Applied Energy 92 (2012) 694–704

Contents lists available at SciVerse ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

A multiscale framework with extended Kalman filter for lithium-ion battery SOC and capacity estimation

Chao Hu [a], Byeng D. Youn [b,*], Jaesik Chung [c]

[a] Department of Mechanical Engineering, University of Maryland at College Park, College Park, MD 20742, USA
[b] School of Mechanical and Aerospace Engineering, Seoul National University, Seoul, Republic of Korea
[c] PCTEST Engineering Laboratory, Columbia, MD 21045, USA

ARTICLE INFO

Article history:
Received 31 December 2010
Received in revised form 27 July 2011
Accepted 1 August 2011
Available online 15 September 2011

Keywords:
Multiscale framework
Time scale separation
State of charge (SOC)
State of health (SOH)
Lithium-ion battery

ABSTRACT

State-of-charge (SOC) and capacity estimation plays an essential role in many battery-powered applications, such as electric vehicle (EV) and hybrid electric vehicle (HEV). However, commonly used joint/dual extended Kalman filter (EKF) suffers from the lack of accuracy in the capacity estimation since (i) the cell voltage is the only measurable data for the SOC and capacity estimation and updates and (ii) the capacity is very weakly linked to the cell voltage. The lack of accuracy in the capacity estimation may further reduce the accuracy in the SOC estimation due to the strong dependency of the SOC on the capacity. Furthermore, although the capacity is a slowly time-varying quantity that indicates cell state-of-health (SOH), the capacity estimation is generally performed on the same time-scale as the quickly time-varying SOC, resulting in high computational complexity. To resolve these difficulties, this paper proposes a multiscale framework with EKF for SOC and capacity estimation. The proposed framework comprises two ideas: (i) a multiscale framework to estimate SOC and capacity that exhibit time-scale separation and (ii) a state projection scheme for accurate and stable capacity estimation. Simulation results with synthetic data based on a valid cell dynamic model suggest that the proposed framework, as a hybrid of coulomb counting and adaptive filtering techniques, achieves higher accuracy and efficiency than joint/dual EKF. Results of the cycle test on Lithium-ion prismatic cells further verify the effectiveness of our framework.

1. Introduction

As a battery cell ages, the cell capacity and resistance directly limit the pack performance through capacity and power fade, respectively [1]. These two degradation parameters are often used to quantify the cell state of health (SOH). Thus, it is important to accurately estimate these parameters to monitoring the present battery SOH and to predict the remaining useful life (RUL). Recent literature reports various approaches to estimate the SOH with a focus on the capacity estimation. Joint/dual extended Kalman filter (EKF) [1] and unscented Kalman filter [2] with an enhanced self-correcting model were proposed to simultaneously estimate the SOC, capacity and resistance. To improve the performance of joint/dual estimation, adaptive measurement noise models of the Kalman filter were recently developed to separate the sequence of SOC and capacity estimation [3]. A physics-based single particle model was used to simulate the life cycling data of Li-ion cells and to study the physics of capacity fade [4,5]. More recently, new techniques for SOH estimation were developed based on a coulomb counting technique with dynamic re-calibration of the cell capacity [6] and the approximate entropies of cell terminal voltage and current [7]. In the PHM society, a Bayesian framework combining the relevance vector machine (RVM) and particle filter was proposed for prognostics (i.e., RUL prediction) of Li-ion battery cells [8]. More recently, the particle filter with an empirical circuit model was used to predict the remaining useful lives for individual discharge cycles as well as for cycle life [9].

Among these techniques, the joint/dual estimation technique is capable of real-time SOC and capacity estimation with noisy voltage and current measurements. Although it provides highly accurate SOC estimation, it suffers from the lack of accuracy in the capacity estimation since (i) the cell voltage is the only measurable data for the measurement-updates in the SOC and capacity estimation and (ii) the capacity is very weakly linked to the cell voltage. Due to the strong correlation between the SOC and capacity, inaccurate capacity estimation may further lead to inaccurate SOC estimation and vice versa. Furthermore, although the capacity is a slowly time-varying quantity that indicates cell state-of-health (SOH), the capacity estimation is generally performed on the same time-scale as the quickly time-varying

* Corresponding author. Tel.: +82 2 880 1919; fax: +82 2 880 8302.
E-mail address: bdyoun@snu.ac.kr (B.D. Youn).

0306-2619/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.apenergy.2011.08.002

SOC, resulting in high computational complexity. To resolve these difficulties, this paper proposes a multiscale framework with EKF for SOC and capacity estimation. The proposed framework comprises two ideas: (i) a multiscale framework to estimate SOC and capacity that exhibit time-scale separation and (ii) a state projection scheme for accurate and stable capacity estimation.

We have successfully implemented the proposed framework on a Li-ion polymer battery (LiPB) cell with simulation and on a Li-ion prismatic battery cell with testing. Since the proposed framework can employ any cell dynamic model that appropriately represents cell dynamics, this framework is expected to work for other cell chemistries and physical configurations as well where the only difference lies in the cell dynamic model. Besides, we can also apply the proposed framework to a standby battery, which temporarily supplies electrical power (non-zero current) in the event of a power outage, by limiting the effective execution period to a time when a power outage occurs. This strategy enables the updating of SOC and capacity when the standby battery is in use. When the standby battery is not in use, the battery is either under the condition of float charge or no charge. Under the condition of float charge, a standby battery is continuously connected to a constant-voltage supply that maintains the battery in a fully charged condition. Under the condition of no charge, a standby battery suffers from reversible and irreversible capacity losses due to undesirable chemical actions during the self discharge. The self discharge rate highly depends on both the cell chemistry and the ambient temperature. A lithium-ion (Li-ion) battery cell typically has a much lower self discharge rate (2–3% per month at the room temperature) compared to those of nickel cadmium (15–20% per month at the room temperature) and nickel metal hydride (30% per month at the room temperature) battery cells. Thus, the capacity loss of a Li-ion battery cell due to the self discharge is very small at the room temperature and, more importantly, most of the capacity loss is reversible (i.e., the loss can be regenerated by charging the cell). The reversible capacity loss can be equivalently treated as the SOC reduction, which can be readily estimated based on the relationship between the open circuit voltage (OCV) and SOC. Furthermore, the proposal framework is also applicable under the condition of energy regeneration in an EV/HEV. Under this condition, the regenerative braking system employs the traction motor as a generator which transfers a portion of the vehicle's kinetic energy to electric energy in order to recharge the battery pack. Since the battery pack undergoes charging cycles (positive charge currents) during the regeneration, the cell SOC and capacity can be readily estimated with the multiscale framework based on the cell current and voltage measurements, just like the case with urban dynamometer drive schedule (UDDS) discharging cycles to be detailed in the simulation and experimental studies. It is also noted that the multiscale framework is generic since it can be used to achieve highly-confident health prognostics for any engineered system with multiple time-scales.

This paper is organized as follows. Section 2 describes the discrete-time state-space model of an engineered system with multiple time-scales. Section 3 reviews the numerical formulation and implementation of the dual EKF method. Section 4 presents the proposed multiscale framework with EKF and introduces the state projection scheme. In the these sections, we intend to present a generic description in the sense that it is applicable to any engineered system with multiple time-scales as well as provide a clear demonstration by mapping important terms to those in the battery system. The proposed ideas are applied to the battery system to estimate SOC and capacity in Section 6. Section 6 contains simulation and experimental results of this application. The paper is concluded in Section 7.

2. System description

To make the discussion more concrete, we will use discrete-time state-space models with multiple time-scales. Without loss of generality, we assume the system has two time-scales: the macro and micro time-scales. System quantities on the macro time-scale tend to vary slowly over time while system quantities on the micro time-scale exhibit fast variation over time. The former are referred to as the model parameters of the system while the latter are called the states of the system. We then begin by defining the nonlinear state-space model considered in this work as

$$\begin{aligned} &\text{Transition}: \quad \mathbf{x}_{k,l+1} = \mathbf{F}(\mathbf{x}_{k,l}, \mathbf{u}_{k,l}, \boldsymbol{\theta}_k) + \mathbf{w}_{k,l}, \boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \mathbf{r}_k, \\ &\text{Measurement}: \quad \mathbf{y}_{k,l} = \mathbf{G}(\mathbf{x}_{k,l}, \mathbf{u}_{k,l}, \boldsymbol{\theta}_k) + \mathbf{v}_{k,l} \end{aligned} \tag{1}$$

where $\mathbf{x}_{k,l}$ is the vector of system states at the time $t_{k,l} = t_{k,0} + l \cdot T$, for $1 \leqslant l \leqslant L$, with T being a fixed time step between two adjacent measurement points, and k and l being the indices of macro and micro time-scales, respectively; $\boldsymbol{\theta}_k$ is the vector of system model parameters at the time $t_{k,0}$; $\mathbf{u}_{k,l}$ is the vector of observed exogenous inputs; $\mathbf{y}_{k,l}$ is the vector of system observations (or measurements); $\mathbf{w}_{k,l}$ and $\mathbf{r}_k$ are the vectors of process noise for states and model parameters, respectively; $\mathbf{v}_{k,l}$ is the vectors of measurement noise; $\mathbf{F}(\cdot,\cdot,\cdot)$ and $\mathbf{G}(\cdot,\cdot,\cdot)$ are the state transition and measurement functions, respectively. Note that L represents the level of time-scale separation and that $\mathbf{x}_{k,0} = \mathbf{x}_{k-1,L}$. With the system defined, we aim at estimating both the system states $\mathbf{x}$ and model parameters $\boldsymbol{\theta}$ from the noisy observations $\mathbf{y}$.

