3-D Temperature Field Reconstruction for a Lithium-Ion Battery Pack: A Distributed Kalman Filtering Approach

Ning Tian, Huazhen Fang, and Yebin Wang

***Abstract*— Despite the ever-increasing use across different sectors, the lithium-ion batteries (LiBs) have continually seen serious concerns over their thermal vulnerability. The LiB operation involves heat generation and buildup effect, which manifests itself strongly, in the form of highly uneven thermal distribution, for a LiB pack consisting of multiple cells. If not well monitored and managed, the heating may accelerate aging and cause unwanted side reactions. In extreme cases, it will even cause fires and explosions. Toward addressing this threat, this brief, for the first time, seeks to reconstruct the 3-D temperature field of a LiB pack in real time. The major challenge lies in how to acquire a high-fidelity reconstruction with constrained computation time. In this brief, a 3-D thermal model is established first for a LiB pack configured in series, which captures the spatial thermal behavior with a combination of high integrity and low complexity. Given the model, the standard Kalman filter is then distributed to attain temperature field estimation with substantially reduced computational complexity. The arithmetic operation analysis and the numerical simulation illustrate that the proposed distributed estimation achieves a comparable accuracy as the centralized approach but with much less computation.**

***Index Terms*— Distributed Kalman filtering, lithium-ion battery (LiB) pack, temperature estimation, thermal modeling.**

I. Introduction

LITHIUM-ION batteries (LiBs) are widely used in various applications due to their high energy/power density, long cycle life, and low self-discharge rate. This trend has stimulated significant research of battery management algorithms [1]–[3]. However, LiBs are also known to be thermally vulnerable. Heat can be generated during charging and discharging due to irreversible overpotential heating, reversible entropic heating from electrochemical reactions, phase change heating, and mixing effects [4]. Without timely removal, the heat can gradually build up, which will lead to not only many side reactions, but also performance degradation, aging acceleration, and even fires [5]. Recent years, hence, have witnessed a growing research in real-time temperature monitoring.

Currently, a large amount of work has been devoted to temperature estimation using a thermal model and the surface temperature measurements [6]–[8]. These studies consider low-order lumped thermal models that concentrate the spatial dimensions into singular points. Though advantageous for computation, lumped models introduce a significant simplification, because the temperature distribution is nonuniform spatially within a cell. For improvement, some recent works [9]–[11] study the temperature estimation with some awareness of the spatial nonuniformity, which use thermal models accounting for the LiB cells spatial dimensions to a certain extent. These models yet are still simplified at the sacrifice of their physical fidelity. It is noteworthy that the foregoing studies are focused on thermal management for a single LiB cell. The issue can become much more challenging when LiB packs are considered. With a few cells stacked in a compact space, a LiB pack has larger dimensions and more complicated thermal behavior that will render the cell-level approaches unproductive. The challenge can become more daunting as large-format high-capacity LiB cells are preferred increasingly to assemble packs and modules, since heat will be generated in larger amounts and more complex manners when the cell increases in size. In [12]–[16], the notion of lumped modeling is extended to depict the thermal dynamics of LiB packs composed of small cylindrical cells. However, a pack's spatially uneven thermal behavior, though crucial, has received rare attention to date.

As the first of its kind, this brief proposes to reconstruct the 3-D temperature field of a LiB pack in real time. The fundamental notion is to acquire a spatially resolved thermal model for a LiB pack and then apply the Kalman filter (KF) technique to estimate the spatially distributed temperature. However, the task is nontrivial given the complexity of a LiB pack's thermal behavior. To fulfill the goal, a twofold effort is made, which lies in modeling and KF-based estimation.

Thermal modeling for LiBs has attracted a wealth of research, with the methodologies falling in three categories: 1) thermal models, which are concerned only with the heat phenomena and based on the thermal energy conservation principle, often given in the form of partial differential equations (PDEs) in 3-D space [17]–[19]; 2) coupled thermal-electrochemical models, which associate the equations for thermal behavior with those for electrochemical reactions [20]–[24]; and 3) lumped parameter models, which reduce the spatially distributed heat transfer into a heat flow passing through several discrete points (e.g., two points representing the cells core and surface and connected by a thermal resistance) [6]–[11], [16]. Among them, coupled thermal-electrochemical models can offer a detailed view of the LiB behavior with electrochemical reactions characterized at multiple scales. This, however, requires computing costs

Manuscript received August 24, 2017; accepted October 31, 2017. Manuscript received in final form November 28, 2017. Recommended by Associate Editor C. Manzie. *(Corresponding author: Huazhen Fang.)*

N. Tian and H. Fang are with the Department of Mechanical Engineering, University of Kansas, Lawrence, KS 66045 USA (e-mail: ning.tian@ku.edu; fang@ku.edu).

Y. Wang is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139 USA (e-mail: yebinwang@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2017.2779434

formidable enough to defy real-time estimation. For lumped models, the simplicity is conducive to estimation design for thermal management, but the spatial information loss weakens their capability for more effective temperature monitoring. While these two types of models represent two extremes in terms of model fidelity or computational efficiency, the thermal models strike a valuable balance, thereby offering great promises for thermal management with spatial awareness. In this regard, despite prolific results on cell-level thermal modeling, the research about pack-level modeling is still at a nascent age. Hence, this brief will investigate the development of a 3-D LiB pack thermal model.

When the thermal model is available, the temperature field reconstruction will depend on the estimation technique, which is meant to estimate the temperature at any spatial point using the model and the temperature measurement data. Here, the celebrated KF is one of the most promising candidate tools due to its ability to deal with the stochastic dynamic systems affected by noise—the thermal dynamics of a LiB pack can be subjected to the process noise in its evolution and the measurement noise when the temperature is measured by sensors. A direct application of the standard centralized KF (CKF) here is possible, but will cause hefty computational burden. This is because of the KF's computational complexity being cubic with the size of the state space [25], and a spatially resolved LiB pack model will have a substantial number of states, especially when the pack comprises many cells. To address this problem, a distributed KF (DKF) approach will be undertaken to enhance computational efficiency, which reduces a global KF into multiple local KFs running in parallel. The overall computational complexity of this approach will increase only linearly with the number of LiB cells in the pack, in contrast with the cubic increase for the CKF. This advantage can considerably benefit practical application.

This brief is organized as follows. Section II develops a spatially resolved thermal model for a LiB pack and its state-space form. Section III develops the computationally efficient DKF for temperature field reconstruction. Section IV presents a simulation to demonstrate the efficacy of the proposed approaches. Finally, concluding remarks are gathered in Section V.

II. Battery Pack Thermal Modeling

This section presents a spatially resolved thermal model for a LiB pack, which is an extension of a cell-level model in [17].

