Battery State of Charge Estimation in Automotive Applications using LPV Techniques

Yiran Hu and Stephen Yurkovich
Center for Automotive Research, The Ohio State University

Abstract—One of the most difficult problems in battery pack management aboard a P/H/EV is the estimation of the state of charge (SoC). Many proposed solutions to this problem have appeared in the literature; in particular, model-based extended Kalman filter approaches have shown great promise. However, the computational burden of implementing an extended Kalman filter is significant. Moreover, some parameters needed to make the extended Kalman filter function correctly are difficult to estimate from measured data. This paper proposes an SoC estimator design using linear parameter varying (LPV) system techniques that provides a low computational alternative to the extended Kalman filter. The stability of this estimator can be verified analytically. The performance of the estimator in terms of convergence and tracking is verified experimentally on an isothermal dataset taken from a lithium ion battery cell.

I. INTRODUCTION

In recent years the high cost of energy, particularly from fossil fuels, has led to innovations in advanced powertrains, such as the hybrid electrical vehicle (HEV), plug-in hybrids (PHEV) and full-electric vehicles (EV). P/H/EVs typically contain two separate power sources, with the primary form being an internal combustion engine and an electrical machine. When the two power sources are used in tandem, they complement one another, and the result is a significant reduction in fuel consumption and emissions output, without dramatic loss of performance and drivability.

The enabling technology behind current P/H/EV development is the rechargeable battery pack. Basically, a typical pack serves as a buffer so that electrical energy can be transferred between various components of the powertrain. The current and previous generations of HEVs primarily use NiMH batteries because they have better energy density than lead acid batteries and are less costly than lithium ion batteries. With the improvement in lithium ion technology, lithium ion batteries are now ready to become the main energy storage source in P/H/EV applications due to their superior energy/power density. In order for lithium ion battery packs to function efficiently and safely, the state of charge (SoC) of the battery inside the battery pack must be precisely maintained. This way the desired functions of the battery pack, such as powering the electric machine(s) to assist in a launch from a complete stop or recovering energy from regenerative braking, can be preformed when most advantageous. Precise management will also avoid undesired behaviors like over-discharging, which can result in premature aging (thus shorter battery life), and over-charging, which can result in catastrophic failure in the form of thermal runaway.

Estimating the SoC in real time, however, is a challenging problem. Formally, SoC is defined as the ratio of available ampere-hour (Ah) to the total Ah available when the battery is fully charged (namely, the capacity). If current through the battery is monitored precisely, the SoC can be calculated using a Coulomb counting process:

\[
z(t) = z(0) - \frac{1}{K} \int_{0}^{t} I dt \tag{1}
\]

where \(z(t)\) is the SoC as a function of time and \(K\) is a factor that contains the nominal capacity as well as any conversion factors needed to match units. However, this calculation is not reliable in real time because the noise in the current measurement over time can cause the estimate to drift significantly. To improve the Coulomb counting methods, an open circuit voltage (OCV) measurement is often used to reset the integrator. As noted by research on battery electrochemistry, the OCV has a one-to-one relationship with the SoC of the battery. Therefore, by measuring the OCV and then inverting the map between OCV and SoC, SoC can be calculated. However, OCV can only be measured after the battery has been rested a significant period of time (often several hours). In applications where the batteries are operated continuously for several hours, this method is not desirable.

Many methods have been suggested to overcome the shortcomings of Coulomb counting and OCV measurement; [1] offers a good summary of these methods. For example, in [2], a fuzzy logic system is constructed between the impedance of the battery at several frequencies and the SoC. However, to use this system to estimate the SoC in real time, a sinusoidal signal must be injected into the battery pack to measure the impedance. This is undesirable for in-vehicle operation. Another attempt using a fuzzy logic based estimator is seen in [3], where the author augments the standard Coulomb counting with an efficiency factor that is an adaptable fuzzy logic function of the current and temperature. While more implementable, this approach still requires periodic knowledge of the true SoC to adapt appropriately.