Let us take the battery system as an example. In the battery system, the system state x refers to the SOC, which changes very rapidly and may transverse the entire range 100–0% within minutes. Here we use an italic, non-bold letter x to indicate that the system state in the battery system is a scalar rather than a vector, and the same notational rule applies to all other functions and variables. The system model parameter θ represents the cell capacity which tends to vary very slowly and typically decreases 1.0% or less in a month with regular use. The state transition equation $F(\cdot,\cdot,\cdot)$ models the variation of SOC over time while the cell dynamic model $G(\cdot,\cdot,\cdot)$ relates the measured cell terminal voltage y with the unmeasured state (SOC) and model parameter (capacity) and the measured exogenous input u. Given the system's state-space model in Eq. (1) and knowledge of the measured input/output signals (cell current/cell terminal voltage), we are interested in estimating the unmeasured state (SOC) and model parameter (capacity) in real-time and in a dynamic environment. The subsequent sections are dedicated to describing an existing technique and our proposed technique for doing so.

3. Review of dual extended Kalman filter method

The dual extended Kalman filter (EKF) method is a commonly used technique to simultaneously estimate the states and model parameters [12]. The essence of the dual EKF method is to combine the state and weight EKFs with the state EKF estimating the system states and the weight EKF estimating the system model parameters. In the algorithm, two EKFs are run concurrently and, at every time step when observations are available, the state EKF estimates the states using the current model parameter estimates from the weight EKF while the weight EKF estimates the model parameters using the current state estimates from the state EKF. This section gives a brief review of the dual EKF method. Section 3.1 presents the numerical formulation of the dual EKF method and the numerical implementation of the recursive derivative computation is described in Section 3.2.

3.1. Numerical formulation: dual estimation

The algorithm of the dual EKF for the system described in Eq. (1) is summarized in Table 1. Since the dual EKF does not take into account the time-scale separation, $\boldsymbol{\theta}_k$ is estimated on the micro time-scale. To reflect this, we use the notations $\boldsymbol{\theta}_{k,l}$ and $\mathbf{r}_{k,l}$ to replace $\boldsymbol{\theta}_k$ and $\mathbf{r}_k$, respectively. Also note that, to be consistent with the system description in Eq. (1), we use two time indices k and l to present the dual EKF algorithm and this presentation is equivalent to a simpler version in [13] with only one time index l.

The algorithm is initialized by setting the model parameters $\boldsymbol{\theta}$ and states $\mathbf{x}$ to the best guesses based on the prior information. The covariance matrices $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ and $\boldsymbol{\Sigma}_{\mathbf{x}}$ of estimation errors are also initialized based on the prior information. At each measurement time step, the time- and measurement-updates are performed in the following two EKFs: weight EKF and state EKF.

3.1.1. Weight EKF (parameter estimation)

The weight EKF first executes the time-update, where prior parameter estimates $\hat{\boldsymbol{\theta}}_{k,l}^-$ and their error covariance $\boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}}^-$ are computed with Eq. (3). Due to the addition of unpredictable process noise $\mathbf{r}_{k,l}$ in Eq. (1), the uncertainties $\boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}}^-$ in the parameter estimates always increase. Following the time-update step, the estimated measurements are then computed by

$$\hat{\mathbf{y}}_{k,l} = \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k,l}^-) \tag{10}$$

The above predicted measurements are compared with the real measurements $\mathbf{y}_{k,l}$ to obtain prediction errors which state the novelty or the new information that the measurements $\mathbf{y}_{k,l}$ brought to the filter relative to the parameters $\boldsymbol{\theta}_{k,l}$. The prediction errors are used to adapt the current parameter estimates and obtain posteriori parameter estimates $\hat{\boldsymbol{\theta}}_{k,l}$ using Eq. (8). Due to the addition of one set of measurements, the error uncertainties are reduced as can be seen in Eq. (8). This process is referred to as the measurement-update.

In the battery system, the measured terminal quantities are the cell terminal voltage y and current u. Since the capacity affects the SOC transition which further affects the cell terminal voltage, the cell terminal voltage measurement y can be used to adapt the capacity by following the steps detailed above.

3.1.2. State EKF (state estimation)

The state EKF essentially follows the same manner as the weight EKF. One difference lies in the fact that the time-update in the state EKF employs the state transition function $\mathbf{F}(\cdot,\cdot,\cdot)$ as shown in Eq. (4). Similar to the weight EKF, the measurement-update in the state EKF also uses the differences between the predicted measurements in Eq. (10) and the real measurements to adapt the states $\mathbf{x}_{k,l}$. As shown in Eq. (6), the posteriori state estimates are obtained by correcting the priori state estimates with the prediction errors multiplied by gain factors.

When applied to the battery system, the state EKF aims at estimating the SOC x based on the measured cell terminal voltage y and current u. Since the SOC directly affects the cell terminal voltage through the cell dynamic model $G(\cdot,\cdot,\cdot)$, the cell terminal voltage measurement, as the model output, can be used to back-estimate the SOC, as one model input, by following the steps detailed above.

3.2. Numerical Implementation: recursive derivative computation

The dual EKF method, which adapts the states and parameters using two concurrently running EKFs, has a recursive architecture associated with the computation of $\mathbf{C}_{k,l}^{\boldsymbol{\theta}}$ in the weight filter. The computation of $\mathbf{C}_{k,l}^{\boldsymbol{\theta}}$ involves a total derivative of the measurement function with respect to the parameters $\boldsymbol{\theta}$ as

Table 1
Algorithm of dual extended Kalman filter [13].

Initialization

$$\hat{\boldsymbol{\theta}}_{0,0} = E[\boldsymbol{\theta}_{0,0}], \boldsymbol{\Sigma}_{\boldsymbol{\theta}_{0,0}} = E\left[(\boldsymbol{\theta}_{0,0} - \hat{\boldsymbol{\theta}}_{0,0})(\boldsymbol{\theta}_{0,0} - \hat{\boldsymbol{\theta}}_{0,0})^{\mathrm{T}}\right],$$
$$\hat{\mathbf{x}}_{0,0} = E[\hat{\mathbf{x}}_{0,0}], \boldsymbol{\Sigma}_{\mathbf{x}_{0,0}} = E\left[(\mathbf{x}_{0,0} - \hat{\mathbf{x}}_{0,0})(\mathbf{x}_{0,0} - \hat{\mathbf{x}}_{0,0})^{\mathrm{T}}\right]. \tag{2}$$

For $k \in \{1,\ldots,\infty\}$, $l \in \{l,\ldots,L\}$, compute

Time-update equations for the weight filter

$$\hat{\boldsymbol{\theta}}_{k,l}^- = \hat{\boldsymbol{\theta}}_{k,l-1}, \quad \boldsymbol{\Sigma}_{\theta_{k,l}}^- = \boldsymbol{\Sigma}_{\theta_{k,l-1}} + \boldsymbol{\Sigma}_{\mathbf{r}_{k,l-1}}. \tag{3}$$

Time-update equations for the state filter

$$\hat{\mathbf{x}}_{k,l}^- = \mathbf{F}(\hat{\mathbf{x}}_{k,l-1}, \mathbf{u}_{k,l-1}, \hat{\boldsymbol{\theta}}_{k,l}^-),$$
$$\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^- = \mathbf{A}_{k,l-1}\boldsymbol{\Sigma}_{\mathbf{x}_{k,l-1}}\mathbf{A}_{\mathbf{x}_{k,l-1}}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{w}_{k,l-1}}. \tag{4}$$

Measurement-update equations for the state filter

$$\mathbf{K}_{k,l}^{\mathbf{x}} = \boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^-(\mathbf{C}_{k,l}^{\mathbf{x}})^{\mathrm{T}}[\mathbf{C}_{k,l}^{\mathbf{x}}\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^-(\mathbf{C}_{k,l}^{\mathbf{x}})^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{v}_{k,l}}]^{-1}. \tag{5}$$

$$\hat{\mathbf{x}}_{k,l} = \hat{\mathbf{x}}_{k,l}^- + \mathbf{K}_{k,l}^{\mathbf{x}}[\mathbf{y}_{k,l} - \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k,l}^-)],$$
$$\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}} = (\mathbf{I} - \mathbf{K}_{k,l}^{\mathbf{x}}\mathbf{C}_{k,l}^{\mathbf{x}})\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^-. \tag{6}$$

