A Behavioral Algorithm for State of Charge Estimation

Ayca Balkan, Min Gao, Paulo Tabuada, Lei He

UC Los Angeles Electrical Engineering

Electrical Engineering Dept., UCLA, Los Angeles, California, 90095, U.S

abalkan, mingao, tabuada, lhe@ee.ucla.edu

Abstract

In Hybrid Electricle Vehicles (HEV), performing online energy management is an important task to be achieved to reduce emissions, fuel consumption and increase vehicle performance. For this task, estimating the State of Charge (SOC) is needed since it serves as a measure of energy that is left inside an electrochemical battery. A variety of methods to solve this estimation problem have been proposed in the literature. However, most of these methods either assume equivalent circuit models for the battery and thus lose their validity under some discharge conditions or depend heavily on the choice of parameters in the algorithm. In this paper, we use behavioral framework to avoid postulation of a specific model for a battery and develop a new and simple SOC estimation algorithm. Once the problem is formulated as the computation of a specific free response of the battery, algorithm computes this response using only terminal current and terminal voltage measurements. We demonstrate the effectiveness of the algorithm with different discharge profiles using both simulated and real data.

Keywords: Battery management, Battery charge, State of charge

1   Introduction

Estimating the state of charge (SOC) is an important, yet challenging task, in the context of battery modeling and management. The SOC provides a measure of how much energy remains stored in a battery. For example, in hybrid electrical vehicles, SOC can be used for both driver notification and online energy management purposes. In this paper, we propose a new method to obtain the SOC in electrochemical batteries using the behavioral framework [11].

There is a broad literature concerning the SOC estimation problem. In [9] and [4], an extended Kalman filter is used to estimate the parameters of an electrical circuit model for a battery. Similarly, [3] also assumes an electrical circuit but instead of trying to identify the circuit elements, it aims to compute the SOC directly from terminal voltage and terminal current. Combined with the Coulomb counting method, [12] uses online least squares regression to compute the parameters of an equivalent RC circuit. All these methods rely on electrical models to explain the battery behavior and thus have important well-known drawbacks. Firstly, the simplified models cannot represent the behavior of the battery exactly. Secondly, the models are not universal, i.e., they may result in highly erroneous SOC estimates in certain battery types and certain discharge conditions while producing adequate results with other battery types or discharge conditions. Moreover, convergence of these algorithms depend on an initial choice of parameters which are again battery-dependent. Lastly, there are neural network based methods such as those presented in [2] and [10]. The drawback of these methods, in general, is that their performance highly depends on the data used for training.

In recent work by Xiao et al. [13], a new universal SOC algorithm was introduced which, in contrast with the previous work, does not postulate the existence of a model for the battery and does not attempt to identify the parameters in such model. Our approach is inspired by this novel aspect of the work in [13]. Extracting a specific response of a system, using only input

and output data without constructing a model, fits perfectly into the behavioral framework proposed by Jan Willems, see [11]. Hence, in this paper we investigate the adequacy of behavioral system identification methods to the SOC estimation problem. The key observation which is also pointed out in Section 3.1 is that estimating the SOC can be solved by estimating the open circuit battery voltage which amounts to compute a free response. In the same section, we also explain how behavioral methods enable us to compute a free response without a model and in Section 3.2, we describe the algorithm used for this computation. Finally in Section 4, we illustrate the performance of these algorithms in simulations using different battery types and current discharge profiles.

2 Background

In the literature there are two main approaches to the SOC estimation problem: Coloumb counting based and voltage based. Although Coloumb counting is a simple method, it involves the integration of current thus leading to accumulated errors. Furthermore, current integration provides an estimate of the SOC provided that the initial SOC is known, an assumption that is not satisfied in many situations. Thus, it is necessary to use the Coloumb counting based methods together with voltage based methods to prevent accumulation of errors in SOC estimates.

Voltage based methods aim to estimate SOC via the computation of the open circuit voltage V_{oc}. The open circuit voltage can be defined as the value reached by the voltage across the battery terminals when we disconnect the external load from the battery. In addition, it is known, see [4] and [1], that the SOC is approximately an affine function of the steady state value of V_{oc}. A more detailed relationship between the SOC and V_{oc} is discussed in [7]. Given these relationships we focus in this paper on the problem of computing V_{oc}.