A more complete algorithm is offered by [4] where an extended Kalman filter is used for the SoC estimation. In this model based approach, the author first identifies a slightly
nonlinear discrete state space model for the battery, which includes the SoC as a state. Then an extended Kalman filter is applied to the model to estimate the states. The results show good convergence after initializing the estimator with a large error. The drawbacks of this approach are two fold. First an extended Kalman filter requires a good statistical model of the model error variance. Because the SoC mostly affects the OCV, which is a very flat curve in SoC, the SoC estimate is particularly sensitive to a poorly chosen model error variance. In addition, selecting the error variance is difficult to do because the model contains components that are not directly measurable. Secondly, to implement an extended Kalman filter in real time requires significant computational resources: within every sampling period, the estimated error covariance of the states must be propagated through the model of the system in order to calculate the Kalman gains. Therefore, in some cases it may not be practical to implement such an estimator on board in real time. From simulations it can be seen that because the model used for SoC estimation is mostly linear, the Kalman gains converge to quasi-steady values. This suggests that another method using non-recursive gains may offer good performance as well.

In this work, a model-based state estimator is designed for SoC estimation as a low computational alternative to the extended Kalman filter method. In particular, a discrete state space model of a lithium ion battery cell is generated using an equivalent circuit approach. Then a predictor/corrector style estimator is designed based on that model. The estimator is designed in such a way that the corrector gain (which can depend on measurable plant parameters) is calculated a priori so that no recursive calculation of the gain is needed. Furthermore, the convergence of this estimator is ensured using linear parameter varying techniques (for more discussion on LPV systems, see [5], [6], [7], [8]). Compared with the extended Kalman filter, this approach provides lower computational intensity and analytically assessable stability. As this is a proof-of-concept, the work reported herein is focused on room temperature operation (isothermal) only, whereas an extension to a temperature range is briefly discussed.

II. BATTERY MODEL

A. Equivalent Circuit Model

In this section, the battery model used to construct the estimator is introduced. The model generated here follows the work done previously in [9] and [10], and is based on an equivalent circuit representation as shown in Figure 1. This equivalent circuit is the standard Randle equivalent circuit and is comprised of an ideal open circuit voltage, an internal resistance $R_0$, and $n$ parallel $RC$ circuits. While not the most sophisticated model structure possible, this structure can generally approximate the battery behavior well, as long as the frequency range of the input signal is limited (for more on battery frequency response, see [11]).

The dynamic equation that describes the voltage across the $i^{th}$ RC circuit is given by

$$\frac{dV_i}{dt} = -A_i V_i + A_i B_i I,$$  \hspace{1cm} (2)

where $A_i = 1/(R_i C_i)$ and $B_i = R_i$. In general, $A_i$, $B_i$, and $R_0$ are functions of the SoC, temperature, and current (in particular the direction of the current). Considering an isothermal condition at room temperature (eliminating the temperature dependence), the dependence on SoC is minimum. Furthermore, the nonlinear dependence on current magnitude is also very small. Therefore, for this paper, these quantities are taken to be binary functions of the current direction only (zero current is resolved using a hysteresis approach; for example, zero current state after charging is still considered charging).

The OCV in this equivalent circuit model is a function of the SoC and temperature (though temperature dependence is ignored since isothermal condition is assumed). The dependence on SoC at a fixed temperature has the following characteristics: (1) the OCV increases monotonically with SoC; (2) the OCV drops in an exponential fashion as the SoC approaches 0%; (3) the OCV rises in an exponential fashion as the SoC approaches 100%; and (4) in the large region between 20 and 85% SoC, the relation is nearly linear. Given these characteristics, the OCV is modeled herein by a double exponential function as

$$V_{oc}(z) = V_0 + \alpha(1-e^{-\beta z}) + \gamma z + \xi (1-e^{-\epsilon z}),$$ \hspace{1cm} (3)

where $z$ is the SoC in decimal form (i.e. $z \in [0,1]$) and $\alpha$, $\beta$, $\gamma$, $\xi$ are constant parameters. Figure 2 shows that this functional form does indeed approximate the measured open circuit voltage curve very well, particularly inside the SoC region relevant to P/H/EV operation (typically 10% to 90%).

