscopy does not include V_{oc} , which is probably the best single indicator of SOC . From a practical viewpoint, collecting the transient data is considerably cheaper and easier than collecting the spectroscopy data because the step charges and discharges are performed on the same piece of equipment (single cell battery tester, e.g. Maccor 4600) as any other charge/discharge profile. The cell aging and the collection of the transient data can be performed on many cells in parallel with minimum engagement of an operator. Moreover, the transient data collection is considerably less expensive because the cost of a channel on a battery tester ($\sim \$1.3k$)² is considerably less than the cost of an impedance analyzer and potentiostat ($\sim \$55k$). The equipment cost and its scale-up and engagement of an operator are of great practical importance, if the approach is to be considered for a real, industrial setting. The rest of the document is concerned only with the information extracted from the transient data.

3. STATE-OF-CHARGE AND CAPACITY ESTIMATION

Two methods of SOC and capacity estimation were employed: Bayesian inference and BRTs. These methods are described in turn in the following subsections.

The data set was based on eight cells, with up to about twenty transients associated with each charge and discharge. For each transient the parameters of the terminal port were extracted. The entire dataset had 2154 values for each parameter. We used six cells for training and two cells for test. In the first approach, we trained the neural networks with six fixed cells ($S1$ - $S6$), and verified results using the remaining

two cells. In the second approach, model parameter values for all 28 combinations of eight cells taken six (for training) at a time, using the remaining two cells for the verification.

3.1. Bayesian inference

The Bayesian approach consists of two steps: *family characterization* and *posterior estimation*. In the family characterization step, we use a held-out training set to compute functions expressing the parameters of the terminal port in terms of the parameters of the known hidden port: $V_{oc}(SOC, Q_c)$, $R_0(SOC, Q_c)$, $R_1(SOC, Q_c)$, $C_1(SOC, Q_c)$, $R_2(SOC, Q_c)$, and $C_2(SOC, Q_c)$. In the posterior estimation step, we use the computed functions to estimate SOC and Q_c at fixed points by way of Bayesian inference.

3.1.1. Family characterization

In the characterization process, parameter fitting is repeated at different levels of state-of-charge and at different ages (number of cycles) for a set of cells. This characterization results in empirical surfaces of the model parameters as a function of state-of-charge and age. At first, a two-layer neural network was employed for modeling the functions³, with forty neurons in the hidden layer, given by

$$\hat{y}_k(SOC, Q_c; \mathbf{w}) = \sigma \left(\sum_{j=1}^{40} w_{kj} \sigma(w_{j1}SOC + w_{j2}Q_c + w_{j0}) + w_{k0} \right), \quad (8)$$

²This price is computed for a 32-channel unit and it would be less if a large system was employed.

³This article employs the notation where the number of layers of a neural network is equal to the number of adaptive weights, as in (Bishop, 1st ed. 2006. Corr. 2nd printing).

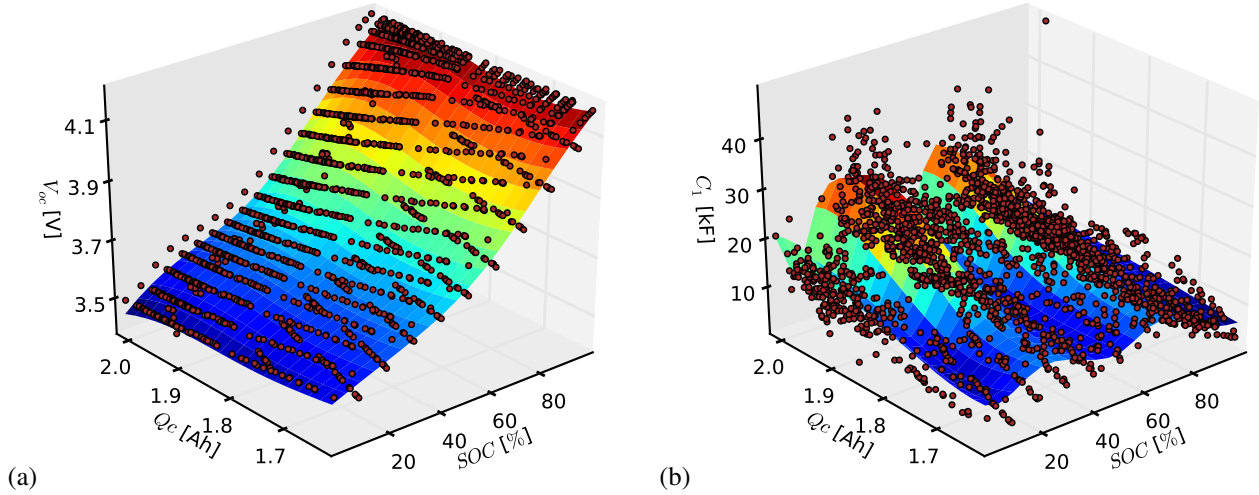


Figure 6. Model parameters and neural network surface in $SOC-Q_c$ plane. (a) Open-circuit voltage V_{oc} . (b) Capacitance C_1 .