3 A Behavioral Algorithm

3.1 V_{oc} estimation as the computation of a free response

Although electrochemical batteries are commonly considered as slowly time varying nonlinear systems, in sufficiently short time windows, they can be seen as linear time invariant (LTI) systems. Hence, when describing our algorithm, we assume that the battery behaves exactly as an LTI system whose (minimal) state-space equations:

$$x(t+1) = Ax(t) + Bu(t),$$
$$y(t) = Cx(t) + Du(t), y \in \mathbb{R}^p, u \in \mathbb{R}^m$$

$$(3.1)$$

require at most n_{max} states, i.e., $x \in \mathbb{R}^{n_{max}}$. We call x the state, u the input, and y the output. In

this paper u models the current entering or leaving the battery and y models the voltage across the battery terminals. Note that we do not know A, B, C, and D and that we will not attempt to determine these matrices. The open circuit voltage at time t_0 can now be seen as:

$$V_{oc}(t_0) = \lim_{t \to \infty} V(t, t_0, 0). \qquad (3.2)$$

where $I : [t_0, +\infty[\to \mathbb{R}$ is the terminal current and $V(t, t_0, I)$ is the terminal voltage of an electrochemical battery at time t when we start feeding I at time t_0. The output generated by a linear system when the input is zero is called a *free response*. If we consider I as the input and V as the output of our system, by equation (3.2) we can say that to find the V_{oc}, it is sufficient to compute the free response of the system. In this paper, our objective is to compute the free response directly from measured data without attempting to construct the state x or the matrices A, B, C, D. We now explain how to compute a free response in a behavioral setting. We start with a few definitions.

Definition 3.1 (Behavior) *The behavior $\mathcal{B}$ of the LTI system (3.1) is the set of all possible input-output trajectory pairs $w = (u, y) \in (\mathbb{R}^{m+p})^{\mathbb{N}}$ that obey equations (3.1). When trajectories are restricted to be T samples long, we denote the behavior as $\mathcal{B}|_{[1,T]}$.*

The behavior $\mathcal{B}$ of the LTI system (3.1) inherits both linearity as well as time-invariance. Linearity of $\mathcal{B}$ amounts to requiring $\mathcal{B}$ to be a vector subspace of $(\mathbb{R}^{m+p})^{\mathbb{N}}$. Time invariance can be expressed as shift invariance.

Definition 3.2 (Shift operator) *The shift operator $\sigma : (\mathbb{R}^{m+p})^{\mathbb{N}} \to (\mathbb{R}^{m+p})^{\mathbb{N}}$ is defined by:*

$$(\sigma(w))(t) = w(t+1) \qquad (3.3)$$

We also define σ^k as the k-fold composition of σ with itself.

Shift invariance can now be formalized as $\sigma(w) \in \mathcal{B}$ for every $w \in \mathcal{B}$.
The computation of free responses requires the manipulation of Hankel matrices.

Definition 3.3 (Block Hankel Matrix) *The L-deep block Hankel matrix of a behavior $w = (u, y) \in \mathcal{B}|_{[1,T]}$ is denoted by $\mathscr{H}_L(w)$ and defined as follows:*

$$\mathscr{H}_L(w) = \begin{bmatrix} w(1) & w(2) & \cdots & w(T-L+1) \\ w(2) & w(3) & \cdots & w(T-L+2) \\ w(3) & w(4) & \cdots & w(T-L+3) \\ \vdots & \vdots & \ddots & \vdots \\ w(L) & w(L+1) & \cdots & w(T) \end{bmatrix}.$$

We note that $\mathscr{H}_L(w)$ can also be understood as a matrix having $w|_{[1,L]}$ in its first block column and having shifted versions of $w|_{[1,L]}$ in the following columns:

$$\mathscr{H}_L(w) = \left[w|_{[1,L]} \mid \sigma(w)|_{[1,L]} \mid \ldots \mid \sigma^{T-L}(w)|_{[1,L]} \right].$$