Measurement-update equations for the weight filter

$$\mathbf{K}_{k,l}^{\boldsymbol{\theta}} = \boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}}^-(\mathbf{C}_{k,l}^{\boldsymbol{\theta}})^{\mathrm{T}}[\mathbf{C}_{k,l}^{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}}^-(\mathbf{C}_{k,l}^{\boldsymbol{\theta}})^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{n}_{k,l}}]^{-1}. \tag{7}$$

$$\hat{\boldsymbol{\theta}}_{k,l} = \hat{\boldsymbol{\theta}}_{k,l}^- + \mathbf{K}_{k,l}^{\boldsymbol{\theta}}[\mathbf{y}_{k,l} - \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k,l}^-)],$$
$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}} = (\mathbf{I} - \mathbf{K}_{k,l}^{\boldsymbol{\theta}}\mathbf{C}_{k,l}^{\boldsymbol{\theta}})\boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}}^-. \tag{8}$$

where

$$\mathbf{A}_{k,l-1} = \left.\frac{\partial \mathbf{F}(\mathbf{x}, \mathbf{u}_{k,l-1}, \hat{\boldsymbol{\theta}}_{k,l}^-)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k,l-1}}, \quad \mathbf{C}_{k,l}^{\mathbf{x}} = \left.\frac{\partial \mathbf{G}(\mathbf{x}, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k,l}^-)}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k,l}^-}, \tag{9}$$
$$\mathbf{C}_{k,l}^{\boldsymbol{\theta}} = \left.\frac{d\mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \boldsymbol{\theta})}{d\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{k,l}^-}.$$

$$\mathbf{C}_{k,l}^{\boldsymbol{\theta}} = \left.\frac{d\mathbf{G}\left(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \boldsymbol{\theta}\right)}{d\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{k,l}^-}. \tag{11}$$

This computation requires a recursive routine similar to a real-time recursive learning [14]. Decomposing the total derivative into partial derivatives and propagating the states back in time results in the following recursive equations

$$\frac{d\mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} = \frac{\partial \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \boldsymbol{\theta})}{\partial \hat{\mathbf{x}}_{k,l}^-}\frac{d\hat{\mathbf{x}}_{k,l}^-}{d\boldsymbol{\theta}}, \tag{12}$$

$$\frac{d\hat{\mathbf{x}}_{k,l}^-}{d\boldsymbol{\theta}} = \frac{\partial \mathbf{F}(\hat{\mathbf{x}}_{k,l-1}, \mathbf{u}_{k,l-1}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} + \frac{\partial \mathbf{F}(\hat{\mathbf{x}}_{k,l-1}, \mathbf{u}_{k,l-1}, \boldsymbol{\theta})}{\partial \hat{\mathbf{x}}_{k,l-1}}\frac{d\hat{\mathbf{x}}_{k,l-1}}{d\boldsymbol{\theta}}, \tag{13}$$

$$\frac{d\hat{\mathbf{x}}_{k,l-1}}{d\boldsymbol{\theta}} = \frac{d\hat{\mathbf{x}}_{k,l-1}^-}{d\boldsymbol{\theta}} - \mathbf{K}_{k,l-1}^{\mathbf{x}}\frac{d\mathbf{G}(\hat{\mathbf{x}}_{k,l-1}^-, \mathbf{u}_{k,l-1}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} + \frac{\partial \mathbf{K}_{k,l-1}^{\mathbf{x}}}{\partial \boldsymbol{\theta}}[\mathbf{y}_{k,l-1} - \mathbf{G}(\hat{\mathbf{x}}_{k,l-1}^-, \mathbf{u}_{k,l-1}, \boldsymbol{\theta})]. \tag{14}$$

The last term in Eq. (14) can be set to zero with the assumption that $\mathbf{K}_{k,l}^{\mathbf{x}}$ is not dependant on $\boldsymbol{\theta}$. Indeed, since $\mathbf{K}_{k,l}^{\mathbf{x}}$ is often very weakly dependent on $\boldsymbol{\theta}$, the extra computational effort to consider this dependence is not worth the improvement in performance.

Therefore, we drop the last term in Eq. (14) in this study. Then the three total derivatives can be computed in a recursive manner with initial values set as zeros. It noted that the partial derivatives of the transition and measurement functions with respect to the states $\mathbf{x}$ and parameters $\boldsymbol{\theta}$ can be easily computed with the explicitly given function forms.

4. A multiscale framework with extended Kalman filter

As discussed in Section 3, the dual EKF method estimates both the states and parameters on the same time-scale. However, for systems that exhibit the time-scale separation, it is natural and desirable to adapt the slowly time-varying parameters on the macro time-scale while keeping the estimation of the fast time-varying states on the micro time-scale. This multiscale framework is expected to reduce the computational effort and provide more stable estimates of model parameters. This section is dedicated to the discussion of this framework and is organized in a similar manner as Section 3: Section 4.1 presents the numerical formulation of the multiscale framework with EKF and the numerical implementation of the recursive derivative computation in the multiscale framework is described in Section 4.2.

4.1. Numerical formulation: multiscale estimation

As opposed to the dual estimation, we intend to derive a multiscale estimation which allows for a time-scale separation in the state and parameter estimation. More specifically, we aim at estimating the slowly time-varying model parameters on the macro time-scale and, at the same time, intend to keep the estimation of fast time-varying states on the micro time-scale to utilize all the measurements. The algorithm of the multiscale framework for the system described in Eq. (1) is summarized in Table 2. Note that, in contrast to the dual EKF algorithms in Table 1, we only use the macro time-scale index k to present the macro EKF since the parameter estimation is performed only every macro time step.

The algorithm is initialized by setting the model parameters $\boldsymbol{\theta}$ and states $\mathbf{x}$ to the best guesses based on the prior information. The covariance matrices $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ and $\boldsymbol{\Sigma}_{\mathbf{x}}$ of estimation errors are also initialized based on the prior information. The main algorithm essentially consists of the so-called micro and macro EKFs running on the micro and macro time-scales, respectively. Note that, the micro time-scale here refers to the time-scale on which system states exhibit fast variation while the macro time-scale refers to the one on which system model parameters tend to vary slowly. For example, in the battery system, the SOC, as a system state, changes every second, which suggests the micro time-scale to be approximately one second. In contrast, the cell capacity, as a system model parameter, typically decreases 1.0% or less in a month with regular use, resulting in the macro time-scale being approximately 1 day or so. The time- and measurement-updates performed in the macro EKF and micro EKF are detailed as follows.

4.1.1. Macro EKF (parameter estimation)

At every macro time step, the macro EKF executes the time-update where prior parameter estimates $\hat{\boldsymbol{\theta}}_k^-$ and their error covariance $\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-$ are computed with Eq. (16). The addition of unpredictable process noise $\mathbf{r}_k$ increases the uncertainties $\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-$ in the parameter estimates. After the time-update step, the state projection is conducted to project the state estimates from the micro EKF through the macro time step, expressed as the state projection function $\mathbf{F}_{0\to L}(\cdot,\cdot,\cdot)$ in Eq. (17). We note that $\mathbf{F}_{0\to L}(\cdot,\cdot,\cdot)$ can be expressed as a nested form of the state transition function $\mathbf{F}(\cdot,\cdot,\cdot)$ in Eq. (1) and that the computational effort involved in computing $\mathbf{F}_{0\to L}(\cdot,\cdot,\cdot)$ is negligible compared to the time- and measurement-updates conducted in L micro time steps. In the measurement-update step, the macro EKF computes the difference between the projected states and the estimated states from the micro EKF and uses the difference to obtain posterior parameter estimates, which is detailed in Eq. (19).

Compared with the weight EKF, the macro EKF possesses two distinctive characteristics: (i) the time- and measurement-updates are performed over the macro time-scale ($L{\cdot}T$) instead of the micro time-scale (T), leading to the possibility to greatly reduce the computational complexity; and (ii) the macro EKF uses the state estimates from the micro EKF for the measurement-update, enabled by the state projection in Eq. (17), and the resulting parameter estimation becomes decoupled with the state estimation where the real measurements in Eq. (1) are used for the measurement-update. The first characteristic could magnify the effect of the parameters on the states, i.e., that the parameters could affect

Table 2
Algorithm of a multiscale framework with extended Kalman filter.