Because the ultimate purpose for this study is to estimate the SoC of the battery using a state estimator, the SoC must be included as a state. This can be accomplished by using the definition of SoC given in (1). To be explicit, if the capacity of the battery is $C_n$ Ah, then the SoC ($\zeta$) is given by

$$\dot{\zeta} = \frac{-I}{3600 C_n}.$$ \hspace{1cm} (4)

Note that aside from being a component of the model, this formulation of SoC is also used to compute the true SoC for experimental datasets after post-processing. Because
the post-processing effectively eliminates noise bias and erroneous sensor calibration, the SoC calculated this way is very accurate.

The last component of the model is a hysteresis element. This element is used to account for the battery behavior where the rested voltage is different at the same SoC depending on the previous excitation [12]. The model selected here to describe the hysteresis voltage $V_h$ is a simple first order dynamic equation given by

$$V_h = \Gamma |I|(M(z, T) - V_h)$$

where $\Gamma$ can be thought of as a hysteresis transition factor (i.e. how fast the hysteresis occurs) and $M$ is the maximum amount of hysteresis voltage that can occur for a given temperature ($T$). Based on physical principles, $M$ should be positive when the battery is being charged and negative when discharging. This is reflected in the dependence of $M$ on $z$, which is a simplified way of writing the current direction.

Finally the battery terminal voltage can be expressed as the sum of all the voltage components discussed previously:

$$V_{bat} = V_{oc} - R_0 I - \sum_{i=1}^{n} V_i - V_h.$$  

B. Model Discretization

Although an estimator can be designed directly using the continuous time model derived in the previous section, because the eventual application is meant for on-board implementation, a discrete version of the continuous model is more practical. This section explains how the continuous model can be discretized.

Given a selected sampling period $T_s$ and a continuous linear system of the form

$$\dot{x} = Ax + Bu$$

where $x, u \in \mathbb{R}$ and the coefficients $a, b$ and the input $u$ are assumed to be constant over each sampling period, the exact discrete equivalent of the system is given by

$$x[k+1] = e^{-aT_s} x[k] + \frac{b}{a} (1 - e^{-aT_s}) u[k].$$

Using this fact, equations (2), (5) can be discretized as follows:

$$V_i[k+1] = \alpha_i V_i[k] + \beta_i I[k]$$

$$V_h[k+1] = \gamma(I[k]) V_h[k] + \kappa(I[k]) M(z, T)$$

where

$$\alpha_i = e^{-A_i T_s}$$

$$\beta_i = B_i (1 - e^{-A_i T_s})$$

$$\gamma(I[k]) = e^{-\Gamma |I[k]|}$$

$$\kappa(I[k]) = 1 - e^{-\Gamma |I[k]|}$$

Equation (4) simply represents an integrator, and can therefore be discretized as

$$z[k+1] = z[k] - \frac{T_s}{3600 C_n} I[k]$$

The complete model then can be written in state variable form as

$$\begin{bmatrix}
    z[k+1] \\
    V_i[k+1] \\
    \vdots \\
    V_n[k+1] \\
    V_h[k+1]
\end{bmatrix} =
\begin{bmatrix}
    1 & 0 & \ldots & 0 & 0 \\
    0 & \alpha_1 & \ldots & 0 & 0 \\
    \vdots & \vdots & \ddots & \vdots & \vdots \\
    0 & 0 & \ldots & \alpha_n & 0 \\
    0 & 0 & \ldots & 0 & \gamma(I[k])
\end{bmatrix}
\begin{bmatrix}
    z[k] \\
    V_i[k] \\
    \vdots \\
    V_n[k] \\
    V_h[k]
\end{bmatrix}$$