By shift invariance, every block column of $\mathscr{H}_L(w)$ belongs to $\mathcal{B}|_{[1,L]}$ and since the behavior is a vector space, linear combinations of the block columns of $\mathscr{H}_L(w)$ also belong to $\mathcal{B}|_{[1,L]}$. Let us now divide the columns of $\mathscr{H}_L(u)$ and $\mathscr{H}_L(y)$ into past and future:

$$\mathscr{H}_L(u) = \begin{bmatrix} U_p \\ U_f \end{bmatrix}, \quad \mathscr{H}_L(y) = \begin{bmatrix} Y_p \\ Y_f \end{bmatrix}$$

with $U_p \in \mathbb{R}^{mn_{max} \times mc}$, $U_f \in \mathbb{R}^{m(L-n_{max}) \times mc}$, $Y_p \in \mathbb{R}^{pn_{max} \times pc}$, and $Y_f \in \mathbb{R}^{p(L-n_{max}) \times pc}$ where $c = T-L+1$. In order to compute the output that a system would generate if a zero input is applied at time $t = T+1$ one needs to determine what is the linear combination of the block columns of $\mathscr{H}_L(w)$, represented by the matrix G, so that the following equations are satisfied:

$$U_p G = u|_{[T-n_{max}+1,T]}, \qquad Y_p G = y|_{[T-n_{max}+1,T]},$$
$$U_f G = 0.$$

(3.4)

If such matrix G is found, it explains what the output is when the input is switched to zero at time $t = T+1$. Such output is given by $Y_f G$. Sufficient conditions ensuring solvability of (3.4) and the uniqueness of the free response computed as $Y_f G$ are given in [5]. When we are dealing with real data, due to the presence of noise, a solution to (3.4) always exists when number of columns of $\begin{bmatrix} U_p \\ U_f \\ Y_p \end{bmatrix}$ is much lager than the number of its rows, i.e. $T - L + 1 \gg L + n_{max}$. The solvability not being an issue, uniqueness of solutions is. Different G's that satisfy (3.4) may yield to computation of different Y_0's that will lead to different errors in estimates. However, as our objective is to study the adequacy of behavioral methods to SOC estimation, we pick a sparse solution and leave for future work the study of other solutions of (3.4). An algorithm for the computation of the free response is presented in the next section.

3.2 A behavioral algorithm

Given a T samples behavior $w = (u,y) \in \mathcal{B}_{[1,T]}$ obtained by recording input current and terminal voltage, Algorithm 1 computes the Δ samples long free response of a system with the initial state being the state at time $t = T+1$. The

---

Input : $u \in \mathbb{R}^T, y \in \mathbb{R}^T, \Delta \in \mathbb{N}, n_{max} \in \mathbb{N}, l \in \mathbb{N}$

Output: $Y_0 \in \mathbb{R}^{\Delta \times 1}$

Initialization;

$k := 0;$

$f_u^{(0)} := \begin{bmatrix} u|_{[T-n_{max}+1,T]} \\ 0 \end{bmatrix};$

$f_{y,p}^{(0)} := y|_{[T-n_{max}+1,T]};$

Main loop;

while $kl < \Delta$ **do**

   compute least squares solution of

   $\begin{bmatrix} U_p \\ U_f \\ Y_p \end{bmatrix} G^{(k)} = \begin{bmatrix} f_u^{(k)} \\ f_{y,p}^{(k)} \end{bmatrix};$

   $Y_0^{(k)} := f_{y,f}^{(k)} := Y_f G^{(k)};$

   $f_y^{(k)} := \begin{bmatrix} f_{y,p}^{(k)} \\ f_{y,f}^{(k)} \end{bmatrix};$

   $f_u^{(k+1)} := \begin{bmatrix} \sigma^\Delta f_u^{(k)} \\ 0_{l \times 1} \end{bmatrix}, f_{y,p}^{(k+1)} :=$

   $\sigma^\Delta f_y^{(k)};$

   $k := k + 1;$

end

Free response;

$Y_0 := \begin{bmatrix} Y_0^{(0)} \\ \vdots \\ Y_0^{(k-1)} \end{bmatrix};$

Algorithm 1: Algorithm for the computation of a Δ samples long free response.

---

algorithm computes l samples of free response in each iteration with the assumption that the data is coming from a single input-single output LTI system which has a minimal state space representation with at most n_{max} states.

We do not prove correctness of Algorithm 1 since the proof follows the same lines of the proof of Algorithm 8.9 in [5]. This reference should also be consulted for further information about the computation of free responses and corresponding algorithms. We also note that in [6], a more general version of the Algorithm 1 is presented.