where $\sigma(\cdot)$ is the logistic sigmoid, \hat{y}_k stands for one of the parameters in the terminal port (V_{oc} , R_0 , R_1 , C_1 , R_2 , and C_2), and \mathbf{w} are the weights. This standard neural network topology, known as the universal function approximator, with its expressive power, and its relation to Kolmogorov theorem is discussed in (Duda, Hart, & Stork, 2000, Section 6.2.2) and (Bishop, 1995, Section 4.7). However, in our case, significantly better performance was achieved after the two-layer topology 2-40-1, was replaced by a three-layer 2-20-20-1 topology with the same total number of neurons. Both networks were trained over 1,200 epochs. Our implementation made use of the *PyBrain* Python library for neural networks (Schaul et al., 2010).

A neural network takes considerably longer to train than a regularized 2D polynomial, or 2D Fourier function approximation (which due to their simplicity were considered first), but the model fit (measured by the total squared error) was found to be considerably better for the neural network.

The performance of the three-layer (2-20-20-1) network is shown for two out of six parameters. Figure 6a shows that the open-circuit voltage V_{oc} is a smooth, monotonic surface in $SOC-Q_c$ space; C_1 's dependence on SOC , on the other hand, is non-monotonic, as shown in Figure 6b. The monotonicity of the remaining four parameters is between V_{oc} (the flattest) and C_1 (the most wavy).

An advantage of learning the inverse relationships, in this case, is that each parameter is learned in two-dimensional space (because the hidden port has two parameters) and can be readily visually inspected and verified that the fits are reasonable and that over-fitting does not occur. When over-fitting was observed (in case of non-regularized least squares for polynomial and Fourier fits), it manifested itself as a large function variation over a small domain area. Thus, a quanti-

tative detection criterion can be formulated as the maximum of the total approximated surface areas above a sub-domain, over the grid of sub-domains. For the surfaces approximated with the neural network model, such over-fitting did not occur. Moreover, because significant number of data points do not lie on the fitted surfaces, we can readily observe that the data over-determines the model and that splitting the dataset into test and verification to avoid over-fitting is not necessary. On the other hand, this may suggest that the bias of the model may be too strong and that adding more variance (using a more complex network) could improve the accuracy.

3.1.2. Posterior estimation

Once parameter's empirical relationships are available, they can be used for Bayesian inference. This approach essentially inverts the relationships $V_{oc}(SOC, Q_c)$, $R_0(SOC, Q_c)$, etc. to estimate SOC and Q_c . Inference here starts with a uniform prior over values for SOC and Q_c , but in practice one can easily embed knowledge about the conditions of the cells into the prior. For example, the probabilities of capacity that exceed a prescribed percent limit (e.g. 90 %) can be set to zero, if there is a strong reason to believe that the population of cells that need the inspection must have less than some specified percentage of the nominal capacity left.

Individual estimates of the parameters of the terminal port are presented to the Bayesian inference engine one at a time, with the posterior distribution narrowing after each new piece of evidence. This is because the probability density of the posterior at the k^{th} step $p_{post,k}$ is proportional to the probability density of the prior $p_{prior,k}$ and probability density of the likelihood estimate $\hat{p}_{lik,k}$:

$$p_{post,k} \propto p_{prior,k} \hat{p}_{lik,k}, \quad (9)$$

where the prior at the k^{th} step is the posterior of the $(k-1)^{\text{th}}$

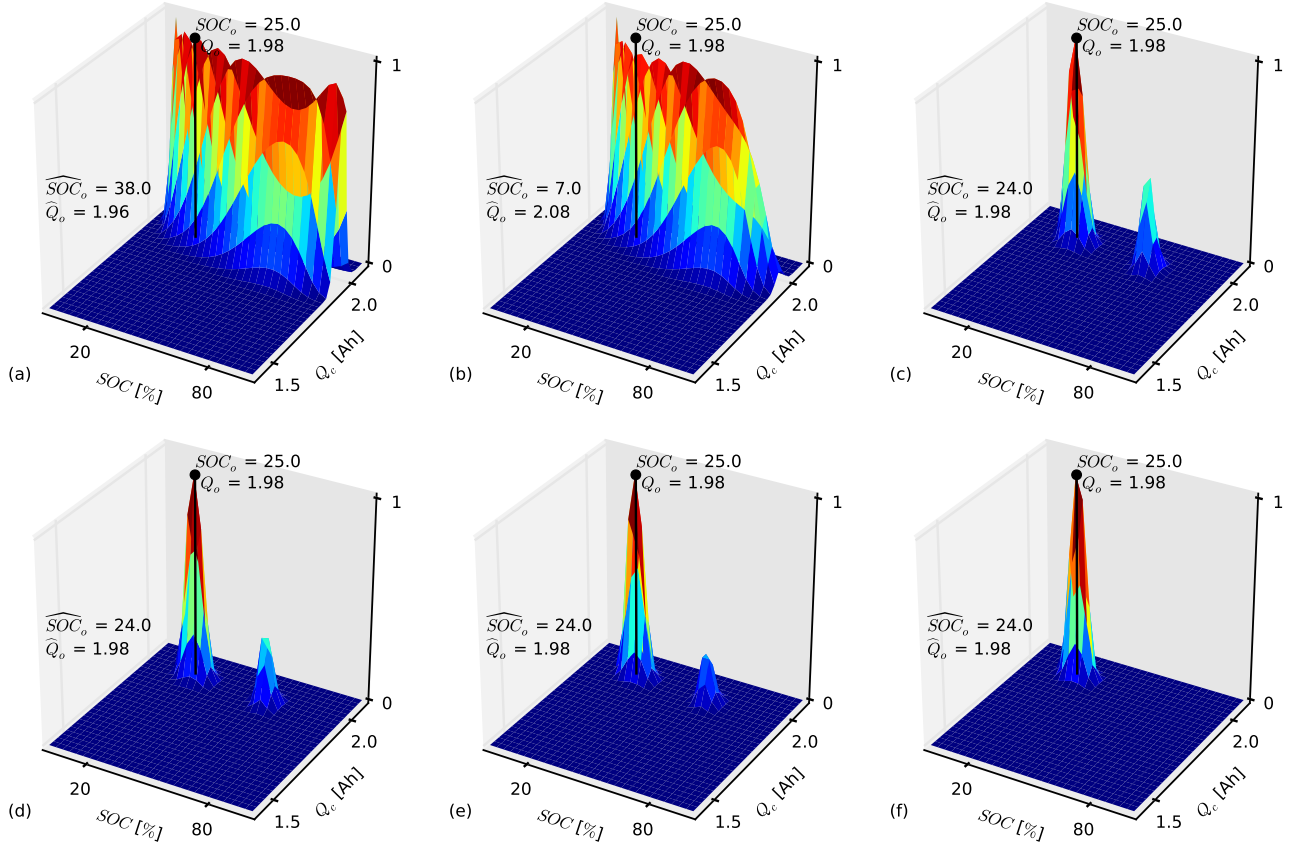


Figure 7. An example of a posterior after Bayesian bivariate estimation. The posterior distribution becomes narrower after each new piece of evidence is presented (a) After \hat{R}_0 . (b) After \hat{R}_0 and \hat{R}_1 . (c) After \hat{R}_0 , \hat{R}_1 , and C_1 . (d) After \hat{R}_0 , \hat{R}_1 , C_1 , and R_2 . (e) After \hat{R}_0 , \hat{R}_1 , C_1 , R_2 , and C_2 . (f) After \hat{R}_0 , \hat{R}_1 , C_1 , R_2 , C_2 , and V_{oc} .

step

$$p_{prior,k} \leftarrow p_{post,k-1}, \quad (10)$$

The initial prior $p_{prior,0}$ is assumed to be user-selected (uniform in our case). Assuming that the estimates of parameters are normally distributed with respect to their approximated surface in $N - SOC$ space, the probability density of the likelihood estimate can be approximated as

$$\hat{p}_{lik,k}(SOC, Q_c) \propto \exp \left[-\frac{(\hat{y}_k - f_{NN,k}(SOC, Q_c))^2}{2\sigma_{NN,k}^2} \right], \quad (11)$$

where the normalization constant is omitted for clarity.