$$+ \begin{bmatrix}
    -\frac{T_s}{3600 C_n} & 0 \\
    0 & \beta_1 \\
    \vdots & \vdots \\
    0 & \beta_n \\
    0 & \kappa(I[k])
\end{bmatrix}
\begin{bmatrix}
    I[k] \\
    M(z, \ldots, T)
\end{bmatrix}$$

where the output equation (battery terminal voltage) is given by

$$V_b[k] = V_{oc} (z[k]) - R_0 I[k] - \sum_{i=1}^{n} V_i[k] - V_h[k].$$

To simplify the notation, write (13) compactly as

$$x[k+1] = Ax[k] + Bu[k]$$

where $x \in \mathbb{R}^{n+2}$ is the state, $A \in \mathbb{R}^{(n+2) \times (n+2)}$, $B \in \mathbb{R}^{n+2}$ is the input matrix, and $u \in \mathbb{R}^2$ is the input. Note here that $A, B$ are actually parameter varying with the parameter being the input current $I$. Also note that because $V_{oc}$ is a nonlinear function of the state $z$, the system is nonlinear.

III. ESTIMATOR DESIGN

Given the model of the plant in (14) and (15), the goal now is to design a state estimator for the state variable $x$ so that the estimates will converge asymptotically to the true values. The design will be done in two steps for clarity.
A. Simplified Case

First, because the system is nonlinear in the output, direct application of linear system tools will not provide a satisfactory answer, especially with regard to stability. Therefore, first assume that the term $V_{oc}(z[k])$ can be approximated as

$$V_{oc}(z[k]) = V_0 + \tilde{z} c[k]$$  (16)

where $\tilde{c}$ is a constant. Under this assumption, the system in question becomes an LPV system, whose parameter (namely $I[k]$) can be measured.

Now design the estimator in the predictor/corrector observer format. Let $\hat{x}_p$ be the prediction state and $\hat{x}$ the corrected state. Then

$$\dot{\hat{x}}_p[k+1] = A\hat{x}_p[k] + Bu[k] \tag{17}$$

$$\dot{\hat{x}}[k+1] = \dot{\hat{x}}_p[k+1] + K(I[k])(V_b[k+1] - \hat{V}_b_p[k+1]) \tag{18}$$

where $\hat{V}_b_p[k+1]$ is the predicted battery terminal voltage for time $k+1$ given by

$$\hat{V}_b_p[k+1] = V_0 + Cx_p[k+1] \tag{19}$$

where $C = [\tilde{c}, -1, \ldots, -1, -1]$ can be considered the output matrix.

For stability requirements, (17) and (18) are combined as

$$\dot{x}[k+1] = Ax[k] + Bu[k] + K(I[k])(V_b[k+1] - \hat{V}_b_p[k+1]) \tag{20}$$

The term $V_b[k+1] - \hat{V}_b_p[k+1]$ can be evaluated as

$$V_b[k+1] - \hat{V}_b_p[k+1] = C(x[k+1] - \hat{x}_p[k+1]) = CA(x[k] - \hat{x}[k]) \tag{21}$$

Form the error variable $\tilde{x}[k] = x[k] - \hat{x}[k]$. Using (20) and (21), $\tilde{x}[k]$ satisfies the equation

$$\tilde{x}[k+1] = (A - KCA)\tilde{x}[k] \tag{22}$$

Denote the matrix $A - KCA$ by $\tilde{A}$ and form a quadratic Lyapunov function

$$L(\tilde{x}[k]) = \tilde{x}^T[k] P(I[k]) \tilde{x}[k]. \tag{23}$$

Note here that $P$ is allowed to be a function of the parameters for maximum flexibility. However, in many cases, a solution exists even if $P$ does not depend on $I$. Therefore, when solving this problem, it is always wise to start by solving for a non-parameter dependent $P$. Only when that fails, parameter dependence is considered.