Initialization

$$\begin{aligned}&\boldsymbol{\theta}_0 = E[\boldsymbol{\theta}_0], \quad \boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k,l}} = E[(\boldsymbol{\theta}_0 - \hat{\boldsymbol{\theta}}_0)(\boldsymbol{\theta}_0 - \hat{\boldsymbol{\theta}}_0)^{\mathrm{T}}],\\ &\hat{\mathbf{x}}_{0,0} = E[\mathbf{x}_{0,0}], \quad \boldsymbol{\Sigma}_{\mathbf{x}_{k,l}} = E[(\mathbf{x}_{0,0} - \hat{\mathbf{x}}_{0,0})(\mathbf{x}_{0,0} - \hat{\mathbf{x}}_{0,0})^{\mathrm{T}}].\end{aligned} \tag{15}$$

For $k \in \{1,\ldots,\infty\}$, compute

Time-update equations for the macro EKF

$$\hat{\boldsymbol{\theta}}_k^- = \hat{\boldsymbol{\theta}}_{k-1}, \quad \boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^- = \boldsymbol{\Sigma}_{\boldsymbol{\theta}_{k-1}} + \boldsymbol{\Sigma}_{\mathbf{r}_{k-1}}. \tag{16}$$

State projection equation for the macro EKF

$$\tilde{\mathbf{x}}_{k-1,L} = \mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \hat{\boldsymbol{\theta}}_k^-). \tag{17}$$

Measurement-update equations for the macro EKF

$$\mathbf{K}_k^{\boldsymbol{\theta}} = \boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-(\mathbf{C}_k^{\boldsymbol{\theta}})^{\mathrm{T}}[\mathbf{C}_k^{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-(\mathbf{C}_k^{\boldsymbol{\theta}})^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{n}_k}]^{-1}. \tag{18}$$

$$\begin{aligned}&\hat{\boldsymbol{\theta}}_k = \hat{\boldsymbol{\theta}}_k^- + \mathbf{K}_k^{\boldsymbol{\theta}}[\hat{\mathbf{x}}_{k-1,L} - \tilde{\mathbf{x}}_{k-1,L}],\\ &\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k} = (\mathbf{I} - \mathbf{K}_k^{\boldsymbol{\theta}}\mathbf{C}_k^{\boldsymbol{\theta}})\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-.\end{aligned} \tag{19}$$

For $l \in \{1,\ldots,L\}$, compute

Time-update equations for the micro EKF

$$\begin{aligned}&\hat{\mathbf{x}}_{k,l}^- = \mathbf{F}(\hat{\mathbf{x}}_{k,l-1}, \mathbf{u}_{k,l-1}, \hat{\boldsymbol{\theta}}_{k-1}),\\ &\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^- = \mathbf{A}_{k,l-1}\boldsymbol{\Sigma}_{\mathbf{x}_{k,l-1}}\mathbf{A}_{k,l-1}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{w}_{k,l-1}}.\end{aligned} \tag{20}$$

Measurement-update equations for the micro EKF

$$\mathbf{K}_k^{\mathbf{x}} = \boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-(\mathbf{C}_k^{\boldsymbol{\theta}})^{\mathrm{T}}[\mathbf{C}_k^{\boldsymbol{\theta}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k}^-(\mathbf{C}_k^{\boldsymbol{\theta}})^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{n}_k}]^{-1}. \tag{21}$$

$$\begin{aligned}&\hat{\mathbf{x}}_{k,l} = \hat{\mathbf{x}}_{k,l}^- + \mathbf{K}_{k,l}^{\mathbf{x}}[\mathbf{y}_{k,l} - \mathbf{G}(\hat{\mathbf{x}}_{k,l}^-, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k-1})],\\ &\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}} = (\mathbf{I} - \mathbf{K}_{k,l}^{\mathbf{x}}\mathbf{C}_{k,l}^{\mathbf{x}})\boldsymbol{\Sigma}_{\mathbf{x}_{k,l}}^-.\end{aligned} \tag{22}$$

where

$$\begin{aligned}&\mathbf{A}_{k,l-1} = \left.\frac{\partial \mathbf{F}(\mathbf{x}, \mathbf{u}_{k,l-1}, \hat{\boldsymbol{\theta}}_{k-1})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k,l-1}},\\ &\mathbf{C}_{k,l}^{\mathbf{x}} = \left.\frac{\partial \mathbf{G}(\mathbf{x}, \mathbf{u}_{k,l}, \hat{\boldsymbol{\theta}}_{k-1})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k,l}^-},\\ &\mathbf{C}_k^{\boldsymbol{\theta}} = \left.\frac{d\mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \boldsymbol{\theta})}{d\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_k^-}.\end{aligned} \tag{23}$$

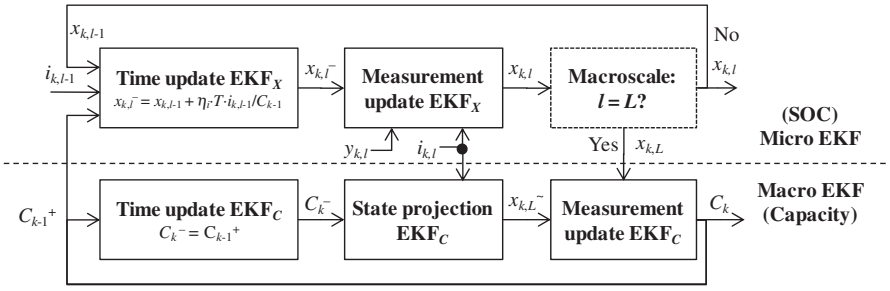

Fig. 1. Flowchart of a multiscale framework with EKF for battery SOC and capacity estimation.

the states projected on the macro time-scale ($L{\cdot}T$) more significantly than those projected on the micro time-scale (T). The second characteristic helps distinguish the effects of the two unknowns (states and parameters) on the only measurements. In the subsequent section, these characteristics will be further explained and verified when we apply the algorithm to the battery system.

4.1.2. Micro EKF (state estimation)

The micro EKF bears a strong resemblance to the state EKF in the dual EKF. The only difference is that, for the state transition, the micro EKF uses the capacity estimate from the previous macro time step (see Eq. (20)) while the state EKF employs that the previous micro time step (see Eq. (4)). It is important to note that, at the start of every macro time step, i.e., at the time $t_{k-1,0}$, the micro EKF sends the state estimate to the macro EKF which then projects it through the macro time step according to the state projection equation in Eq. (17). Upon the completion of the state projection at the end of every macro time step, i.e., at the time $t_{k-1,L}$, the micro EKF sends another state estimate to the macro EKF which then compares it with the projected estimate and uses the difference to adapt the parameter estimate in the measurement-update step detailed in Eq. (19).

4.2. Numerical implementation: recursive derivative computation

In the multiscale framework, the computation of $\mathbf{C}_k^{\theta}$ in the macro EKF involves a total derivative of the state projection function with respect to the parameters $\boldsymbol{\theta}$ as

$$\mathbf{C}_k^{\theta} = \left.\frac{d\mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \boldsymbol{\theta})}{d\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_k^-}. \tag{24}$$

Similar to the total derivative in Eq. (11), this computation also requires a recursive routine. Decomposing the total derivative into partial derivatives, we then obtain the following equation

$$\begin{aligned}\frac{d\mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \boldsymbol{\theta})}{d\boldsymbol{\theta}} &= \frac{\partial \mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \\ &+ \frac{\partial \mathbf{F}_{0\to L}(\hat{\mathbf{x}}_{k-1,0}, \mathbf{u}_{k-1,0:L-1}, \boldsymbol{\theta})}{\partial \hat{\mathbf{x}}_{k-1,0}} \\ &\times \frac{d\hat{\mathbf{x}}_{k-1,0}}{d\boldsymbol{\theta}}.\end{aligned} \tag{25}$$

The total derivative in the last term can be obtained by using the recursive equations Eqs. (12)–(14). The partial derivatives of the state projection function with respect to the states $\mathbf{x}$ and parameters $\boldsymbol{\theta}$ can be easily computed with the explicitly given function forms.

5. Application to Li-ion battery system

In this section, we use the proposed framework to estimate the SOC and capacity in a Li-ion battery system. When applied to the battery system, the multiscale framework can be treated as a hybrid of coulomb counting and adaptive filtering techniques and comprises two new ideas: (i) a multiscale framework to estimate SOC and capacity that exhibit time-scale separation and (ii) a state projection scheme for accurate and stable capacity estimation. Section 5.1 presents the discrete-time cell dynamic model used in this study. Section 5.2 presents the multiscale estimation of SOC and capacity in the battery system.

5.1. Discrete-time cell dynamic model

In order to execute the time-update in the micro and macro EKFs, we need a state transition model that propagate the SOC forward in time. In order to execute the measurement-update in the micro-EKF, we need a "discrete-time cell dynamic model" that relates the SOC to the cell voltage. Here we employ the enhanced self-correcting (ESC) model which considers the effects of OCV, internal resistance, voltage time constants and hysteresis [1]. The effects of voltage time constants and hysteresis in the ESC model can be expressed as [1]

$$\begin{aligned}\begin{bmatrix} f_{k,l+1} \\ h_{k,l+1} \end{bmatrix} &= \begin{bmatrix} \mathrm{diag}(\boldsymbol{\alpha}) & 0 \\ 0 & \varphi(i_{k,l+1}) \end{bmatrix} \begin{bmatrix} f_{k,l} \\ h_{k,l} \end{bmatrix} \\ &+ \begin{bmatrix} 1 & 0 \\ 0 & 1-\varphi(i_{k,l+1}) \end{bmatrix} \begin{bmatrix} i_{k,l} \\ M(x,\dot{x}) \end{bmatrix}, \\ \varphi(i_{k,l+1}) &= \exp\left(-\left|\frac{\eta_i \cdot i_{k,l} \cdot \gamma \cdot T}{C_k}\right|\right).\end{aligned} \tag{26}$$

where x is the SOC, f the filter state, h the hysteresis voltage, $\boldsymbol{\alpha}$ the vector of filter pole locations, γ the hysteresis rate constant, i the current, $M(\cdot,\cdot)$ maximum hysteresis, η_i the coulombic efficiency, T the length of measurement interval, C the nominal capacity. The coulombic efficiency of a battery cell is defined as the ratio of the amount of charge that is stored in the cell during charging compared to the amount that can be extracted from the cell during discharging. We then obtain the state transition and measurement equations as

$$\begin{aligned} x_{k,l+1} &= F(x_{k,l}, i_{k,l}, C_k) = x_{k,l} + \frac{\eta_i \cdot T \cdot i_{k,l}}{C_k}, \\ y_{k,l+1} &= G(x_{k,l}, i_{k,l}, C_k) = \mathrm{OCV}(z_k) - i_{k,l} \cdot R + h_{k,l+1} + S \cdot f_{k,l+1}. \end{aligned} \tag{27}$$

where OCV is the open circuit voltage, y_k the predicted cell terminal voltage, R the cell resistance, S a vector of constants that blend the time constant states together in the output.