In Eq. (11), \hat{y}_k , $f_{NN,k}()$, and $\sigma_{NN,k}$ signify, respectively, one of the estimated parameters of the terminal port ($\hat{R}_0, \hat{R}_1, \hat{C}_1, \hat{R}_2, \hat{C}_2$, or \hat{V}_{oc}) from a single transient via minimization of J given by Eq. (2), the neural network mapping of the estimated parameter, and the standard deviation of the neural network model for the given parameter.

In Eqs. (9-10) the dependence of the probability density functions on state-of-charge and capacity is not made explicit for readability, but clearly *all* probability density functions are given with respect to $SOC-Q_c$. In practical calculations, evaluations of these density functions are carried out on a discrete mesh, with mesh density affecting result precision.

Figure 7 shows an example of the posterior bivariate distributions of SOC and Q_c after each new piece of evidence is presented. The posterior distributions are shown as surfaces and the actual values of SOC and Q_c are shown as a stem. Maximum a posteriori estimates, \hat{SOC} and \hat{Q}_c are also indicated as plot annotations. As noted, the posterior becomes narrower with each piece of evidence, with the maximum a posteriori estimate approaching the actual values. A 30×30 mesh was used for this evaluation. Consider first Figure 7a, which shows the posterior computed using exp Eq. (9), given the uniform prior and the likelihood computed using Eq. (11), where R_0 was extracted from a waveform selected from test and verification data set of measurements and $f_{NN,0}$ is a neural network mapping of R_0 as a function of SOC and Q_c , $\hat{R}_0 = f_{NN,0}(SOC, Q_c)$. This distribution was then used as

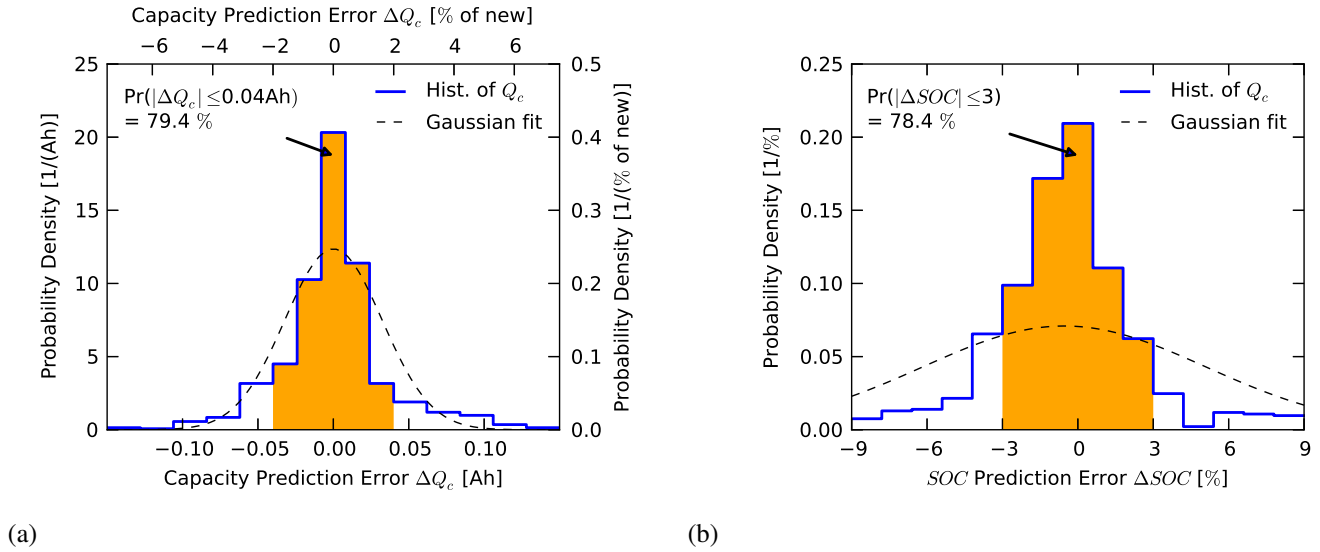


Figure 8. (a) Histogram of estimation errors of the capacity Q_c . (b) Histogram of estimation errors of SOC .

the prior for the next piece of evidence computed from the test dataset, R_1 and the likelihood is computed using Eq. (11), where we now have $\hat{R}_1 = f_{NN,1}(SOC, Q_c)$. The new prior and the likelihood are combined again using Eq. (9) and the result is shown in Figure 7b. Figure 7c through Figure 7f were generated using the same process: in each step the previous posterior becomes a prior and the likelihood given by Eq. (11), where neural network mappings $f_{NN,2}$ – $f_{NN,5}$, are subtracted from C_1 , R_2 , C_2 , and V_{oc} , respectively.