A. Spatially Resolved Thermal Model

Consider a LiB pack shown in Fig. 1. This pack consists of multiple prismatic cells configured in series (see [17] for the cell description). Each cell has two areas: core region and cell case. The core region is the main body of a cell. It consists of many smaller cell units connected in parallel to provide high capacity, with each unit composed of electrodes, current collectors, and a separator. While this makes the cell similar to a module, we still refer to it as a cell as it is the basic building block of a battery pack. The cell case is a metal container and also includes a contact layer filled with liquid electrolyte and in touch with the core region. In this setting, modeling will

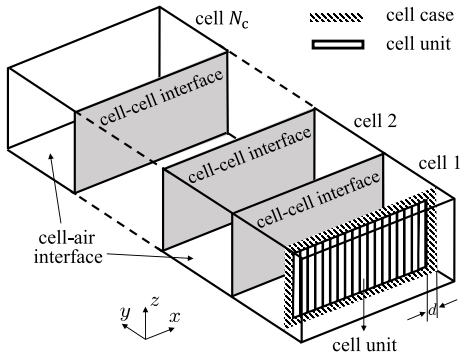

Fig. 1. Schematic of a LiB pack.

be performed next to describe the thermal dynamics in the core region and the case and on the boundaries, including the core–case interface boundary, cell–cell interface boundary, and cell–air interface boundary.

Consider the core region first. It is known that there are three ways for heat transfer: conduction, convection, and radiation. Here, convection and radiation can be ignored, since the liquid electrolyte is of limited mobility, and electromagnetic waves can hardly transmit through the cell. Consequently, conduction dominates the heat transfer within the core region, which can be expressed as

$$\rho_{\rm co} c_{\rm co} \frac{\partial T}{\partial t} = \lambda_{\rm co} \nabla^2 T + q \tag{1}$$

where $\rho_{\rm co}$, $c_{\rm co}$, and $\lambda_{\rm co}$ are the mass density, specific heat capacity, and thermal conductivity of the core region, respectively. In addition, T and q, respectively, denote the core region's temperature in kelvins and heat generation density. As shown in [5], [18], and [26], a general characterization of q is offered by

$$q = \frac{I}{V_{\rm co}} \left[(U_{\rm ocv} - U_{\rm t}) - T \frac{dU_{\rm ocv}}{dT} \right] \tag{2}$$

where $V_{\rm co}$, I, $U_{\rm ocv}$, $U_{\rm t}$, and $dU_{\rm ocv}/dT$ denote the total volume of the core region, the current through pack (positive for discharge, negative for charge), the open-circuit voltage, the terminal voltage, and the entropic heat coefficient, respectively. Here, the heat is assumed to be generated uniformly across the core region. The first term on the right-hand side of (2) is the irreversible heating, and the second term is the reversible entropic heating from electrochemical reactions.

Similarly, conduction makes up the dominant part of the heat transfer in the case region, that is,

$$\rho_{\rm ca} c_{\rm ca} \frac{\partial T}{\partial t} = \lambda_{\rm ca} \nabla^2 T \tag{3}$$

where $\rho_{\rm ca}$, $c_{\rm ca}$, and $\lambda_{\rm ca}$ are the mass density, specific heat capacity, and thermal conductivity of the case, respectively.

Next, consider the boundaries. To begin with, heat transfer on the core–case interface is mainly due to conduction. Assuming a perfect core–case contact, the temperature and heat flux can be considered continuous on the interface. Hence, the boundary conditions on the core–case interface are given by

$$T|_{\rm core} = T|_{\rm case}, \quad \lambda_{\rm co} \left. \frac{\partial T}{\partial n} \right|_{\rm core} = \lambda_{\rm ca} \left. \frac{\partial T}{\partial n} \right|_{\rm case} \tag{4}$$

where n is the normal direction. The continuity of temperature and heat flux at a boundary also holds for the cell–cell interface if LiB cells are in close contact. For example, on the interface between cells 1 and 2 in Fig. 1, one has

$$T|_{\text{cell 1}} = T|_{\text{cell 2}}, \quad \lambda_{\text{ca}} \frac{\partial T}{\partial n}\bigg|_{\text{cell 1}} = \lambda_{\text{ca}} \frac{\partial T}{\partial n}\bigg|_{\text{cell 2}}. \tag{5}$$

In addition, the cell–air interface will see heat transfer due to all of conduction, convection, and radiation. Therefore, the energy balance on this boundary is

$$\lambda_{\text{ca}} \frac{\partial T}{\partial n}\bigg|_{\text{cell}} = q_{\text{conv}} + q_{\text{r}} \tag{6}$$

where q_{conv} and q_{r} represent the convective and radiative heat flux on the cell–air interface, respectively. They are given by

$$q_{\text{conv}} = h_{\text{conv}}(T - T_{\text{air}}) \tag{7}$$

$$q_{\text{r}} = \varepsilon\sigma\left(T^4 - T_{\text{air}}^4\right) \tag{8}$$

where h_{conv}, ε, σ, and T_{air} are the convective heat transfer coefficient, the emissivity, the Stefan–Boltzmann's constant, and the ambient air temperature, respectively. Note that if $|T - T_{\text{air}}|/T_{\text{air}} \ll 1$ as is often the case of LiB operation, (8) can be linearized around T_{air} as

$$q_{\text{r}} = 4\varepsilon\sigma T_{\text{air}}^3 (T - T_{\text{air}}) = h_{\text{r}}(T - T_{\text{air}}) \tag{9}$$

where h_{r} is the radiative heat transfer coefficient. Combining (7) and (9), the energy balance (6) can be rewritten as

$$\lambda_{\text{ca}} \frac{\partial T}{\partial n}\bigg|_{\text{cell}} = h(T - T_{\text{air}}) \tag{10}$$

where h is a combined heat transfer coefficient.

Summarizing (1)–(10), one will obtain a complete thermal model for the considered LiB pack. Spatially resolved, this model can capture the spatial temperature distribution. According to (1) and (2), when a current flows through the pack, heat is produced within the core region and transferred across the region by conduction. Conduction will also enable the propagation of heat within the case region, which is shown in (3). The boundary conditions at the core–case and cell–cell interfaces can be determined as in (4) and (5) on the reasonable assumption of continuous temperature and heat flux. Meanwhile, heat will travel from the cell surface to the air driven by a mix of conduction, convection, and radiation, as shown in (6). The radiation effect at the cell–air interface is further linearized to simplify the model, which would lead to (10).

Remark 1 (Extensions of the Thermal Model): The above-mentioned thermal model is developed in a basic battery pack setting but able to capture the most critical heat transfer phenomena underlying a pack's thermal behavior. It can be extended to more sophisticated settings.

1) *Extension to a Battery Pack with a Cooling System:* The cooling effects can be accounted for in two ways. First, as suggested in [27], one can regard the cooling system as the boundaries of the battery pack's thermal model and thus modify the boundary conditions accordingly. Second, one can develop a separate heat transfer model for the cooling system and determine its interaction with the pack's model. The two models can be combined to offer a complete description of the battery pack under cooling conditions. This idea is exploited in [13]–[15].
2) *Extension to Nonuniform Heat Generation:* As shown in (2), the heat generation is assumed to be even across a cell's core region. A modification can be made to cope with spatially nonuniform heating if the gradient distribution of the potential and current density is captured. The literature includes some studies on this topic (see [5] and the references therein).
3) *Extension to Heterogeneous Cells:* While the above considers identical cells, cells of the same type but of a different state or aging level can be dealt with by changing the model parameters. Furthermore, if cells of different electrochemistries are used in an extreme case, one can first build separate models for each cell type and then couple them using the same heat transfer principles to obtain a pack-level model. It is noteworthy that, though based on the basic model in (1)–(10), the temperature estimation methodology to be proposed next is still applicable to the extended models.