4 Simulation Results

In this section, we illustrate the performance of our V_{oc} estimation algorithm by presenting various test results and comparisons with the algorithm introduced in [13]. We implemented Algorithm 1 where T is now the window length. We

---

shift this window at each sampling instant when we get new current and voltage data. We use two sources for battery data: a real electrochemical battery and the Dualfoil battery simulator [8] with the default parameters given for a lithium ion battery. We note that the advantage of using Dualfoil is that given a current profile, it provides the V_{oc} of the battery at every time instant.
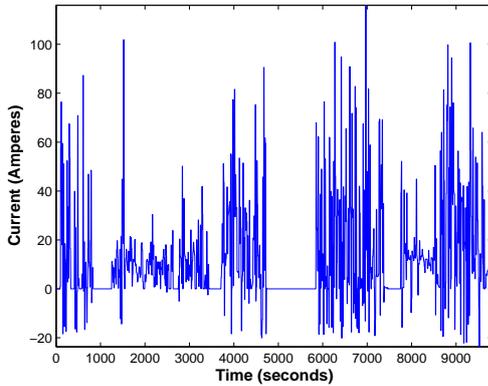
Figure 1: Current profile during the normal operation of an electric golf cart.

The first test case is generated via Dualfoil by using the realistic current trajectory depicted in Figure 1. This current profile was obtained from an electric golf-cart. We set the window length T to 450, Δ to 80 and n_{max} to 8. The top of Figure 2 illustrates the V_{oc} estimation of Dualfoil, of our algorithm and of the algorithm in [13], while the bottom of Figure 2 shows the percent relative V_{oc} error defined as

$$\left| \frac{V_{oc,estimated} - V_{oc,simulated}}{V_{oc,simulated}} \right| \times 100 \quad (4.1)$$

for both algorithms. Clearly, behavioral algorithm shows less variation than the algorithm in [13] and consequently leads to smaller errors except the intervals where the average energy extracted from the battery is small. In these intervals, it would take less time for terminal voltage to settle down to current V_{oc} value. Hence, computing fewer samples of the free response would be enough to estimate the V_{oc} and that would lead to better results. We will investigate the effect of adjusting the number of samples of free response computed based on the current profile observed as future work.

Figure 4, shows the performance of our algorithm in another test case generated by Dualfoil where the power delivered is kept constant at $50W/m^2$ during the discharge of the battery. The corresponding current profile can be seen in Figure 3. Here, we set the window length T to 8500, Δ to 50 and n_{max} to 2. The degradation in the performance is due to the fact that the current is not changing as much as it was in the previous case. When the current changes slowly, the least squares problem solved in Algorithm 1 becomes more sensitive to discrepancies in the data resulting from nonlinearities of the battery.
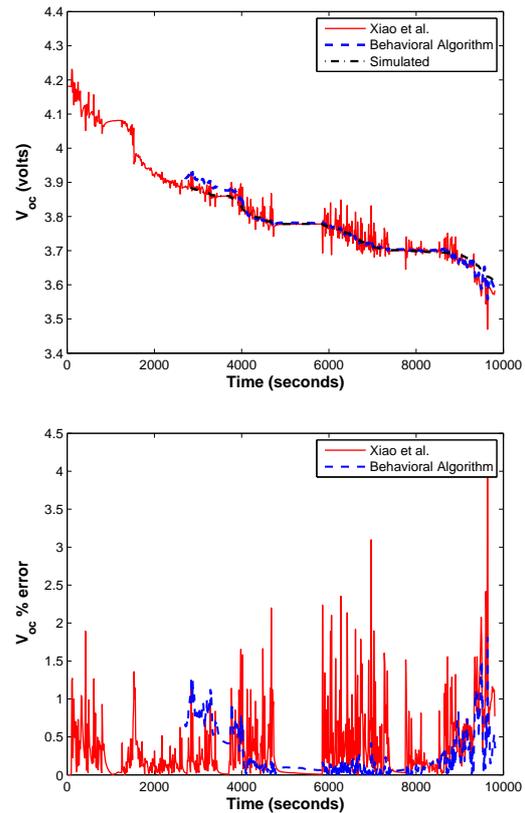
Figure 2: Comparison of the algorithms and simulated data.