5.2. Multiscale estimation of SOC and capacity

We then begin to introduce the multiscale framework with EKF for the Li-ion battery system by drawing a flowchart in Fig. 1, where T is a fixed time step between two adjacent measurement points, $x_{k,l}$ is the SOC estimate at the time $t_{k,l} = t_{k,0} + l{\cdot}T$., for $1 \leqslant l \leqslant L$ (k and l are the indices of macro and micro time-scales,

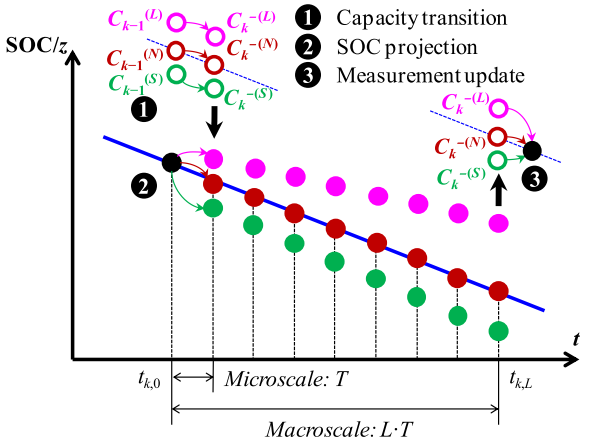

Fig. 2. Procedures of capacity estimation in macro EKF.

respectively), y and i are the cell voltage measurement and the cell current measurement (equivalent to u used before), respectively, and C is the cell capacity estimate (equivalent to θ used before).

The framework consists of two EKFs running in parallel: the top one (micro EKF) adapting the SOC in the micro time-scale and the bottom one (macro EKF) adapting the capacity in the macro time-scale. The micro EKF sends the SOC estimate to the macro EKF and receives the capacity estimate from the macro EKF. In what follows, we intend to elaborate on the macro EKF, the key technical component of the multiscale framework, which consists of the following recursively executed procedures (see Fig. 2):

Step 1: At the macro time step k, the capacity transition step, also referred to as the time-update step, computes the expected capacity and its variance based on the updated estimates at the time step $k-1$, expressed as

$$C_k^- = C_{k-1}^+, \quad \Sigma_{C_k}^- = \Sigma_{C_{k-1}}^+ + \Sigma_{r_{k-1}}. \tag{28}$$

For a stable system, the capacity variance term $\Sigma_{C_{k-1}}^+$ tends to decrease over time with the measurement-update (see *Step 3* in Fig. 2). However, the process noise term $\Sigma_{r_{k-1}}$ always increases the uncertainty of the capacity estimate. To clearly illustrate the proposed idea, we intend to classify the capacity estimates into three cases (see Fig. 2): a larger estimate $C_{k-1}^{(L)}$, an accurate estimate $C_{k-1}^{(N)}$, and a smaller estimate $C_{k-1}^{(S)}$.

Step 2: Based on the capacity estimate C_k^-, the state projection scheme projects the SOC through the macro time step, expressed as a state projection equation derived from Eqs. (17) and (27).

$$x_{k,L} = x_{k,0} + \frac{\eta \cdot T}{C_k^-} \cdot \sum_{j=0}^{L-1} i_{k,j}. \tag{29}$$

As can be seen in Fig. 2, the projected SOCs exhibit large deviations from their true values (obtained from the micro EKF), which suggests a magnified effect of the capacity on the SOC.

Step 3: Following the state projection step, the difference between the projected SOC and the estimated SOC by the micro EKF is used to update the capacity estimate, known as the measurement-update. It is noted that the measurement-update requires accurate SOC estimates which can be obtained from the micro EKF. The updated capacity estimate equals the predicted capacity estimate in *Step 1* plus a correction factor, expressed as

$$C_k^+ = C_k^- + K_k^C[\hat{\mathbf{x}}_{k,L} - \tilde{\mathbf{x}}_{k,L}], \quad \Sigma_{C_k}^+ = (1 - K_k^C C_k^C)\Sigma_{C_k}^-. \tag{30}$$

where the Kalman gain K_k^C and the total derivative C_k^C can be estimated using Eqs. (18) and (23), respectively.

5.3. Remarks on mutiscale framework

We note that the proposed framework decouples the SOC and capacity estimation in terms of both the measurement and time-scale, with an aim to avoid the concurrent SOC and capacity estimation relying on the only measurement (cell terminal voltage) in the dual EKF [1]. In fact, the very motivation of this work lies in the fact that the coupled estimation in the dual EKF falls short in the way of achieving stable capacity estimation, precisely because it is difficult to distinguish the effects of two states (SOC and capacity) on the only measurement (cell terminal voltage), especially in the case of the micro time-scale where the capacity only has a very small influence on the SOC. Regarding the measurement decoupling, the multiscale framework uses the cell terminal voltage exclusively as the measurement to adapt the SOC (micro EKF) which in turn serves as the measurement to adapt the capacity (macro EKF). Regarding the time-scale decoupling, the state projection using the coulomb counting in Eq. (29) significantly magnifies the effect of the capacity on the SOC, i.e., that the capacity affects the SOC projected on the macro time-scale ($L \cdot T$) more significantly than that projected on the micro time-scale (T). The larger influence of the capacity on the SOC leads to the possibility of more stable capacity estimation, and that is precisely the main technical characteristic that distinguishes our approach from the dual EKF.

6. Simulation and experimental results

The verification of the proposed multiscale framework was accomplished by conducting an extensive urban dynamometer drive schedule (UDDS) test. This section reports the results of this test. In Section 6.1, the synthetic data using a valid dynamic model of a high power Li-ion polymer battery (LiPB) cell are used to verify the effectiveness of the multiscale framework. The UDDS test results of a Li-ion prismatic cell are reported in Section 6.2.

6.1. SOC and capacity estimation with synthetic data of high power cell

6.1.1. Synthetic data generation

In order to evaluate the performance of our proposed approach, we generated the synthetic data ($T = 1$ s) using an ESC model of a prototype LiPB cell with a nominal capacity of 7.5 Ah [11]. The root-mean-square (RMS) modeling error compared to cell tests was reported to be less than 10 mV [10]. A sequence of 15 urban dynamometer driving schedule (UDDS) cycles (see Fig. 3a), separated by 30 A constant current discharge and 5 min rest, result in the spread of SOC over the 100–4% range (see Fig. 3b). To account for the measurement error, the current and voltage data were contaminated by zero mean Gaussian noise with standard deviations 200 mA and 10 mV, respectively.

6.1.2. Capacity estimation results

To test the performance of the dual EKF and the multiscale framework with EKF, we intentionally offset the initial capacity value (7.0 Ah) from the true value (7.5 Ah). To minimize the effect of randomness in measurement noise, we repeated this simulation process ten times to obtain ten simulation data sets. The results of capacity estimations on five data sets by these two methods are summarized in Fig. 3c and d, respectively, from which three important observations can be made. First of all, both methods produced converged capacity estimates with similar convergence rates. Since a UDDS cycle covers 7.45 miles [1] and the convergence of the capacity estimate takes about six such cycles (see Fig. 3c and d), the capacity estimate converges within 50miles.