•

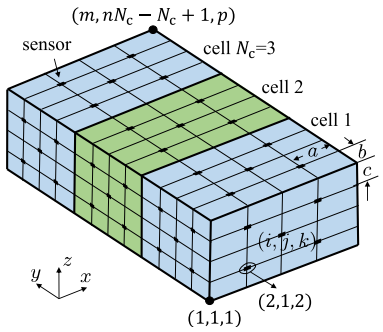

Fig. 2. Schematic of the nodes in a LiB pack.

B. Reduction to State-Space Model

For the purpose of estimation, the above thermal model must be discretized in space and time to derive a control-oriented state-space model. The finite-difference formulation can be applied here. Consider the LiB pack comprising N_c cells. It can be subdivided into a large number of volume elements, giving rise to a 3-D grid with many nodes as in Fig. 2. Specifically, a LiB cell is subdivided by a grid with $(m \times n \times p)$ nodes. Here, two adjacent cells share the mp nodes on their interface. Then, each node can be labeled by its coordinates, i.e., (i, j, k), which ranges from $(1, 1, 1)$ to $(m, (n-1)N_c+1, p)$. A node is linked with a control volume. Within this volume, the temperature is considered uniform and assigned as the temperature of the node. The finite-difference equation at each node is developed by the energy balance approach [28]. The detailed procedure is omitted here for the sake of space, but a reader can refer to [29] for details.

Reorganizing the obtained finite-difference equations, one can derive a high-dimensional state-space representation of the following general form:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{G}_k \boldsymbol{u}_k. \tag{11}$$

Here, $\boldsymbol{x}$ is the state vector summarizing the temperature at all the nodes. Specifically, $\boldsymbol{x} = \text{column}(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N_c)})_{N\times 1}$,

and $\boldsymbol{x}^{(l)} = \text{column}(T^{(l)}_{1,1,1}, \ldots, T^{(l)}_{m,n,p})_{N_l \times 1}$, with $N = N_c N_l$. Note that, for ease of the decomposition strategy design toward distributed estimation, the global state vector $\boldsymbol{x}$ is established as an aggregation of local state vector $\boldsymbol{x}^{(l)}$ of each cell. To accommodate this representation, the above-introduced pack-based numbering method is modified accordingly to be cell-based—for each cell, the nodes are numbered from $(1, 1, 1)$ to (m, n, p). The input $\boldsymbol{u}$ is $\boldsymbol{u} = [T_{\text{air}} \ I]^\top$, which comprises the ambient temperature and applied current. Furthermore, $\boldsymbol{F}$ and $\boldsymbol{G}$ can be expressed in a block form

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_{11} & \boldsymbol{F}_{12} & & \boldsymbol{0} \\ \boldsymbol{F}_{21} & \boldsymbol{F}_{22} & \ddots & \\ & \ddots & \ddots & \ddots \\ \boldsymbol{0} & & \ddots & \boldsymbol{F}_{N_c,N_c} \end{bmatrix}, \quad \boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_1 \\ \vdots \\ \boldsymbol{G}_{N_c} \end{bmatrix}.$$

Note that the tridiagonal $\boldsymbol{F}$ is also diagonally dominant and sparse, because a cell only exchanges heat with its adjacent cells and adjacent cells share a limited number of nodes.

Let cell l in the pack be equipped with M_l thermocouples to measure the temperature. The measurement equation can then be expressed as

$$\boldsymbol{y}^{(l)}_k = \boldsymbol{H}_l \boldsymbol{x}^{(l)}_k \tag{12}$$

where $\boldsymbol{y}^{(l)} \in \mathbb{R}^{M_l}$ and $\boldsymbol{H}_l \in \mathbb{R}^{M_l \times N_l}$ are the temperature measurement and measurement matrix, respectively. For $\boldsymbol{H}$, the entries corresponding to the nodes directly measured are set equal to 1, and all the other entries zero. Aggregating all the measurements together, the measurement equation for the entire pack is then given by

$$\boldsymbol{y}_k = \boldsymbol{H}\boldsymbol{x}_k. \tag{13}$$

Here, $\boldsymbol{y} = \text{column}(\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(N_c)})_{M \times 1}$, and $\boldsymbol{H} = \text{blkdiag}(\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{N_c})_{M \times N}$, where $M = N_c M_l$.

From above, (11) and (13) form the state-space model, which characterizes the propagation and measurement of a LiB pack's thermal dynamics. With this model, let us conduct state estimation to reconstruct the temperature field.

III. KF-Based Temperature Field Estimation

This section develops KF-based approaches to achieve reconstruction of the temperature field. In the following, the CKF will be introduced first, and then, its distributed versions presented and analyzed in detail.

A. Centralized Kalman Filtering

Replicate (11) and (13) with noise terms added as follows:

$$\begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{G}_k \boldsymbol{u}_k + \boldsymbol{w}_k \\ \boldsymbol{y}_k = \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{v}_k. \end{cases} \tag{14}$$

Here, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are added to account for the process noise and measurement noise that exist in the thermal dynamic processes of a LiB pack. They are assumed to be zero-mean Gaussian white noises with covariances of $\boldsymbol{Q} \geq 0$ and $\boldsymbol{R} > 0$, respectively. Suppose the initial guess of $\boldsymbol{x}_0$ is $\hat{\boldsymbol{x}}_{0|0}$, with an estimation error covariance $\boldsymbol{P}_{0|0}$. Then, application of the standard CKF to (14) can be performed at each time instant. This procedure consists of two steps: *prediction* and *update*.

When the state estimate $\hat{\boldsymbol{x}}_{k-1|k-1}$ is generated, the one-step-forward prediction can be made through

$$\hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F}_{k-1}\hat{\boldsymbol{x}}_{k-1|k-1} + \boldsymbol{G}_{k-1}\boldsymbol{u}_{k-1} \tag{15}$$

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{F}_{k-1}\boldsymbol{P}_{k-1|k-1}\boldsymbol{F}^\top_{k-1} + \boldsymbol{Q}. \tag{16}$$

When the new measurement $\boldsymbol{y}_k$ becomes available, the update step can be performed as follows to correct the prediction:

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k-1}) \tag{17}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{H}^\top(\boldsymbol{H}\boldsymbol{P}_{k|k-1}\boldsymbol{H}^\top + \boldsymbol{R})^{-1} \tag{18}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{H}\boldsymbol{P}_{k|k-1}. \tag{19}$$

The CKF executes the above steps recursively over time to generate the state estimate at each time instant. However, it is not suited to estimate the temperature for a LiB pack, because the high-dimensional thermal model of the LiB pack implies considerable computation (the detailed computational complexity analysis will be given in Section III-D). Hence, the CKF will be distributed next to reduce the computational cost.

B. Distributed Kalman Filtering

Rather than estimate the global state in a centralized manner, the DKF will consider the pack system as a combination of multiple cell-based subsystems and run a series of local KFs in parallel, each one corresponding to a cell. Because of the mutual influence between the cells and their thermal behavior, the local KFs will exchange information according to the existing communication topology to accomplish the estimation. The local estimates, when collected and put together, will comprise a complete picture of the entire pack's temperature field.