Lastly, in Figure 5 we present the performance of our algorithm on real current and real voltage data (see Figure 6) collected from a electrical golf-cart in a typical driving cycle. In this case, unfortunately, we do not have real V_{oc} data to analyze the accuracy of the estimates. Instead, we only compare the estimates of the algorithm in [13] with ours and show that both yield similar estimates with ours having less fluctuations.

5 Discussion

In this paper, we proposed a behavioral algorithm to compute the open circuit voltage of a battery based on terminal voltage and terminal current measurements. The proposed algorithm does not postulate a model for the battery neither it tries to identify a model from data. As we illustrated with simulations, whenever the current profile is sufficiently diverse (changes fast), the results are comparable with the results from [13] and show less variability in the V_{oc} estimate. Due to the nonlinearities of the batteries, the performance of the algorithm depends on the choice of parameters T and Δ that must be adjusted in order to get good estimates. The results are promising enough to suggest further study of behavioral algorithms in the context of state of charge estimation. In particular, more sophisticated ap-
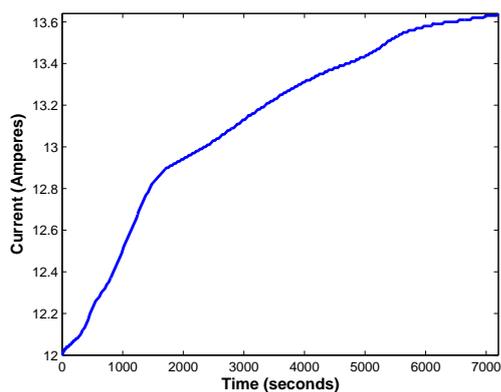
Figure 3: Current profile obtained when constant power $50W/m^2$ is delivered
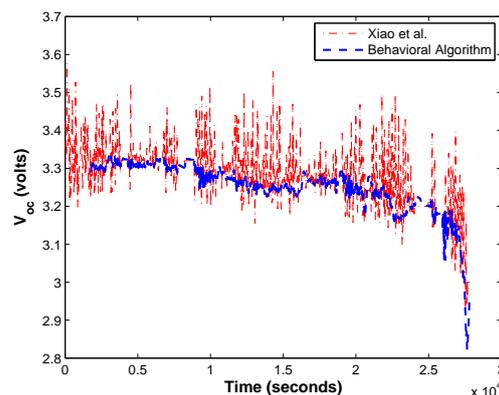
Figure 5: Comparison of the algorithms with real data from an electrical golf-cart.
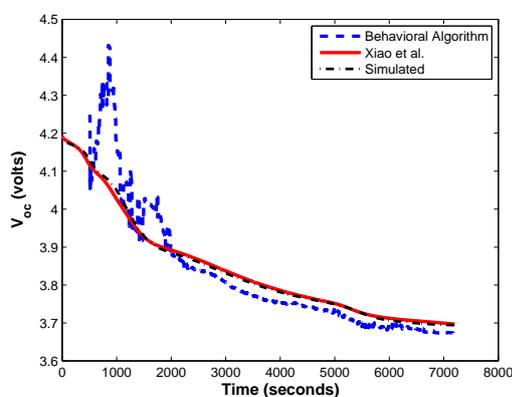
Figure 4: Comparison of the algorithms and simulated data when constant power of $50W/m^2$ is delivered
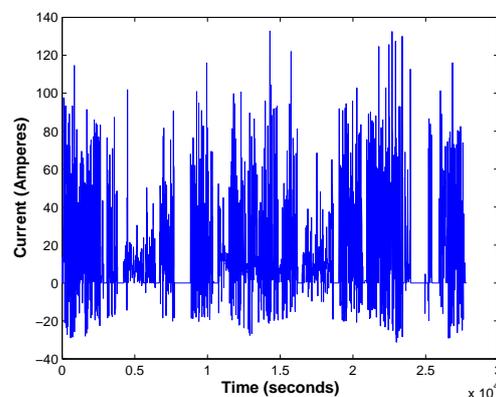
Figure 6: Real current profile obtained from an electrical golf-cart.

proximate solutions to equation (3.4) need to be investigated. Future work will also address the implementability of the algorithm on embedded hardware.