From the Lyapunov stability theorem, if $P$ is positive definite (regardless of $I$), and $L(\tilde{x}[k+1] - L(\tilde{x}[k]) < 0$, then $\tilde{x}[k+1]$ tends to zero asymptotically. In other words,

$$L(\tilde{x}[k+1]) - L(\tilde{x}[k]) = \tilde{x}^T[k+1] P(I[k+1]) \tilde{x}[k+1] - \tilde{x}^T[k] P(I[k]) \tilde{x}[k]$$

$$= \tilde{x}^T[k] (\tilde{A}(I[k]) P(I[k+1]) - P(I[k])) \tilde{x}[k]$$

$$< 0 \tag{24}$$

This requires $\tilde{A}(I[k]) P(I[k+1]) - P(I[k])$ to be a negative definite matrix for any values of the $I[k]$ and $I[k+1]$. Therefore, the problem can be summarized as a single linear matrix inequality (LMI)

$$\begin{bmatrix} \tilde{A}(I[k]) P(I[k+1]) - P(I[k]) & 0 \\ 0 & -P(I[k]) \end{bmatrix} < 0 \tag{25}$$

This equation highlights the difference between this estimator design and typical estimator design for linear time invariant (LTI) systems. For LTI systems, $\tilde{A}$ and $P$ would be independent of time $k$ and therefore well established tools exist to solve this equations (also known as the discrete Lyapunov equation) for unknown $P$ and $K$. In the LPV case, however, the LMI depends on a parameter. The general approach is to grid the parameter space and solve the LMI at each grid point via a convex optimization process [7], [13].

It is also important to note that two time indices appear for the matrix $P$. LMI solvers are not efficient in dealing with two time indices. However, if $P$ is assumed to be of an affine form, i.e.

$$P(I[k]) = P_0 + I[k] P_1, \tag{26}$$

then [14] shows that if

$$\begin{bmatrix} \tilde{A}(I) (P(I) + \Delta_1 P_1) \tilde{A}(I) - P(I) & 0 \\ 0 & -P(I) \end{bmatrix} < 0 \tag{27}$$

holds for any $I$, then (25) will also hold. Note that $\Delta_1$ is the maximum change in $I$ over one sampling period. Using this simplification, the explicit dependence on two time indices is removed and standard LMI tools can be applied. If $P$ is selected to be independent of the parameter, then this latter portion will not be needed, which would simplify the problem significantly.

B. Nonlinear Output Equation

In the previous section, the problem of estimator design is solved for the case when the battery OCV curve is approximated using a linear function. While this assumption does hold reasonably well when the battery is operated between approximately 30% to 85%, it typically does not hold when the SoC is outside of this range. However, the previous design can still be used with some modification. The predictor and corrector defined in (17) and (18) are still used. But the predicted battery terminal voltage is changed to use the nonlinear output equation as

$$\hat{V}_b_p[k+1] = V_{oc}(\hat{z}_p[k+1]) - R_0(I[k]) I[k]$$

$$- \sum_{i=1}^{n} V_i[k+1] - V_h[k+1] \tag{28}$$

Then the corrector feedback term $V_b[k+1] - \hat{V}_b_p[k+1]$ has $V_{oc}(\hat{z}_p[k+1]) - V_{oc}(\hat{z}_p[k+1])$ as a term instead of $\tilde{c}(z[k] - \hat{z}_p[k+1])$. We can write