Consider a LiB pack composed of N_c cells wired in series, which are numbered in order from 1 to N_c. For cell l and i, they are said to be neighbors if they are adjacent. The neighborhood of l, $\mathcal{N}_l$, is defined as the set of its neighbor cells, and in this setting, $\mathcal{N}_l = \{l-1, l+1\}$. It is assumed here that cell l can communicate with $\mathcal{N}_l$. According to (11) and (12), the state-space equation for cell l can be written as

$$\begin{cases} \boldsymbol{x}^{(l)}_{k+1} = \boldsymbol{F}_{ll,k}\boldsymbol{x}^{(l)}_k + \sum\limits_{i \in \mathcal{N}_l} \boldsymbol{F}_{li,k}\boldsymbol{x}^{(i)}_k + \boldsymbol{G}_{l,k}\boldsymbol{u}_k + \boldsymbol{w}^{(l)}_k \\ \boldsymbol{y}^{(l)}_k = \boldsymbol{H}_l\boldsymbol{x}^{(l)}_k + \boldsymbol{v}^{(l)}_k \end{cases} \tag{20}$$

where cellwise decomposition is also applied to $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$. Note that the evolution of cell l's state is not only self-driven but also affected by the neighboring cells. All cells yet share the same $\boldsymbol{u}_k$ values, because the serial connection implies the same charging/discharging current across the circuit. In addition, each cell is only aware of its own temperature measurements.

For the above cell l-based subsystem, the CKF approach can be adjusted for local state estimation. This can be attained by applying the prediction–update procedure in analogy to (15)–(19). Specifically, the prediction can be given by

$$\hat{\boldsymbol{x}}^{(l)}_{k|k-1} = \boldsymbol{F}_{ll,k-1}\hat{\boldsymbol{x}}^{(l)}_{k-1|k-1} + \sum_{i \in \mathcal{N}_l} \boldsymbol{F}_{li,k-1}\hat{\boldsymbol{x}}^{(i)}_{k-1|k-1} + \boldsymbol{G}_{l,k-1}\boldsymbol{u}_{k-1} \tag{21}$$

$$\boldsymbol{P}^{(l)}_{k|k-1} = \boldsymbol{F}_{ll,k-1}\boldsymbol{P}^{(l)}_{k-1|k-1}\boldsymbol{F}^\top_{ll,k-1} + \boldsymbol{Q}_l. \tag{22}$$

Here, cell l's state prediction, $\hat{\boldsymbol{x}}_{k|k-1}^{(l)}$, depends on not only its own but also its neighbors' state estimates from the previous time instant. On its arrival, $\boldsymbol{y}_k^{(l)}$ can be used to update $\hat{\boldsymbol{x}}_{k|k-1}^{(l)}$

$$\hat{\boldsymbol{x}}_{k|k}^{(l)} = \hat{\boldsymbol{x}}_{k|k-1}^{(l)} + \boldsymbol{K}_k^{(l)}\big(\boldsymbol{y}_k^{(l)} - \boldsymbol{H}_l\hat{\boldsymbol{x}}_{k|k-1}^{(l)}\big) \tag{23}$$

$$\boldsymbol{K}_k^{(l)} = \boldsymbol{P}_{k|k-1}^{(l)}\boldsymbol{H}_l^\top\big(\boldsymbol{H}_l\boldsymbol{P}_{k|k-1}^{(l)}\boldsymbol{H}_l^\top + \boldsymbol{R}_l\big)^{-1} \tag{24}$$

$$\boldsymbol{P}_{k|k}^{(l)} = \boldsymbol{P}_{k|k-1}^{(l)} - \boldsymbol{K}_k^{(l)}\boldsymbol{H}_l\boldsymbol{P}_{k|k-1}^{(l)}. \tag{25}$$

Note that no information exchange with neighbors is required in the update step. Here, (21)–(25) constitute the DKF algorithm. Running the DKF in parallel for each cell leads to local temperature field estimation. Combining the local estimation, one can obtain a full view of the pack's temperature field.

Compared with the CKF, the DKF algorithm involves approximation. First, each cell only has a local rather than global knowledge of the system's dynamic behavior, implying information loss inherent in each local DKF. To see this, the nominal prediction error covariance $\boldsymbol{P}_{k|k-1}^{(l)}$, differing from the CKF, evolves only from its predecessor without fusion of the counterparts of the other cells, as shown in (22). This shows that $\boldsymbol{P}_{k|k-1}^{(l)}$ is only approximate to the true prediction error covariance. Second, part of the approximation is made to reduce the communication and computation costs. Looking at (22) again, one can see that a local cell does not consider its neighbors in its forward propagation of its prediction error covariance. This will obviate the need for the exchange of the estimation error covariance between neighboring cells and, further, the local computational effort. However, the approximation will not seriously compromise the estimation accuracy. Since $\boldsymbol{F}$ is diagonally dominant and $\boldsymbol{H}$ block-diagonal due to the pack's serial connection, $\boldsymbol{P}$ will also be diagonally dominant. The self-propagation of the local estimation error covariance, as a result, will not bring much loss of estimation accuracy.

C. Steady-State Distributed Kalman Filtering

It is identified that, if some mild reduction is introduced for the considered thermal model, we can obtain another DKF approach with much higher computational efficiency. To be specific, consider the heat generation equation (2). Many studies in the literature suggest that its second term often has a negligible magnitude in comparison with the first term and thus can be ignored [8], [27]. With this simplification, (2) can be reduced as $q = I(U_{\mathrm{ocv}} - U_{\mathrm{t}})/V_{\mathrm{co}}$. It is then found that $\boldsymbol{F}$ becomes time-invariant in this case, which will allow us to develop a more computationally efficient DKF for temperature field reconstruction. The development is as follows.

Before proceeding further, we make an assumption at first.

Assumption 1: The pair $(\boldsymbol{F}_{ll}, \boldsymbol{H}_l)$ is detectable and the pair $(\boldsymbol{F}_{ll}, \boldsymbol{Q}_l^{\frac{1}{2}})$ stabilizable for $l \in \{1, \ldots, N_c\}$.

Given Assumption 1, the DKF algorithm in (21)–(25) will gradually achieve steady state. Specifically, $\boldsymbol{P}_{k|k-1}^{(l)}$ will converge to a unique stabilizing solution, $\bar{\boldsymbol{P}}^{(l)}$, of the discrete algebraic Riccati equation

$$\boldsymbol{X} = \boldsymbol{F}_{ll}\boldsymbol{X}\boldsymbol{F}_{ll}^\top - \boldsymbol{F}_{ll}\boldsymbol{X}\boldsymbol{H}_l^\top\big(\boldsymbol{H}_l\boldsymbol{X}\boldsymbol{H}_l^\top + \boldsymbol{R}_l\big)^{-1}\boldsymbol{H}_l\boldsymbol{X}\boldsymbol{F}_{ll}^\top + \boldsymbol{Q}_l$$

where $\boldsymbol{X}$ is an unknown symmetric positive-definite matrix. The gain matrix $\boldsymbol{K}_k^{(l)}$ in (24) then will approach a fixed point

$$\bar{\boldsymbol{K}}^{(l)} = \bar{\boldsymbol{P}}^{(l)}\boldsymbol{H}_l^\top\big(\boldsymbol{H}_l\bar{\boldsymbol{P}}^{(l)}\boldsymbol{H}_l^\top + \boldsymbol{R}_l\big)^{-1} \tag{26}$$