The histogram in Figure 8a summarizes the capacity estimation performance of the approach over the test data. The errors were computed by subtracting the final estimates (such as those shown in Figure 7f) from the ground truth. As discussed at the beginning of Section 3, data from two cells were used for validation. In this figure, the bottom x -axis corresponds to capacity error in amp-hours and the top axis corresponds to the percent error relative to a nominal, new cell. Similarly, the left y -axis corresponds to the bottom x -axis with units in $1/(Ah)$, while the right y -axis corresponds to the top x -axis, with units in $1/(\% \text{ of new})$. Finally, the histogram in Figure 8b summarizes the SOC estimation performance of the approach over the test data.

3.2. Boosted Regression Trees

BRT combine the simplicity and data-handling flexibility of decision trees (Breiman, Friedman, Stone, & Olshen, 1984) with improved performance provided with boosting. Because they are often outperformed by competing models, decision trees are now mostly used in ensembles, where multiple individual models are suitably combined to arrive at a model of a considerably better performance.

There are many ensemble models, but they are based on the three main methods: boosting, bagging, and stacking. In boosting (Freund & Schapire, 1995), the training examples are weighted and the weights are suitably adjusted in each iteration based on the errors in the previous iteration. In bagging (Breiman, 1996), several models are trained from the dataset using resampling and the results are combined by voting. In stacking (Wolpert, 1992), the learners are hierarchically arranged so that outputs of a lower-level learner becomes inputs of a higher level learner. One of the most widely used ensemble approaches is random forest (Breiman, 2001a), which combines bagging with random selection of features.

With a statistical insight (Friedman, Hastie, Tibshirani, et al., 2000), regression boosting can be viewed as a “functional gradient descent”, which minimizes a loss function. Ability to select a suitable loss function (beyond square error) greatly increases the power of boosting, including zero-one loss, log loss, exponential loss, etc. For example, some loss functions (e.g. absolute error and Huber error) are less sensitive to outliers than other (square error).

Boosted regression starts with a single tree of a prescribed (typically low) complexity and fits it according to an algorithm for fitting binary trees using a standard approach (e.g., CART). The next tree fits the residual – the error between the estimation of the first tree and the data, and the process continues in a sequential manner: each subsequent tree is “an expert of the errors of its predecessor”. In the end, the trees are added together resulting in a model with an improved accuracy. BRTs have a nice property that existing trees do not change as more trees are added, much like Fourier coefficients in Fourier function approximation (and in contrast to