$$V_{oc}(z[k+1]) - V_{oc}(\hat{z}_p[k+1])$$

$$= \hat{V}_{oc}(z[k+1] - \hat{z}_p[k+1]) \frac{(z[k+1] - \hat{z}_p[k+1])}{(z[k+1] - \hat{z}_p[k+1])} \tag{29}$$
where \( \bar{\chi} \) represents the divided difference shown in the previous line. Note that because the OCV function is a continuously differentiable, monotonically increasing function, \( \bar{\chi} \) is well defined and not sign varying. In other words, there exists \( \bar{\chi}_1, \bar{\chi}_2 \in \mathbb{R} \) such that \( 0 < \bar{\chi}_1 < \bar{\chi} < \bar{\chi}_2 \). As a result, let \( C = [\bar{\chi} -1 \ldots -1 -1] \) be the output matrix. Then the error dynamics can again be written as in (22), except \( \bar{A} = A - KCA \) is now a function of two parameters, \( I[k] \) and \( \bar{\chi} \). A Lyapunov function of the form (23) can still be used. However, because \( \bar{\chi} \) is uncertain, \( P \) must be selected to work for any \( \bar{\chi} \in [\bar{\chi}_1, \bar{\chi}_2] \). In other words, the Lyapunov function must be robust with respect to \( \bar{\chi} \). This is in general difficult to do. However, because the bound of \( \bar{\chi} \) is actually fairly narrow and not sign-varying, this approach can work. Then the same analysis can be done to arrive at the same LMI specified in (27). The only difference now is that the choice of \( P \) and \( K \) must allow this LMI to hold for all \( \bar{\chi} \in [\bar{\chi}_1, \bar{\chi}_2] \) and for all \( I \).

C. A Note on Designing the Corrector Gain

The results of the previous subsection outline the LMI conditions that the estimator gains must satisfy to allow the estimator to converge asymptotically. To design the gains, one approach is to simply use an LMI tool to solve the LMI for both the gain \( K \) and the matrix \( P \) (for instance, Matlab software suite from Mathworks has a fully functional LMI solver). While that certainly will generate a valid \( K \), the result may not possess any optimality in terms of convergence speed and tracking accuracy. An alternative approach is to find \( K \) via an optimization over a comprehensive dataset so that some optimal convergence property (via some cost function) is satisfied. Then the optimized gains can be back checked via the LMI to ensure stability. In this way, the LMI problem is simplified so that the LMI solver is more likely to find a solution. The gains found via the optimization typically have a good chance of producing a stable estimator because the dataset is comprehensive.

D. Computational Complexity

One goal of this work is to have a SoC estimator design that has lower computational complexity than the extended Kalman filter. To apply an EKF to an \( n^{th} \) order state space model, a \( n \times n \) error covariance matrix must be propagated through the system model at every sample to compute the Kalman gain. This requires two multiplication of \( n \times n \) matrices. The other steps of the EKF is essentially one iteration of the system model. Propagating the covariance matrix is approximately \( n \) single iterations of the system model. Therefore, the step of the covariance matrix propagation can cost \( n \) times as many computations as the other steps. Because the LPV method only requires one iteration of the system model, it requires at most half the number of computations (a very conservative estimate) as the EKF method. Therefore from the computational perspective, the advantage of this method is significant.

IV. RESULTS

In this section, the estimator results are presented. The data collected is produced from an A123 lithium ion iron-phosphate battery that has a nominal capacity of 2.3 Ah and nominal voltage of 3.2 V. The experiments were carried out in the battery characterization and aging laboratories at the Center for Automotive Research at Ohio State University. This laboratory contains multiple stations capable of running tests of arbitrary lengths and can accommodate 24/7 unattended operation if needed. Each station testing a single cell or module consists of a pair of programmable load (800W) and power supply (1.2 kW), a data acquisition and control computer and Peltier junctions, and associated controllers to provide a controlled thermal environment for the battery specimen under test for temperatures ranging between -25 to +60 °C. The true SoC for each dataset is found by integrating the post-processed measured current (care is taken in sensor calibration to ensure accuracy). No special factor is used to characterize the charge/discharge efficiency (thus both factors are taken to be 1). Each dataset is always preceded by a charge to 100% via manufacture specifications so that the initial condition for the current integration is known. Datasets are usually limited to three hours or less to minimize the effect of SoC drift.