which can ensure $\boldsymbol{F}_{ll}(\boldsymbol{I}_l - \bar{\boldsymbol{K}}^{(l)}\boldsymbol{H}_l)$ to be stable [30]. With fixed $\bar{\boldsymbol{P}}^{(l)}$ and $\bar{\boldsymbol{K}}^{(l)}$, the state prediction and update can be accomplished more efficiently

$$\hat{\boldsymbol{x}}_{k|k-1}^{(l)} = \boldsymbol{F}_{ll}\hat{\boldsymbol{x}}_{k-1|k-1}^{(l)} + \sum_{i\in\mathcal{N}_l}\boldsymbol{F}_{li}\hat{\boldsymbol{x}}_{k-1|k-1}^{(i)} + \boldsymbol{G}_l\boldsymbol{u}_{k-1} \tag{27}$$

$$\hat{\boldsymbol{x}}_{k|k}^{(l)} = \hat{\boldsymbol{x}}_{k|k-1}^{(l)} + \bar{\boldsymbol{K}}^{(l)}\big(\boldsymbol{y}_k^{(l)} - \boldsymbol{H}_l\hat{\boldsymbol{x}}_{k|k-1}^{(l)}\big) \tag{28}$$

which together form the steady-state DKF (SS-DKF) algorithm. It is seen that the SS-DKF does not maintain the estimation error covariance and that its gain matrix can be computed offline prior to the estimation run. Although this incurs certain sacrifice of estimation accuracy, it presents much appeal from a computational perspective. Before moving on to the computational complexity analysis in Section III-D, the stability of the SS-DKF algorithm is examined in the remainder of this section.

We define the real state error $\boldsymbol{e}_k = \hat{\boldsymbol{x}}_{k|k} - \boldsymbol{x}_k$. For the real state error $\boldsymbol{e}_k^{(l)}$ at cell l, combining (27) and (28), we have

$$\begin{aligned}\boldsymbol{e}_k^{(l)} = {} & (\boldsymbol{I} - \bar{\boldsymbol{K}}^{(l)}\boldsymbol{H}_l)\boldsymbol{F}_{ll}\boldsymbol{e}_{k-1}^{(l)} + \sum_{i\in\mathcal{N}_l}(\boldsymbol{I} - \bar{\boldsymbol{K}}^{(l)}\boldsymbol{H}_l)\boldsymbol{F}_{li}\boldsymbol{e}_{k-1}^{(i)} \\ & - (\boldsymbol{I} - \bar{\boldsymbol{K}}^{(l)}\boldsymbol{H}_l)\boldsymbol{w}_{k-1}^{(l)} + \bar{\boldsymbol{K}}^{(l)}\boldsymbol{v}_k^{(l)}.\end{aligned} \tag{29}$$

Aggregating $\boldsymbol{e}_k^{(l)}$ for $l \in \{1, \ldots, N_c\}$ together yields

$$\boldsymbol{e}_k = (\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}\boldsymbol{e}_{k-1} - (\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{w}_{k-1} + \bar{\boldsymbol{K}}\boldsymbol{v}_k \tag{30}$$

where $\bar{\boldsymbol{K}} = \mathrm{blkdiag}\{\bar{\boldsymbol{K}}^{(1)}, \ldots, \bar{\boldsymbol{K}}^{(N_c)}\}$. The stability of the SS-DKF algorithm is stated as follows.

Theorem 1: Let $\boldsymbol{A} = (\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}$ and $\boldsymbol{B} = (\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{Q}(\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})^\top + \bar{\boldsymbol{K}}\boldsymbol{R}\bar{\boldsymbol{K}}^\top$. If $\boldsymbol{A}$ is stable, the true error covariance $\boldsymbol{\Sigma}_k = \mathbf{E}[\boldsymbol{e}_k\boldsymbol{e}_k^\top]$ will converge to the unique solution of the discrete-time Lyapunov equation $\boldsymbol{\Sigma} = \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^\top + \boldsymbol{B}$.

Proof: It is seen that the propagation of $\boldsymbol{\Sigma}_k$ is governed by $\boldsymbol{\Sigma}_k = \boldsymbol{A}\boldsymbol{\Sigma}_{k-1}\boldsymbol{A}^\top + \boldsymbol{B}$. Since $\boldsymbol{A}$ is stable, $\lim_{k\to\infty}\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$ (see [31, Ch. 3.3]). □

From above, Assumption 1 lays the foundation for the derivation of the SS-DKF, and Theorem 1 indicates that a stable $(\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}$ can guarantee the stability of the SS-DKF algorithm. Yet, a question then is: will Assumption 1 and the stability of $(\boldsymbol{I} - \bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}$ hold for the LiB pack model?

An examination is given as follows. First, consider Assumption 1. Note that the thermal physics implies that the model established in Section II is stable if the numerical stability condition is satisfied in discretization. Hence, $\boldsymbol{F}$ will be stable. Next, we partition $\boldsymbol{F}$ into the following form:

$$\boldsymbol{F} = \boldsymbol{F}_{\mathrm{d}} + \boldsymbol{F}_{\mathrm{od}} \tag{31}$$

where the subscripts d and od denote diagonal blocks and off-diagonal blocks, and $\boldsymbol{F}_{\mathrm{d}} = \mathrm{blkdiag}\{\boldsymbol{F}_{11}, \ldots, \boldsymbol{F}_{N_c,N_c}\}$. According to [32, Corollary 5.6.14], one will have

$$\rho(\boldsymbol{F}_{\mathrm{od}}) = \lim_{k\to\infty}\big\|\boldsymbol{F}_{\mathrm{od}}^k\big\|^{1/k}. \tag{32}$$

TABLE I
ARITHMETIC OPERATION REQUIREMENTS OF THE CKF, DKF, AND SS-DKF ALGORITHMS

Algorithm	Number of multiplications	Number of additions	Complexity
CKF	$3N^3+2N^2M+2NM^2+M^3+N^2+2NM+2N$	$3N^3+2N^2M+2NM^2+M^3-N^2+N$	$O(N^3)$
DKF	$N_c\left(3N_l^3+2N_l^2M_l+2N_lM_l^2+M_l^3+3N_l^2+2N_lM_l+2N_l\right)$	$N_c\left(3N_l^3+2N_l^2M_l+2N_lM_l^2+M_l^3+N_l^2+N_l\right)$	$O(N_cN_l^3)$
SS-DKF	$N_c\left(2N_l^2+2N_lM_l+2N_l\right)$	$N_c\left(3N_l^2+2N_lM_l+N_l\right)$	$O(N_cN_l^2)$

It is interesting to note that $\boldsymbol{F}_{\text{od}}^2=\boldsymbol{0}$ in this application due to the serial structure of the LiB pack, and hence, $\rho(\boldsymbol{F}_{\text{od}})=0$ in this case. Invoking [32, Lemma 5.6.10], there exists a matrix norm $\|\cdot\|_*$ for any given $\epsilon>0$ such that

$$\rho(\boldsymbol{F}_{\text{d}})\le\|\boldsymbol{F}_{\text{d}}\|_*\le\rho(\boldsymbol{F}_{\text{d}})+\epsilon. \tag{33}$$

Since it satisfies the triangle inequality, then

$$\|\boldsymbol{F}_{\text{d}}\|_*\le\|\boldsymbol{F}\|_*+\|\boldsymbol{F}_{\text{od}}\|_*\le(\rho(\boldsymbol{F})+\epsilon_1)+(\rho(\boldsymbol{F}_{\text{od}})+\epsilon_2)$$

where $\epsilon_1>0$ and $\epsilon_2>0$. It follows from $\rho(\boldsymbol{F}_{\text{od}})=0$ that:

$$\|\boldsymbol{F}_{\text{d}}\|_*\le\rho(\boldsymbol{F})+\epsilon_1+\epsilon_2. \tag{34}$$

Thus, one can always find ϵ_1 and ϵ_2 to prove that $\boldsymbol{F}_{\text{d}}$ is stable, thus validating Assumption 1.