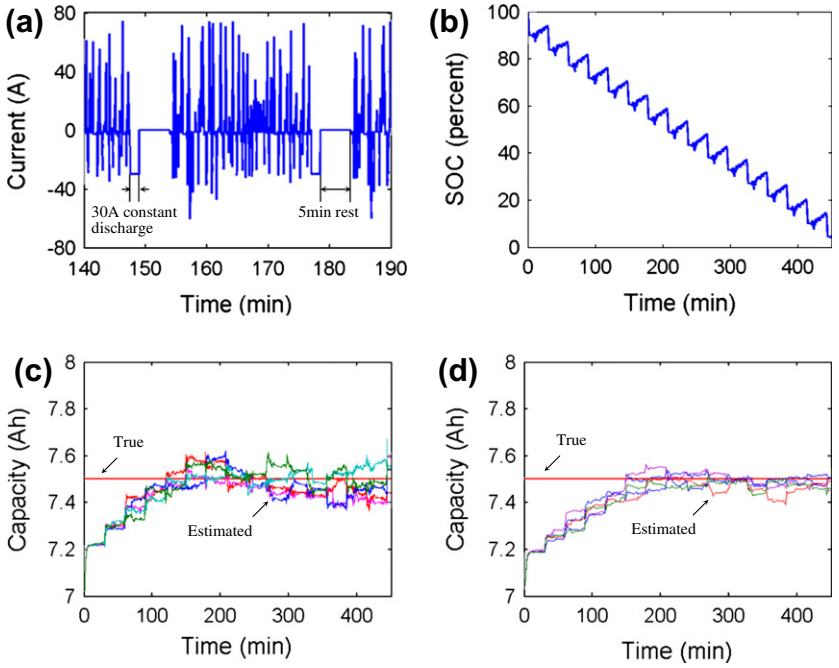

Fig. 3. Synthetic data and results of capacity estimation. (a) Plots the rate profile for one UDDS cycle and (b) plots the SOC profile; (c and d) plot the results of capacity estimation by dual EKF and multiscale framework with EKF, respectively.

Assuming the capacity degradation of an HEV cell over its lifecycle (e.g., 160,000 miles) being less than 20%, we can estimate the time constant (or rate parameter) of the capacity degradation to be in the order of 50,000 miles and thus the required time constant of the capacity filter to be in the order of 10,000 miles (i.e., the capacity filter is five times faster than the capacity degradation). Here, the convergence of the capacity estimate within 50 miles indicates that the time constant should be much less than 50 miles (i.e., only 1/200 of required time constant). Therefore, the convergence by both methods is much faster than required. We note that, with high quality control in cell manufacturing, the initial capacity in real applications would be much closer to the true value than the two extreme cases (much larger and much smaller than the true value) considered here, resulting in a much shorter convergence time. Indeed, the convergence rate can be adjusted by varying the process and measurement noise covariances which, respectively, represent the process uncertainty resulting from the model inaccuracy and the measurement uncertainty resulting from external disturbance that corrupts the measurement data. Secondly, in general, the dual EKF yielded inaccurate and noisy capacity estimation (see Fig. 3c) while the multiscale framework ($L = 100$) with EKF produced more accurate and stable capacity estimation (see Fig. 3d). This can be attributed to the fact that the state projection in Eq. (29) magnifies the effect of the capacity on the SOC as well as removes to some extent the measurement noise. The average RMS errors after convergence (at $t = 200$mins) on the 10 data sets were computed as 0.048 Ah (relative error 0.640%) and 0.033 Ah (relative error 0.440%) for the dual EKF and the multiscale framework with EKF, respectively. Thirdly, it is observed that, although the multiscale framework with EKF produced stable capacity estimation, the estimate still exhibits small fluctuation over time. It is fair to say, however, that this small noise does not really affect the practical use of this estimate.

6.1.3. Computational efficiency

In the previous subsection, we have demonstrated that the proposed multiscale framework yielded higher accuracy than the dual EKF. In this subsection, we compare the two methods in terms of computational efficiency. To minimize the effect of randomness in measurement noise, we employed the ten synthetic data sets with each being executed ten times. Our computations were carried out on a processor Intel Core i5 760 CPU 2.8 GHz and 4 GByte RAM. The codes for both methods were self-devised hand-optimized MATLAB codes running in Matlab environment (MATLAB Version 7.11.0.584, The MathWorks, Inc., Natick, MA, USA). To make our comparison of general use to other engineering systems, we ruled out the computational time required to execute the ESC model in this study. In fact, the measurement functions of two engineered systems may exhibit a large difference in the level of computational complexity, resulting in different amounts of computational time. Thus, we intend to minimize the effect of system-to-system variation and focus on the general functions in an EKF by assuming a negligibly small amount of time for the execution of the system-specific measurement function (ESC model). Table 3 summarizes the mean computational times. It is observed that the multiscale framework with EKF requires a smaller amount of computational time of 1.456 s for the sequence of 15 UDDS cycles, a 34.145% reduction over the dual EKF whose computational time is 2.210 s. Note that the percent of improvement is less than 50.000%. This can be attributed to the following two reasons: (i) from the standpoint of computations on the micro time-scale, it is noted that, in addition to the time- and measurement-update computations for SOC estimation, both methods also require the recursive derivative computation which, to some extent, reduces their efficiency gap; and (ii) from the standpoint of computations on the macro time-scale, although the macro-EKF is executed only upon the completion of $L = 100$ executions of the micro-EFK, it still requires a certain amount of time to compute the time- and

Table 3
Comparison results of computation efficiency with ten synthetic data sets.

Method	Computational time (s)	Improvement (%)
DualEKF	2.210	–
Mutiscale Framework with EKF	1.456	34.145

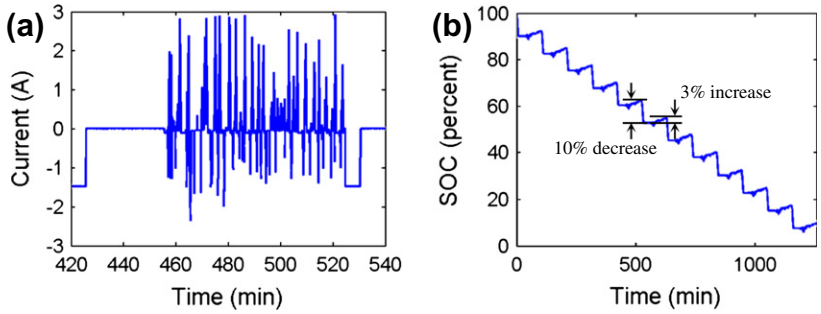

Fig. 4. SOC profile and one cycle rate profile for UDDS cycle test. (a) Plots the rate profile for one UDDS cycle and (b) plots the SOC profile for 12 UDDS cycles.

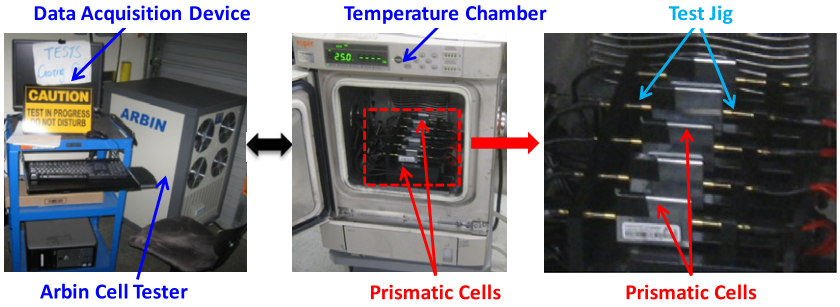

Fig. 5. Experiment setup – UDDS test system.

measurement-updates for capacity estimation. In spite of these points, it is fair to say, however, that the proposed method achieves considerable improvement over the dual EKF in terms of computational efficiency. This improvement is critical to alleviating the computational burden imposed on the hardware and thus enhancing the feasibility of applications.

6.2. SOC and capacity estimation with UDDS cycle test of a prismatic cell

6.2.1. Description of test procedure

In addition to the numerical study using synthetic data, we also conducted the UDDS cycle test to verify the effectiveness of the multiscale framework. The cells used in the test are Li-ion prismatic cells with a nominal capacity of 1.5 Ah. Since the cell cannot withstand the high current pulse on a typical HEV cell, the UDDS profile (see Fig. 3a) was scaled down to within the rate range of ±2 C. The scaled UDDS cycle was replicated two times to obtain the final UDDS cycle used in this test (see Fig. 4a). It is noted that, in a battery system, we often use C or C-rate to measure the rate at which a cell is charged or discharged relative to its full capacity. For the Li-ion prismatic cell with the capacity of 1.5 Ah, a discharge current with a 1 C rate (1.5 A) will discharge the full cell capacity in 1 h. Here, 2 C or a 2 C rate (3.0 A) defines the upper and lower bounds (±3.0 A) of the scaled UDDS profile (see Fig. 4a). The cycle test is composed of 12 UDDS cycles, separated by 1C constant current discharge for 6 min and 30 min rest. This test profile resulted in the spread of SOC over the 100–4% range. The SOC profile for 12 UDDS cycles is plotted in Fig. 4b, where the cell experiences an SOC increase by about 3% during each UDDS cycle, and an SOC decrease by about 10% due to the 1C discharge between cycles. The discharge setting (1C for 6 min) was designed in order to excite the entire SOC range (100–4%) for the UDDS cycle test as well as to practice the UDDS cycle test at many different SOC levels separated by a small gap (about 7%).

We set up a UDDS test system (see Fig. 5) which comprises of an Arbin BT2000 cycle tester with a data acquisition device, an Espec SH-241 temperature chamber at 25 °C and a test jig as a connector holder for prismatic cells. Five prismatic cells were placed in the temperature chamber and held by the test jig throughout the test.