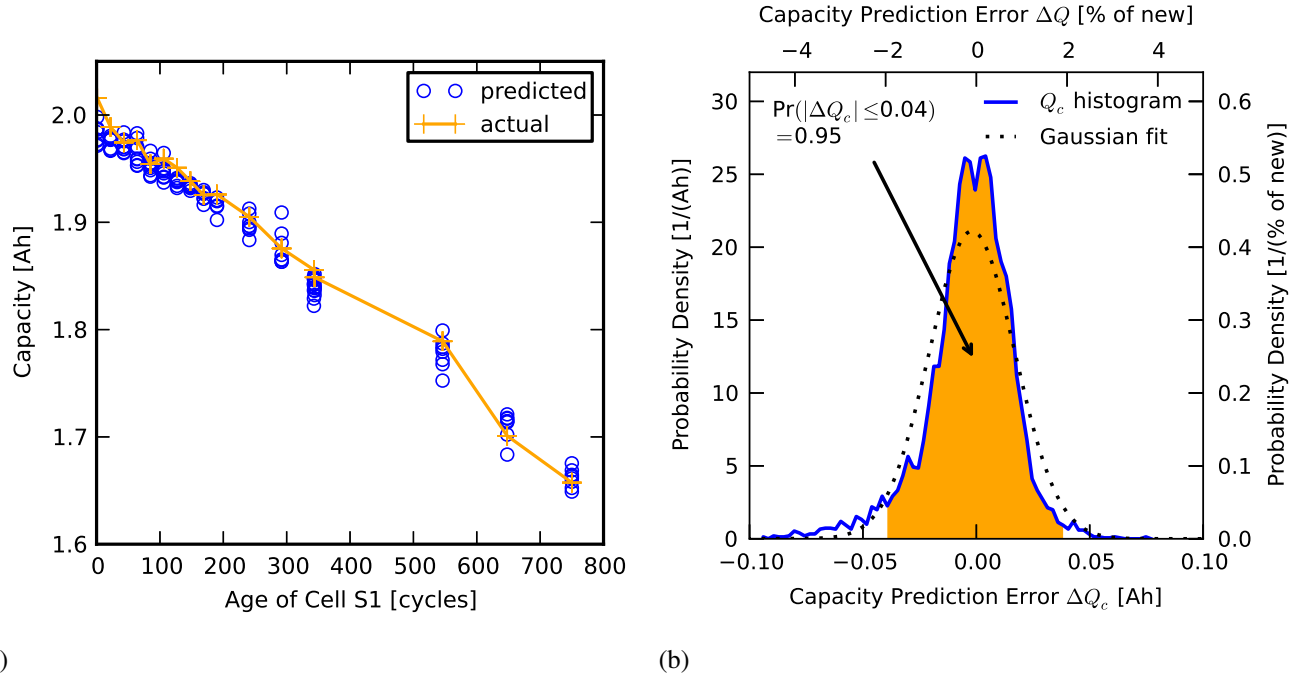


Figure 9. BRT performance (a) Actual and estimated capacity as a function of the cell age. (b) A summary histogram of the performance of all BRT models on each model's verification data.

polynomial coefficients in polynomial fits). The training process is controlled by tree complexity, which controls the number of interactions among features; learning rate, or shrinkage, which controls the contribution of a tree in the growing model; and number of trees, which are not independent. For example, decreasing learning rate increases the number of trees. In summary, the advantages of decision trees are that they have the ability to handle heterogeneous data, do not require pre-processing, support different loss functions, automatically detect feature interactions, and execute (to make predictions) rapidly. Chief limitations are that they are slow to train, require careful tuning during training, and do not extrapolate. Unlike bagging, where individual models are trained independently and the process lends itself to parallel processing, boosting is a staged process, with limited potential for acceleration.

In our application, we used the data from a single step charge or discharge to estimate the parameters of the terminal port, then used a BRT model to estimate the hidden port parameter of interest from the terminal port parameters; this is the same as in the Bayesian inference approach, with Bayesian posterior estimation replaced by a BRT. While flexibility of BRTs loss function allows exploitation of multidimensional outputs, essentially multiple BRTs are needed, one for each of the outputs. For simplicity, we chose to test performance of BRTs on estimation of only one of the two hidden parameters. Q_c was selected over SOC because it is both more

critical and more difficult to estimate.

For our experiments, we used data from six cells for training and data from the remaining two cells to verify the training quality. We trained individual BRTs for the double exponential using model parameter values for all 28 combinations of eight cells taken six (for training) at a time, using the remaining two cells for the verification. To accomplish this, we used the *scikit-learn* Python software library (Pedregosa et al., 2011) to train and evaluate the BRTs. The BRT training was performed using *GradientBoostingClassifier* (Prettenhofer & Louppe, n.d.) with *least absolute deviation* ('lad') loss function, the learning rate set to 0.1, 200 boosting stages, and the maximum depth of individual regression estimators set to 10.

Figure 9a shows the estimated and actual capacity for a randomly selected cell, labeled "S1". The estimated values (open circles) for a given cell age correspond to each of the charging steps during the step test. As shown, each estimation is close to the value measured during the step charge test (the solid line).

Figure 9b likewise shows a histogram of the estimation error of Q_c for the verification cells' data, with results from all 28 models combined. The estimation errors' mean and standard deviation are -0.0019 and 0.019 Ah, respectively. The actual error is skewed toward negative estimation errors, which correspond to underestimates of the cell capacity, which is desirable for our target application domain. The model gen-

eralizes well; 95 % of the estimation errors are within ± 0.04 Ah or ± 2 % of new cell capacity.

4. CONCLUSIONS AND FUTURE WORK

We presented a method for assessing a used cell's capacity and state of charge via estimation of the parameters of the double-exponential model (Figure 3). This approach required characterization of the cell family over the range of interest for capacity and state-of-charge. The model parameters were extracted from either time domain or the frequency domain, but only time-domain data were subjected to full data analysis. Two approaches were considered for estimating the capacity: Bayesian inference and boosted regression trees. Both methods demonstrated statistical effectiveness in estimation for a set of eight tested batteries, albeit without perfect estimation. Noting however, from Figure 1, that the intrinsic, in-class spread of the cells with respect to Q_c is ± 2 %, we conclude the achieved accuracy of assessment with the accuracy of ± 2 % with the capacity estimation error of less than 5 % is reasonable. While the Bayesian inference approach employs a model and allows direct human interpretation of the results; BRT implementation is more of an *algorithmic* (Breiman, 2001b) approach: less transparent, but ultimately yields higher accuracy.