Now, consider the stability of $(\boldsymbol{I}-\bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}$. For notational simplicity, we denote $\boldsymbol{I}-\bar{\boldsymbol{K}}\boldsymbol{H}$ as $\tilde{\boldsymbol{I}}$. Then, the objective is to show that $\tilde{\boldsymbol{I}}\boldsymbol{F}$ is stable. Recalling the matrix norm $\|\cdot\|_*$ in (33), it is also submultiplicative and implies

$$\begin{aligned}\|\tilde{\boldsymbol{I}}\boldsymbol{F}\|_*&\le\|\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}}\|_*+\|\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{od}}\|_*\\&\le\|\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}}\|_*+\|\tilde{\boldsymbol{I}}\|_*\|\boldsymbol{F}_{\text{od}}\|_*\\&\le(\rho(\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}})+\epsilon_3)+(\rho(\tilde{\boldsymbol{I}})+\epsilon_4)(\rho(\boldsymbol{F}_{\text{od}})+\epsilon_5)\\&\le\rho(\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}})+\epsilon_3+\rho(\tilde{\boldsymbol{I}})\epsilon_5+\epsilon_4\epsilon_5\end{aligned} \tag{35}$$

where $\epsilon_3>0$, $\epsilon_4>0$, and $\epsilon_5>0$. Then, because $\boldsymbol{F}_{ll}(\boldsymbol{I}_l-\bar{\boldsymbol{K}}^{(l)}\boldsymbol{H}_l)$ is stable, the matrix $\boldsymbol{F}_d\tilde{\boldsymbol{I}}$ is stable. Following that $\lim_{k\to\infty}(\boldsymbol{F}_{\text{d}}\tilde{\boldsymbol{I}})^k=\boldsymbol{0}$, $(\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}})^k$ can be constructed as $\tilde{\boldsymbol{I}}(\boldsymbol{F}_{\text{d}}\tilde{\boldsymbol{I}})^{k-1}\boldsymbol{F}_{\text{d}}$, such that $\lim_{i\to\infty}(\tilde{\boldsymbol{I}}\boldsymbol{F}_{\text{d}})^k=\boldsymbol{0}$. Therefore, it is always possible to find ϵ_3, ϵ_4, and ϵ_5 to make the right-hand side of (35) smaller than 1. Subsequently, $\rho(\tilde{\boldsymbol{I}}\boldsymbol{F})<1$ and $(\boldsymbol{I}-\bar{\boldsymbol{K}}\boldsymbol{H})\boldsymbol{F}$ is stable.

Remark 2 (Extension to Thermal Runaway Detection): The above develops the DKF and SS-DKF to reconstruct the temperature field of a battery pack. They can be used as a tool to monitor the spatially distributed thermal behavior critical for a battery pack's safety. An extension of them to detect thermal runaway can be hopefully made. An idea is to consider the thermal runaway as an unknown disturbance that abruptly appears and applies to the model in (14). Then, the thermal runaway detection can be formulated as the problem of disturbance detection. KF-based approaches have been studied extensively for disturbance detection in the literature (see [33]) and can be potentially exploited here. Combining this idea and the design in this brief, we can promisingly build DKF-based approaches for thermal runaway detection. This will be an important part of our future work.

D. Computational Complexity Analysis

As aforementioned, distributing the CKF is to improve the computational efficiency toward enabling real-time reconstruction of a LiB pack's temperature field. In this section, the CKF, DKF, and SS-DKF algorithms are analyzed and compared in terms of computational complexity. The analysis is to determine the number of arithmetic operations needed by each algorithm. To proceed further, we consider the basic matrix operations. For two $n\times n$ matrices, their addition involves n^2 elementary additions, and their multiplication involves n^3 elementary multiplications and $(n-1)n^2$ elementary additions. The inverse of an $n\times n$ matrix requires n^3 elementary multiplications and n^3 elementary additions. The complexity of each algorithm can be assessed by summing up all the arithmetic operations required at each time instant, as summarized in Table I.

It is demonstrated in Table I that the CKF has the heaviest computation at $O(N^3)$, which increases cubically with the size of the state space of the entire pack. This also implies that, when a pack has more cells, the computation would rise cubically with the cell number. Computational complexity at such a level is unaffordable for a real-world onboard computing platform. By contrast, the DKF is much more efficient. Given that $N=N_cN_l$, the DKF's arithmetic operations at $O(N_cN_l^3)$ are only about one (N_c^2)th of the CKF's. In addition, with the computation increasing only linearly with the cell number, the DKF well lends itself to parallel processing, where the estimation for each cell is performed on a separate microprocessor at a complexity $O(N_l^3)$. In this scenario, an increase in the cell number will not add cost to the existing microprocessors. The SS-DKF unsurprisingly is the most computationally competitive. Its complexity at $O(N_cN_l^2)$ is even one order less than that of the DKF. Just like the DKF, it is also well suited for parallel processing-based execution.

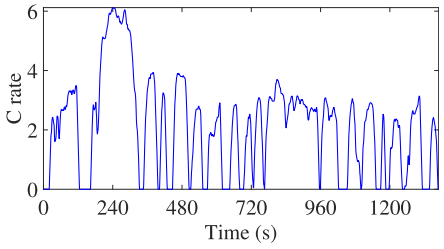

Fig. 3. Discharging current profile based on UDDS.

IV. NUMERICAL SIMULATION

In this section, numerical simulation with a practical LiB pack is offered. The simulation is performed using MATLAB.

A. Simulation Setting

Consider a LiB pack that consists of three large-format high-capacity prismatic LiB cells connected and stacked