Three RC circuits are selected for the model, which is identified over a dataset containing a series of asymmetrical steps; this series is designed to bring the SoC from 90% to 10% and then back to 90%. The temperature of the experiment is kept at a constant value of 25°C via the use of Peltier junctions. As Figure 3 shows, the fit of the model to the actual measurement is very good.

As the corrector design process outlined, the gains can depend on the operating conditions. In this case, the only measurable operating condition is the current. Thus we can schedule the gains on the current. However, the model’s primary dependence on the current is on the current direction. The model difference between charging and discharging is
small. Therefore it is worth trying a corrector gain that does not depend on current. Not only will this simplify the LMIs, it also makes the gain more likely to be robustly stable. The corrector gains are found via the optimization method discussed earlier. To ensure that the searching process is unbiased, the estimator is initialized with random initial conditions. Furthermore the starting index of the dataset is varied to simulate cases when the battery is not initially at rest. A genetic algorithm is used to pick the gain so that over the various initial conditions the mean squared error between the estimated SoC and the true SoC is minimized. The dataset over which the optimization is performed is the asymmetrical step profile used for model identification. After that, the LMI toolbox in Matlab is used to perform a feasibility check to make sure that a positive definite $P$ exists that satisfies (27). In this particular case, a constant $P$ matrix can be found to satisfy this LMI.

The performance of the estimator can be verified experimentally. As Figures 4 shows, the estimator does a good job of quickly converging to the correct SoC from a large initialization error. Then the estimator continues to track the SoC throughout the dataset. We can see from the figure that there is some error near the lower SoC area when the battery is at rest. This is an error that results from the inaccuracies in the model’s open circuit voltage curve. Despite this, the RMS error of the estimator over the entire dataset is less than 4% SoC. To further ensure robustness, a separate dataset is used to check the performance of the estimator. This dataset is a constant discharging pulse that was used to measured the capacity of the battery. In other words, this dataset has the SoC decreasing linearly from 100% to 0%. As Figure 5 shows, the estimator again quickly converges to the correct SoC from a large initial error and then continue to track the SoC with very small error. Some error is expected especially at the lower SoC because the model is only identified between 10 to 90% SoC. Nonetheless, the RMS error over the entire dataset is again less than 4%.

A. Extension to Multi-temperatures

The results shown here are only applicable to the room temperature model. An effective on-board implementation must be valid for all P/H/EV temperature. These results can, however, be extended to work for a temperature range. Here we will outline a basic process to extend the results here to work for all useful temperatures. First, a model must be identified to approximate battery operation inside the temperature range of interest. This would result in a model like (13) whose coefficients are dependent on temperature. Then a predictor-corrector estimator format can again be used. Because the behavior of the battery at lower temperature is significantly different from its behavior at higher temperatures, the corrector gain will likely have to be a function of the temperature (note that this is a good reason why the choice of constant gains at single temperatures can simplify the problem significantly). Given this, LMI conditions can be formulated to ensure that the estimator has the desired convergence property. Like before, an optimization routine can be used to select the gain so that the convergence and tracking performance is adequate over representative datasets.

V. CONCLUSION

In this paper, a SoC estimator is designed using linear parameter varying techniques. This estimator carries a small computational requirement compared to the commonly used extended Kalman filter-based estimator seen in the open literature. Therefore, this design is more amenable to real time implementation. Furthermore, the stability of this estimator can be verified analytically using Lyapunov stability criteria represented by a set of linear matrix inequalities. The design is verified experimentally over a test dataset to show good convergence and tracking properties. Because this is a proof-of-concept, the estimator is only designed for room temperature operation. Work is underway, however, to extend
these results to a full SoC estimator that is valid in all temperature ranges relevant to P/H/EV operations. Similar construction ideas are used to design such an SoH estimator, and those results will appear in an upcoming paper.

REFERENCES