6.2.2. Training of ESC cell model

The current and voltage measurements of one cell (cell 1) were used to train the ESC model [1] while the other four cells (cells 2–5) were treated as the testing cells. We followed the procedures described in [15] to obtain the open circuit voltage (OCV) curve. Through numerical optimization, optimum ESC model parameters were obtained which minimize the root mean squared (RMS) error of the cell terminal voltage. The numerical optimization was performed using with a sequential quadratic programming (SQP) method. In this study, we employed a nominal capacity of 1.5 Ah, a measurement interval of $T \approx 1$s with "≈" indicating small measurement-to-measurement fluctuation, and four filter states $n_f = 4$. The voltage modeling results for one UDDS cycle are shown in 0a, where a good agreement can be observed between the modeled and measured cell terminal voltage. The RMS error of voltage modeling for 12 UDDS cycles was 13.3 mV.

6.2.3. SOC and capacity estimation results

The SOC estimation results for the training cell for all 12 UDDS cycles, the 3rd UDDS cycle and the 7th UDDS cycle are shown in Fig. 6b–d, respectively, where the initial SOC is set to be smaller (90%) than the true SOC (100%) and the multiscale framework ($L = 1200$) still produced converged SOC estimate. Table 4 summarizes the SOC estimation errors under four different settings of the initial SOC and capacity. Here, the RMS and maximum errors take into account the initial offset in the case of an incorrect initial SOC and are formulated as

$$\varepsilon_{RMS} = \sqrt{\frac{1}{nm}\sum_{k,j}(\hat{x}_{k,l} - x_{k,l})^2}, \quad \varepsilon_{Max} = \max_{k,j}|\hat{x}_{k,l} - x_{k,l}|. \tag{31}$$

where nm is the number of measurements and reads 74,484 (about 1250 min) in this study; and $x_{k,l}$ is the true SOC at the time $t_{k,l}$. In

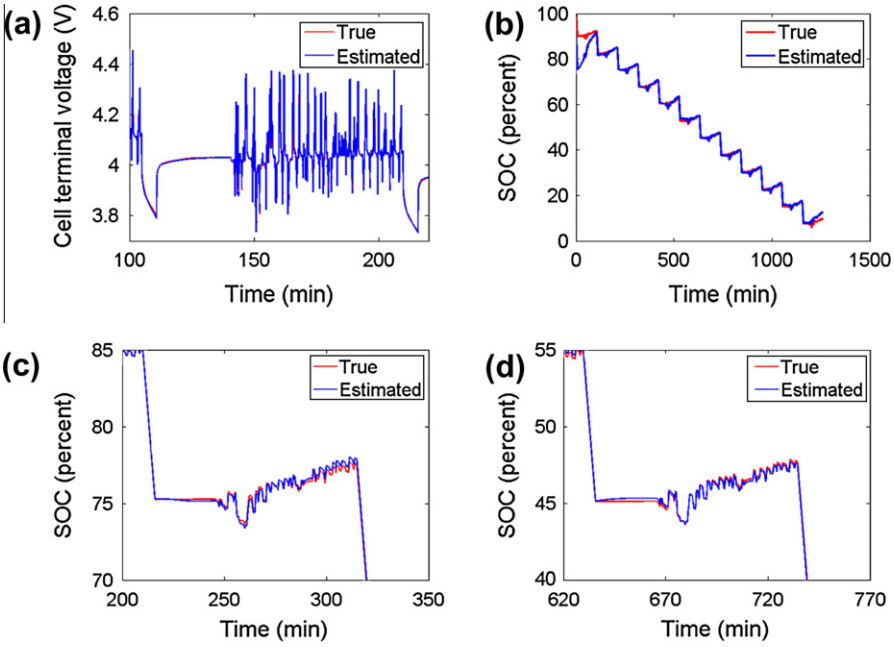

Fig. 6. Voltage modeling results and SOC estimation results for UDDS cycle test. (a) Plots modeled and measured cell terminal voltage for one UDDS cycle; (b–d) plot the estimated and true SOCs for all 12 UDDS cycles, the 3rd UDDS cycle and the 7th UDDS cycle, respectively.

Table 4
SOC estimation results under different settings of initial SOC and capacity.

Initial SOC	Initial capacity	SOC errors	Cell1	Cell 2	Cell 3	Cell 4	Cell 5
Correct (100%)	Correct (1.5 Ah)	RMS (%)	1.02	1.34	0.81	1.05	0.75
		Max (%)	2.19	2.37	1.82	2.79	1.91
	Incorrect (1.0 Ah)	RMS (%)	1.31	1.59	1.10	1.39	1.14
		Max (%)	4.84	4.84	4.84	4.85	4.84
	Correct (1.5 Ah)	RMS (%)	1.91	2.07	1.84	2.03	1.8
		Max (%)	10.00	10.00	10.00	10.00	10.00
Incorrect (90%)	Incorrect (1.0 Ah)	RMS (%)	2.65	2.77	2.59	2.76	2.58
		Max (%)	14.74	14.75	14.74	14.76	14.75

this study, we computed the true SOCs based on the coulomb counting technique. We attempted to achieve an accurate approximation to the true SOCs through the following ways: (i) we first fully charge a battery cell with a constant-current (0.1 A constant current up to 4.2 V) constant-voltage (4.2 V constant voltage down to 0.01 A) strategy to ensure an accurate initial SOC (100%); (ii) we measure the cell current with the Arbin current sensors whose high measurement accuracy leads to a sufficiently small error in the current accumulation over a relatively short test duration (around 20 h).

Three important observations can be made from the results. First of all, it is observed that the RMS SOC estimation errors produced by the multiscale framework are less than 3.00%, regardless of initial values of the SOC and capacity. Secondly, under both initial SOC settings, the SOC estimation errors with incorrect initial capacity (1.0 Ah) are consistently larger than those with correct initial capacity (1.5 Ahs). These results suggest that the SOC is strongly dependant on the capacity and that the lack of accuracy in the capacity estimation may reduce the accuracy in the SOC estimation. It is thus important to produce accurate capacity estimation not only to provide insights into the cell SOH but also to enable accurate SOC estimation. Thirdly, under both initial capacity settings, the SOC estimation with incorrect initial SOC (90%) consistently shows larger errors than that with correct initial SOC (100%). Clearly, the larger SOC errors under incorrect initial SOC (90%) can be attributed to the larger errors before the convergence of SOC estimation (at the initial stage). After the convergence, the SOC errors under different initial SOCs become almost the same. We note that the RMS SOC estimation errors with incorrect initial SOC (90%) are still less than 3.00% since the multiscale framework produced converged SOC estimate for both cases.

Regarding the capacity estimation, both methods with initial values smaller than the true value (see Fig. 7a and c) and larger than the real value (see Fig. 7b and d) for all the five cells achieves convergence to the true capacity within an error range of around 5%. Compared with the capacity estimation (see Fig. 7a and b) by the multiscale framework, the capacity estimation (see 0c and 0d) by the dual EKF contains larger noise. The poorer accuracy produced by the dual EKF (consisting of an SOC EKF and a capacity EKF) can be attributed to the measurement and time-scale coupling in the SOC and capacity estimation. Regarding the measurement coupling, the dual EKF uses the cell terminal voltage as the measurement to adapt both the SOC and capacity. When the voltage modeling contains relatively large errors, the capacity estimation can be largely compromised by the measurement update (in the capacity EKF) which only aims at minimizing the difference between the modeled and measured voltages. In other words, the measurement update may give an incorrect capacity estimate to counteract the voltage modeling error. In this experimental study, the relatively large voltage modeling errors directly affect the

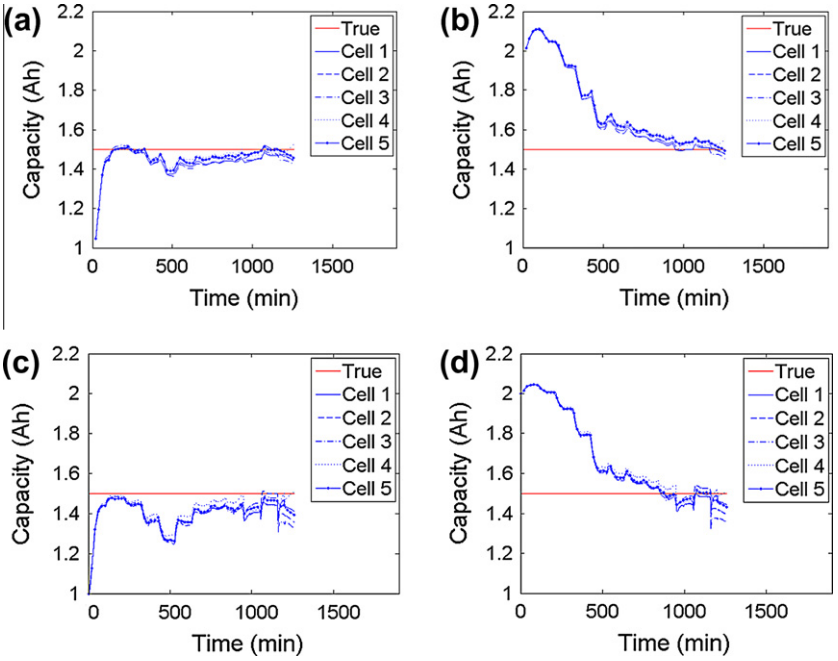

Fig. 7. Capacity estimation results for UDDS cycle test. (a and b) Plot capacity estimation results by the multiscale framework with the initial values smaller than and larger than the true value, respectively; (c and d) plots capacity estimation results by the dual EKF with the initial values smaller than and larger than the true value, respectively.