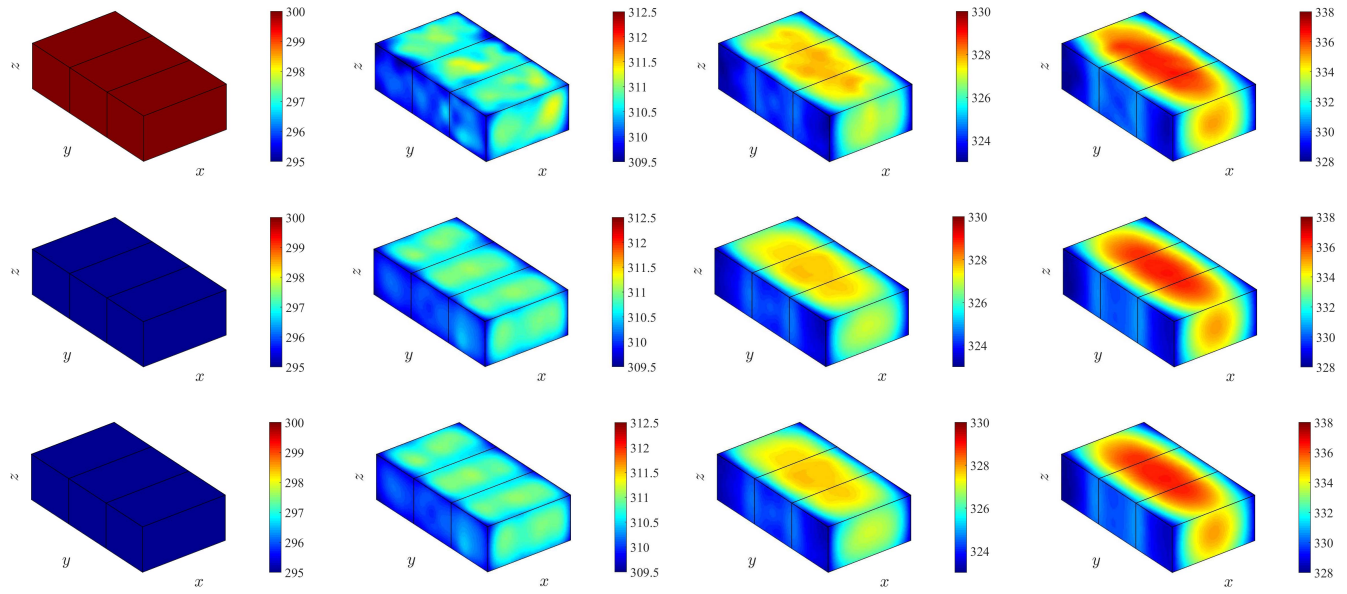

Fig. 4. First row: real temperature fields through time for the LiB pack discharged at UDDS-based current loads. Second row: CKF-based reconstruction. Third row: DKF-based reconstruction. (a) $t = 0$ s. (b) $t = 240$ s (c) $t = 480$ s. (d) $t = 1200$ s.

in series. Here, the cells are the same ones as in [17]. Each cell has a capacity of 185.3 Ah and is 19.32 cm long, 10.24 cm wide, and 10.24 cm high. As mentioned in Section II, the cell has two portions: the core region and the metallic case. The core region is 19.08 cm long, 10 cm wide, and 10 cm high, housing three hundred smaller cell units in parallel. The key parameters of the LiB cell can refer to [17]. In the simulation, it is assumed that the LiB pack operates in an environment with temperature maintained at 300 K. The convective heat transfer coefficient, emissivity on pack surface, and entropic heat transfer coefficient dU_{ocv}/dT are set to be 30 W/(m^2·K), 0.25, and 0.00022 VK^{-1} [17], respectively. The battery pack is discharged using a time-varying current profile, which is shown in Fig. 3 and derived from the Urban Dynamometer Driving Schedule (UDDS) [34]. Sensors are mounted on the battery pack as shown in Fig. 2, i.e., five sensors are placed on each cell–air interface of a cell. Such a placement is straightforward and easy to implement. Associated with this, an intriguing question is how to optimally deploy the sensors toward achieving satisfactory estimation performance with a minimum number of sensors. While some results are reported in the literature (see [15]), further research is still required to fully address this question.

The thermal dynamics of the considered pack can be characterized by the PDE-based model in Section II. Each cell is gridded in space with $m = 9$, $n = 5$, and $p = 5$ and in time with $\Delta t = 1$ s. In general, one can increase m, n, and p and reduce Δt to increase the accuracy of simulation. This, however, will increase computational cost. Another risk lies in numerical instability, which can be caused if the selected m, n, p, and Δt fail to satisfy certain conditions [28]. To find a satisfactory set, one can consider a few candidates. She/he can first check the numerical stability for each set using the conditions in [28]. Then, run the simulation for the sets that pass the check, and choose the set that leads to acceptable

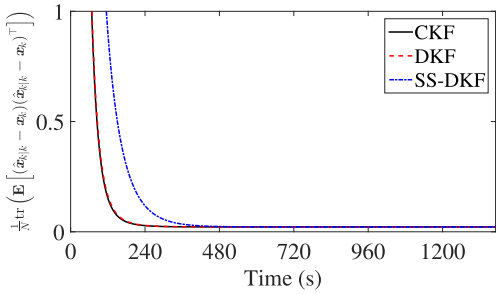

Fig. 5. Evaluation of the estimation error over time when the simplified heat generation model is used.

accuracy with minimum computational cost. This process understandably may require repeated trial effort.

In the simulation, the pack's initial temperature is 300 K, the same with the ambient temperature. Yet, for the purpose of illustrating the estimation, the initial guess is 295 K in the simulation. The noise covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ are chosen as $\boldsymbol{Q} = 0.05^2 \cdot \boldsymbol{I}$ and $\boldsymbol{R} = 0.3^2 \cdot \boldsymbol{I}$, respectively.

B. Simulation Results

The real temperature field is obtained by running the state equation of (14) using MATLAB with the effects of process noise included. Sensor-based measurements are obtained according to the measurement equation of (14), which are subjected to sensor noise. Using the measurements and based on the model, the CKF and DKF are applied to reconstruct the temperature field. The simulation results are summarized in Fig. 4. The first row shows the real temperature field that evolves over time. Here, trilinear spatial interpolation is used to generate spatially continuous temperature fields. As is shown, the pack sees an obvious temperature rise, despite the convection cooling and only three cells. In addition, it can be easily found that the temperature differs spatially across the pack, with a high gradient buildup at the end of the simulation.

The second and third rows show the reconstructed temperature fields using the CKF and DKF, respectively. It is seen that, although the initial guess differs from the truth, both of them can generate temperature field estimation that gradually catches up with the truth. Next, let us further examine the SS-DKF approach. In this case, the heat generation model (2) is simplified. In order to save space, the visual demonstration of the temperature field estimation is omitted. Yet a comparison of accuracy is provided in Fig. 5, where the metric $(1/N)\text{trace}(\mathbf{E}[(\hat{\boldsymbol{x}}_{k|k}-\boldsymbol{x}_k)(\hat{\boldsymbol{x}}_{k|k}-\boldsymbol{x}_k)^\top])$ is the averaged trace of the estimation error covariance. It is observed that the SS-DKF is less accurate in the initial stage compared with the CKF and DKF. However, it can achieve approximately the same accuracy after about 400 s. Given this result and its superior computational efficiency, the SS-DKF can be a worthy tool in practice.

V. Conclusion

Thermal monitoring is crucial for ensuring the safety of LiBs. This brief is proposed to achieve 3-D temperature field reconstruction, which is beneficial and necessary but still absent in the literature. A thermal model is presented first to capture the thermal dynamics of a LiB pack, which is based on heat transfer and energy balance analysis. Based on the model, the well-known KF approach is distributed to achieve global temperature field reconstruction through localized estimation, reducing the computational complexity remarkably. A DKF algorithm, which well fits the considered problem, is offered, and its steady-state version, SS-DKF, would require even less computation time. A detailed computational complexity analysis highlights the advantages of the distributed estimation approaches. Simulation with a LiB pack based on genuine cells demonstrates the effectiveness of the proposed DKF and SS-DKF algorithms. A diversity of work will be performed in the future along this brief, including temperature field reconstruction based on more sophisticated models, DKF-based thermal runaway detection, and experimental validation.