Future work may include investigation of performance when training data includes an indicator separating step charge and step discharge data, or experimenting with bias-variance trade-offs in neural network models employed in Bayesian approach. Also, because the single exponential model provides a combined representation of the short- and long-term transients, additional investigations into ways to improve its performance and decrease the testing time may lead to simpler practical solutions.

ACKNOWLEDGMENTS

This work was made possible by the Office of Naval Research under Award No. N0004-07-1-0823 and New York State Pollution Prevention Institute. We gratefully acknowledge the help of our colleagues from Rochester Institute of Technology: Robert Kosty for his assistance in collecting data and Scott Dewey for setting up of the test processes. All data graphs were produced using the *matplotlib* Python library (Hunter, 2007). The authors are grateful to J.D. Hunter and matplotlib open-source community for creating this flexible tool for data visualization.

REFERENCES

- Amine, K., Chen, C., Liu, J., Hammond, M., Jansen, A., Dees, D., ... Henriksen, G. (2001). Factors responsible for impedance rise in high power lithium ion batteries. *Journal of power sources*, 97, 684–687.
- Bishop, C. M. (1995). Neural Networks for Pattern Recognition. In (p. 137-140). Oxford: Oxford University Press.
- Bishop, C. M. (1st ed. 2006. Corr. 2nd printing). *Pattern recognition and machine learning*. New York: Springer.
- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2), 123–140.
- Breiman, L. (2001a). Random forests. *Machine learning*, 45(1), 5–32.
- Breiman, L. (2001b). Statistical modeling: The two cultures. *Statistical Science*, 16(3), 199–231.
- Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.
- Broussely, M., Biensan, P., Bonhomme, F., Blanchard, P., Herreyre, S., Nechev, K., & Staniewicz, R. (2005). Main aging mechanisms in Li ion batteries. *Journal of power sources*, 146(1), 90–96.
- Chen, M., & Rincon-Mora, G. (2006). Accurate electrical battery model capable of predicting runtime and IV performance. *Energy Conversion, IEEE Transactions on*, 21(2), 504–511.
- Dees, D., Gunen, E., Abraham, D., Jansen, A., & Prakash, J. (2005). Alternating current impedance electrochemical modeling of lithium-ion positive electrodes. *Journal of the Electrochemical Society*, 152, A1409.
- Duda, R., Hart, P., & Stork, D. (2000). Pattern classification. 2nd edn Wiley. In (p. 287-288).
- Elith, J., Leathwick, J. R., & Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, 77(4).
- Freund, Y., & Schapire, R. E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory* (pp. 23–37).
- Friedman, J., Hastie, T., Tibshirani, R., et al. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2), 337–407.
- Goebel, K., Saha, B., Saxena, A., Celaya, J. R., & Christophersen, J. P. (2008). Prognostics in battery health management. *IEEE instrumentation & measurement magazine*, 11(4), 33.
- Hawkins, J. (1994). Some field experience with battery