References

[1] *Genesis Application Manual*. H. E. P. Inc., 5th edition, 2000.

[2] A. Affanni, A. Bellini, C. Concari, G. Franceschini, E. Lorenzani, and C. Tassoni. Ev battery state of charge: neural network based estimation. In *Electric Machines and Drives Conference, 2003. IEMDC'03. IEEE International*, volume 2, pages 684 – 688 vol.2, june 2003.

[3] J. Chiasson and B. Vairamohan. Estimating the state of charge of a battery. *Control Systems Technology, IEEE Transactions on*, 13(3):465 – 470, may 2005.

[4] Seongjun Lee, Jonghoon Kim, Jaemoon Lee, and B.H. Cho. State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. *Journal of Power Sources*, 185(2):1367 – 1373, 2008.

[5] I. Markovsky, J. C. Willems, B. De Moor, and S. Van Huffel. *Exact and Approximate Modeling of Linear Systems: A Behavioral Approach*. Number 11 in Monographs on Mathematical Modeling and Computation. SIAM, March 2006.

[6] Ivan Markovsky and Paolo Rapisarda. Data-driven simulation and control. *International Journal of Control*, 81(12):1946–1959, December 2008. simulation, data-driven control, output matching, linear quadratic tracking, system identification.

[7] P. L. Moss, G. Au, E. J. Plichta, and J. P. Zheng. An electrical circuit for modeling the dynamic response of li-ion polymer batteries. *Journal of The Electrochemical Society*, 155(12):A986–A994, 2008.

[8] John Newman. Fortran programs for the simulation of electrochemical systems. http://www.cchem.berkeley.edu/jsngrp/fortran.html

[9] Shuo Pang, J. Farrell, Jie Du, and M. Barth. Battery state-of-charge estimation. In *American Control Conference, 2001. Proceedings of the 2001*, volume 2, pages 1644 –1649 vol.2, 2001.

[10] Jinchun Peng, Yaobin Chen, and R. Eberhart. Battery pack state of charge estimator design using computational intelligence approaches. In *Battery Conference on Applications and Advances, 2000. The Fifteenth Annual*, pages 173 –177, 2000.

[11] Jan W. Polderman and Jan C. Willems. *Introduction to mathematical systems theory : a behavioral approach*. Texts in applied mathematics, 26. Springer, 1998.

[12] Mark Verbrugge and Edward Tate. Adaptive state of charge algorithm for nickel metal hydride batteries including hysteresis phenomena. *Journal of Power Sources*, 126(1-2):236 – 249, 2004.

[13] Bingjun Xiao, Yiyu Shi, and Lei He. A universal state-of-charge algorithm for batteries. In *DAC*, pages 687–692, 2010.

Authors

Ayca Balkan received her B.S. degree in Electrical and Electronics Engineering from Middle East Technical University in 2010 and M.S. degree in Electrical Engineering from UCLA in 2011. Her research interests include behavioral systems theory, system identification, smart buildings and robust cyber-physical systems.

Min Gao received his B.E. Degree in power engineering and automation from Nanjing University of Technology, Nanjing, China. He is now pursuing M.S. Degree in electrical engineering from University of California, Los Angeles. His research interests include electric vehicle modeling and smart grid transmission, distribution and control strategy.

Paulo Tabuada is an associate professor of electrical engineering at UCLA and vice-chair for graduate affairs. His research has been recognized by several awards including the 2009 Donald P. Eckman award from the American Automatic Control Council and the 2011 George S. Axelby award from the IEEE Control Systems Society. His research interests include modeling, analysis, and control of cyber-physical systems, geometric control theory, and mathematical systems theory.

Dr. Lei He is a professor at UCLA Henry Samueli School of Engineering,the founding director of clean energy research center, Los Angeles, and aguest chair professor at Fudan University, China. He was a faculty memberat University of Wisconsin, Madison, held visiting or consulting positionswith Cisco, Cadence Design Systems, Hewlett-Package, Intel, and Synopsys, andhas been a technical advisory board member for Apache Design Solutions(acquired by Ansoft in 2011), Rio Design Automation (acquired by MagmaDesign Automation in 2006), Empyrean Software, Pride Power Systems, andDB New Energy. His research interests include information technology and energy systems.