References

[1] C. D. Rahn and C.-Y. Wang, *Battery Systems Engineering*. Hoboken, NJ, USA: Wiley, 2013.

[2] Y. Wang, H. Fang, L. Zhou, and T. Wada, "Revisiting the state-of-charge estimation for lithium-ion batteries: A methodical investigation of the extended Kalman filter approach," *IEEE Control Syst.*, vol. 37, no. 4, pp. 73–96, Aug. 2017.

[3] H. Fang, Y. Wang, and J. Chen, "Health-aware and user-involved battery charging management for electric vehicles: Linear quadratic strategies," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 911–923, May 2017.

[4] D. Bernardi, E. Pawlikowski, and J. Newman, "A general energy balance for battery systems," *J. Electrochem. Soc.*, vol. 132, no. 1, pp. 5–12, 1985.

[5] T. M. Bandhauer, S. Garimella, and T. F. Fuller, "A critical review of thermal issues in lithium-ion batteries," *J. Electrochem. Soc.*, vol. 158, no. 3, pp. R1–R25, 2011.

[6] C. Forgez, D. V. Do, G. Friedrich, M. Morcrette, and C. Delacourt, "Thermal modeling of a cylindrical $LiFePO_4$/graphite lithium-ion battery," *J. Power Sources*, vol. 195, no. 9, pp. 2961–2968, 2010.

[7] X. Lin *et al.*, "Online parameterization of lumped thermal dynamics in cylindrical lithium ion batteries for core temperature estimation and health monitoring," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 5, pp. 1745–1755, Sep. 2013.

[8] X. Lin *et al.*, "A lumped-parameter electro-thermal model for cylindrical batteries," *J. Power Sources*, vol. 257, pp. 1–11, Jul. 2014.

[9] J. Sun *et al.*, "Online internal temperature estimation for lithium-ion batteries based on Kalman filter," *Energies*, vol. 8, no. 5, pp. 4400–4415, 2015.

[10] M. Debert, G. Colin, G. Bloch, and Y. Chamaillard, "An observer looks at the cell temperature in automotive battery packs," *Control Eng. Pract.*, vol. 21, no. 8, pp. 1035–1042, 2013.

[11] Y. Xiao, "Model-based virtual thermal sensors for lithium-ion battery in EV applications," *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 3112–3122, May 2015.

[12] C. Park and A. K. Jaura, "Dynamic thermal model of Li-ion battery for predictive behavior in hybrid and fuel cell vehicles," in *Proc. Future Transp. Technol. Conf.*, 2003.

[13] R. Mahamud and C. Park, "Reciprocating air flow for Li-ion battery thermal management to improve temperature uniformity," *J. Power Sources*, vol. 196, no. 13, pp. 5685–5696, 2011.

[14] X. Lin, H. E. Perez, J. B. Siegel, A. G. Stefanopoulou, Y. Ding, and M. P. Castanier, "Parameterization and observability analysis of scalable battery clusters for onboard thermal management," *Oil Gas Sci. Technol.*, vol. 68, no. 1, pp. 165–178, 2011.

[15] X. Lin, A. G. Stefanopoulou, J. B. Siegel, and S. Mohan, "Temperature estimation in a battery string under frugal sensor allocation," in *Proc. ASME Dyn. Syst. Control Conf., Amer. Soc. Mech. Eng.*, 2014, p. V001T19A006.

[16] Y. Shi, K. Smith, E. Wood, and A. Pesaran, "A multi-node thermal system model for lithium-ion battery packs," in *Proc. Amer. Control Conf.*, Jul. 2015, pp. 723–727.

[17] S. C. Chen, C. C. Wan, and Y. Y. Wang, "Thermal analysis of lithium-ion batteries," *J. Power Sources*, vol. 140, no. 1, pp. 111–124, 2005.

[18] S.-C. Chen, Y.-Y. Wang, and C.-C. Wan, "Thermal analysis of spirally wound lithium batteries," *J. Electrochem. Soc.*, vol. 153, no. 4, pp. A637–A648, 2006.

[19] G. Guo, B. Long, B. Cheng, S. Zhou, P. Xu, and B. Cao, "Three-dimensional thermal finite element modeling of lithium-ion battery in thermal abuse application," *J. Power Sources*, vol. 195, no. 8, pp. 2393–2398, 2010.

[20] P. M. Gomadam, J. W. Weidner, R. A. Dougal, and R. E. White, "Mathematical modeling of lithium-ion and nickel battery systems," *J. Power Sources*, vol. 110, no. 2, pp. 267–284, 2002.

[21] C. Y. Wang and V. Srinivasan, "Computational battery dynamics (CBD)—Electrochemical/thermal coupled modeling and multi-scale modeling," *J. Power Sources*, vol. 110, no. 2, pp. 364–376, 2002.

[22] V. Srinivasan and C. Wang, "Analysis of electrochemical and thermal behavior of Li-ion cells," *J. Electrochem. Soc.*, vol. 150, no. 1, pp. A98–A106, 2003.

[23] K. Smith and C.-Y. Wang, "Power and thermal characterization of a lithium-ion battery pack for hybrid-electric vehicles," *J. Power Sources*, vol. 160, no. 1, pp. 662–673, 2006.

[24] G.-H. Kim, A. Pesaran, and R. Spotnitz, "A three-dimensional thermal abuse model for lithium-ion cells," *J. Power Sources*, vol. 170, no. 2, pp. 476–489, 2007.

[25] C.-S. Hsieh and F.-C. Chen, "Optimal solution of the two-stage Kalman estimator," *IEEE Trans. Autom. Control*, vol. 44, no. 1, pp. 194–199, Jan. 1999.

[26] S. Al Hallaj, H. Maleki, J.-S. Hong, and J. R. Selman, "Thermal modeling and design considerations of lithium-ion batteries," *J. Power Sources*, vol. 83, nos. 1–2, pp. 1–8, 1999.

[27] Y. Kim, S. Mohan, J. B. Siegel, A. G. Stefanopoulou, and Y. Ding, "The estimation of temperature distribution in cylindrical battery cells under unknown cooling conditions," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 6, pp. 2277–2286, Nov. 2014.

[28] T. L. Bergman, F. P. Incropera, D. P. DeWitt, and A. S. Lavine, *Fundamentals of Heat and Mass Transfer*, 6th ed. Hoboken, NJ, USA: Wiley, 2006.

[29] N. Tian, H. Fang, and Y. Wang. (2017). "Three-dimensional temperature field reconstruction for a lithium-ion battery pack: A distributed Kalman filtering approach." [Online]. Available: https://arxiv.org/abs/1709.08819

[30] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2015.

[31] P. R. Varaiya and P. Kumar, *Stochastic Systems: Estimation, Identificat. Adapt. Control*, 1st ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1986.

[32] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 1st ed. Cambridge, U.K.: Cambridge Univ. Press, 1985.

[33] F. Gustafsson, *Adaptive Filtering and Change Detection*, vol. 1. New York, NY, USA: Wiley, 2000.

[34] *The EPA Urban Dynamometer Driving Schedule (UDDS)*. Accessed: Jul. 20, 2017. [Online]. Available: https://www.epa.gov/sites/production/files/2015-10/uddscol.txt