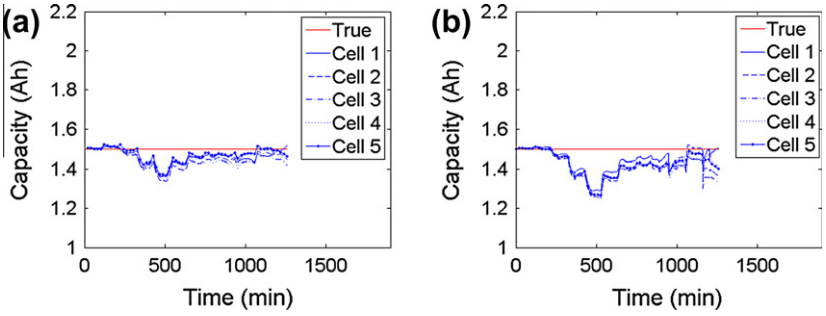

Fig. 8. Capacity estimation results after convergence (by setting initial capacity as real value). (a and b) Plot capacity estimation results by the multiscale framework and by the dual EKF, respectively.

accuracy in the capacity estimation by the dual EKF. Regarding the time-scale coupling, the dual EKF estimates both the SOC and capacity on the micro time-scale, which makes the capacity estimation vulnerable to the local voltage modeling error (on the micro time-scale). This vulnerability further leads to noisy capacity estimation. In contrast, the multiscale framework decouples the SOC and capacity estimation in terms of both the measurement and time-scale and avoids the concurrent SOC and capacity estimation relying on the only measurement (cell terminal voltage). The decoupling enables accurate capacity estimation in spite of SOC estimation error. To quantify the accuracy of both methods, we computed average RMS errors after convergence (at $t = 200$min and 1000 min for smaller and larger initials, respectively). For the smaller initial, the average RMS errors produced by the dual EKF and the multiscale framework are 0.108 Ah (relative error 7.227%) and 0.063 Ah (relative error 4.200%), respectively. For the larger initial, the errors are 0.049 Ah (relative error 3.233%) and 0.023 Ah (relative error 1.533%). Finally, we note that, since the 12 UDDS cycle test on one cell can be treated as one unique cycle test, we do not expect large deviation from the current results (based on the small difference between capacity estimation results on different cells in 0) if we conduct another 12 UDDS cycle test.

To investigate the long-term behavior of capacity estimation after convergence, we set the initial capacity value as the real value and executed the two methods over the 12 UDDS cycles for all the five cells. It is noted that this is virtually equivalent to adding another 12 UDDS cycles (after convergence of capacity estimates) for the cases of smaller and larger initial capacities (see Fig. 7). As can be seen in Fig. 8a and b, both methods produced capacity estimates around the real value. Again, the capacity estimation (see Fig. 8a) by the multiscale framework contains smaller noise than that (see Fig. 8b) by the dual EKF. To quantify the accuracy of both methods, we computed average RMS errors over the entire time domain. The errors produced by the dual EKF and the multiscale framework read 0.099 Ah (relative error 6.573%) and 0.059 Ah (relative error 3.931%), respectively.

In addition to the accuracy comparison, we also compared the two methods in terms of computational efficiency. To minimize the effect of randomness, we executed both methods ten times with the test data obtained from each of the five cells. The mean computational time is summarized in Table 5, where we observe that the multiscale framework consumed less computational time than the dual EKF and the improvement ratio shows good consistency to what we observe in the synthetic data (see Table 3).

Table 5
Comparison results of computation efficiency with UDDS test data.

Method	Computational time (s)	Improvement (%)
DualEKF	5.813	–
Mutiscale Framework with EKF	3.711	36.163

7. Conclusion

The multiscale framework with EKF is proposed for efficient and accurate state and parameter estimation for engineered systems that exhibit time-scale separation. The proposed framework was applied to the Li-ion battery system for SOC and capacity estimation. When applied to the battery system, the multiscale framework can be treated as a hybrid of coulomb counting and adaptive filtering techniques. Our contribution to battery SOC and capacity estimation lies in the construction of a multiscale computational scheme that decouples the SOC and capacity estimation from two perspectives, namely the measurement and time-scale. The resulting decoupled estimation greatly reduces the computational time involved in obtaining the SOC and capacity estimates, while enhancing the accuracy in the capacity estimation. It is noted that the higher efficiency makes the proposed methodology more suitable for onboard estimation devices that require computationally efficient estimation techniques. Experiments with the synthetic data and UDDS cycle test verify that the proposed framework achieves more accurate and efficient capacity estimation than the dual EKF, suggesting that the proposed framework is a promising methodology for the battery prognostics.

In this work, we place our focus on SOC and capacity estimation at the cell level. Since battery packs in EVs/HEVs often contain thousands of cells, it is thus of significant importance to extend the proposed framework from a cell level to a pack level in order to make it practically useful. One possible way to accomplish this extension is to replicate the multiscale framework M times to compute cell SOCs and capacities in a pack consisting of M cells connected in series, parallel, or a mixed mode. Clearly, this will work, as long as we have (1) well-trained cell dynamic model for individual cells and (2) the voltage measurements of each parallel pathway and the current measurements of each series pathway. However, a simple replication results in $2M$ EKFs (e.g., 100 cells need 200 EKFs) which require a prohibitively large amount of computational efforts. Thus, it will be very interesting to investigate how we can utilize the strong correlation between SOCs and capacities of all interconnected cells in a battery pack in order to significantly reduce the computational efforts for real-time estimation at the pack level. Other future works under contemplation include the investigation the effect of the level of time-scale separation on the accuracy, the verification of the practical feasibility of the multiscale framework by burning it into integrated circuit (IC) chips in battery management systems (BMSs) and conducting extensive verification tests, and the extension of the proposed method to the state of life (SOL) prediction with lifetime cell aging test.

Acknowledgments

This research was partially supported by the Maryland Industrial Partnerships Program (MIPS) Contract, the Institute of Advanced Machinery and Design at Seoul National University (SNU-IAMD), and the National Research Foundation of Korea (NRF) grant funded by the Korea government (No. 2011-0022051). The authors gratefully acknowledge PCTEST Engineering Laboratory Inc. for providing testing facilities and Prof. Gregory L. Plett for providing the UDDS profile for this research.

References

[1] Plett GL. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs – Part 3. State and parameter estimation. J Power Sources 2004;134(n2):277–92.
[2] Plett GL. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs – Part 2: Simultaneous state and parameter estimation. J Power Sources 2006;161(n2):1369–84.
[3] Lee S, Kim J, Lee J, Cho BH. State-of-charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. J Power Sources 2008;185(n2):1367–73.
[4] Zhang Q, White RE. Capacity fade analysis of a lithium ion cell. J Power Sources 2008;179(n2):793–8.
[5] Zhang Q, White RE. Calendar life study of Li-ion pouch cells – Part 2: Simulation. J Power Sources 2008;179(n2):785–92.
[6] Ng KS, Moo CS, Chen YP, Hsieh YC. Enhanced coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries. Appl Energy 2009;86(n9):1506–11.
[7] Sun YH, Jou HL, Wu JC. Auxiliary health diagnosis method for lead-acid battery. Appl Energy 2010;87(n9):3691–8.
[8] Saha B, Goebel K, Poll S, Christophersen J. Prognostics methods for battery health monitoring using a Bayesian framework. IEEE Trans Instr Measur 2009;58(n2):291–6.
[9] Saha B, Goebel K. Modeling Li-ion battery capacity depletion in a particle filtering framework. In: Proceedings of annual conference of the PHM Society, San Diego, CA; 2009 [September 27–October 1].
[10] Plett GL. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs – Part 2. Modeling and identification. J Power Sources 2004;134(n2):262–76.
[11] Plett GL. Sigma-point Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1: Introduction and state estimation. J Power Sources 2006;161(n2):1356–68.
[12] Haykin S. Kalman filtering and neural networks. New York: Wiley/Inter-Science; 2001.
[13] Wan E, Nelson A. Dual extended Kalman filter methods. In: Haykin S, editor. Kalman filtering and neural networks. New York: Wiley/Inter-Science; 2001. p. 123–74.
[14] Williams RJ, Zipser D. A learning algorithm for continually running fully recurrent neural networks. Neural Comput 1989;1(2):270–80.
[15] Plett G. Results of temperature-dependent LiPB cell modeling for HEV SOC estimation. In: Proceedings of the 21st electric vehicle symposium (EVS21), Monaco; 2005 (April 2–6).