- impedance measurement as a useful maintenance tool. In *Telecommunications Energy Conference, 1994. INT-ELEC'94., 16th International* (pp. 263–269).
- Hunter, J. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 90–95.
- Jones, E., Oliphant, T., Peterson, P., et al. (2001–). *SciPy: Open source scientific tools for Python*. Retrieved from <http://www.scipy.org/>
- Lasia, A. (1999). Electrochemical impedance spectroscopy and its applications. *Modern aspects of electrochemistry*, 32, 143–248.
- Liu, D., Pang, J., Zhou, J., Peng, Y., & Pecht, M. (2013). Prognostics for state of health estimation of lithium-ion batteries based on combination Gaussian process functional regression. *Microelectronics Reliability*, 53(6), 832–839.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Plett, G. L. (2004). Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation. *Journal of power sources*, 134(2), 277–292.
- Prettenhofer, P., & Louppe, G. (n.d.). Gradient Boosted Regression Trees in Scikit-Learn. In *PyData, London 2014*.
- Saha, B., & Goebel, K. (2008). Uncertainty management for diagnostics and prognostics of batteries using Bayesian techniques. In *Aerospace Conference, 2008 IEEE* (pp. 1–8).
- Saha, B., & Goebel, K. (2009). Modeling Li-ion battery capacity depletion in a particle filtering framework. In *Proceedings of the annual conference of the prognostics and health management society* (pp. 1–10).
- Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2009). Prognostics methods for battery health monitoring using a Bayesian framework. *Instrumentation and Measurement, IEEE Transactions on*, 58(2), 291–296.
- Sankararaman, S., Daigle, M., Saxena, A., & Goebel, K. (2013). Analytical algorithms to quantify the uncertainty in remaining useful life prediction. In *Aerospace Conference, 2013 IEEE* (pp. 1–11).
- Schaul, T., Bayer, J., Wierstra, D., Sun, Y., Felder, M., Sehnke, F., ... Schmidhuber, J. (2010). PyBrain. *Journal of Machine Learning Research*, 11, 743–746.
- Spotnitz, R. (2003). Simulation of capacity fade in lithium-ion batteries. *Journal of Power Sources*, 113(1), 72–80.
- Tröltzsch, U., Kanoun, O., & Tränkler, H. (2006). Characterizing aging effects of lithium ion batteries by impedance spectroscopy. *Electrochimica acta*, 51(8-9), 1664–1672.
- Vetter, J., Novak, P., Wagner, M., Veit, C., Möller, K.-C., Besenhard, J., ... Hammouche, A. (2005). Ageing mechanisms in lithium-ion batteries. *Journal of power sources*, 147(1), 269–281.
- Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2), 241–259.
- Zhang, J., & Lee, J. (2011). A review on prognostics and health monitoring of Li-ion battery. *Journal of Power Sources*, 196(15), 6007–6014.

BIOGRAPHIES



Nenad Nenadic received his B.S. in Electrical Engineering from University of Novi Sad (Novi Sad, Serbia) in 1996 and his MS and Ph.D. in Electrical and Computer Engineering from University of Rochester (Rochester, NY, USA) in 1998 and 2001, respectively. He joined Kionix Inc. in 2001, where he worked on design, analysis, and testing of micro-electromechanical inertial sensors. Since 2005, he has been with Center for Integrated Manufacturing Studies (CIMS) at Rochester Institute of Technology, where he is currently a Research Associate Professor. His research interest include design, analysis, and monitoring of electrical and electromechanical devices and systems. He has two patents in electromechanical design and seven publications. He co-authored a textbook “Electromechanics and MEMS”. He is a member of IEEE and AHS.



Howard Bussey Howard Bussey earned the Masters degree in electrical engineering from the University of California (Berkeley, CA, USA). He is a co-founder of Tixlers’ Letters, providing educational resources. He was a Senior Staff Engineer in the Golisano Institute of Sustainability, and has also worked at Estman Kodak, Bell Communication Research and Bell Laboratories. He has six patents and 15 publications. He is a member of IEEE, Eta Kappa Nu, and Tau Beta Pi.



Paul Ardis Paul Ardis received his B.S. in Computer Science from Purdue University (WestLafayette, IN, USA) in 2005 and his M.S. and Ph.D. in Computer Science from the University of Rochester (Rochester, NY, USA) in 2007 and 2009 respectively. He is a Lead Scientist at GE Global Research, and was formerly a Research Associate Professor at the Center for Integrated Manufacturing Studies (CIMS) at Rochester Institute of Technology and Research Scientist under contract to the Air Force Research Laboratory. He holds two patents in signal processing and has published eleven scholarly articles in machine learning, computer and human vision, biometrics, and machine diagnostics and prognostics. His research interests include data-driven predictive modeling, decision theory, surveillance, natural language processing, and machine sensing.



Michael Thurston received his B.S. and M.S. in Mechanical Engineering from Rochester Institute of Technology (Rochester, NY, USA) in 1988, and his Ph.D. in Mechanical and Aerospace Engineering from the University of Buffalo (Buffalo, NY, USA) in 1998. He is the Technical Director and Research Associated Professor at the Center of Integrated Manufacturing Studies at Rochester Institute of Technology. He formerly held positions in air conditioning system development at General Motor and Delphi, and as a Researcher at the Applied Research Laboratory at Penn State University. He holds 7 patents in the areas of air conditioning and asset health monitoring. His research interests include: sustainable design and production, condition based maintenance and prognostics, and asset health management. He is a member of the Society of Automotive Engineers, and was awarded the Boss Kettering Award for product innovation by Delphi.