Observer and Controller Designs for Lithium Ion Batteries

Zhilong Liu, Niloofar Shahmohammadhamedani

***Abstract*—Lithium ion battery is a popular type of clean energy storage that is widely used in industry. In this project, we study the observer and controller design of a cylindrical electro-thermal lithium ion battery model. The model, consisting of a two-state equivalent circuit model and a two-state thermal model, captures the state of charge (SOC), terminal voltage and the battery core temperature. We first study the observability of the model and present two observer designs, namely an Extended Kalman Filter (EKF) observer and an Ensemble Kalman Filter (EnKF) observer, to estimate the core temperature. The second part of the project is to implement three controllers on fast-charging.Controllers are Dynamic Programming and two tracking controllers, sliding mode and feedback linearization,**

***Index Terms*—Observer, optimal control, nonlinear control, Lithium battery.**

I. Introduction

A. Motivation & Background

Battery technology is growing rapidly in our daily life. From consumer electronics to Electric Vehicles (EV), we are taking advantage of the benefits of batteries. As an example, lithium ion batteries are widely used in Hybrid electric vehicles (HEV) and plug-in electric vehicles (PHEV). They are considered as a promising solution to carbon emission in the automotive industry. They are not only better alternatives for gasoline-based vehicles, but also widely used in vehicle-to-grid (V2G) type of application to generate clean electricity. Due to the high demand, batteries have been a subject under active research.

Among all types of batteries, lithium ion batteries are the most widely used due to their advantages in energy/power density and charge efficiency. One of the most important characteristics of a battery is lifetime, which is mostly constrained by the charging and discharging cycles. In V2G applications, frequent charging and discharging are expected, which may greatly reduce the battery life. Thus, optimizing the charging and discharging control strategy is essential to lengthen the battery endurance.

Two important factors for charging and discharging the batteries are 1) Making sure the core temperature is within the desired limits and 2) charging the battery to a desired State Of Charge (SOC). The core temperature is important to not only because of safety concerns, but also since excessive heat can damage and shorten the battery life. However, we can't obtain the battery core temperature by direct measurements.

To address this issue, we first built non-linear observers to estimate the missing core temperature, and compare the estimates with the true values. To test the robustness of the observer design, the measurements will be contaminated with white Gaussian noise. Secondly we will implement multiple controller designs to achieve a desired SOC.

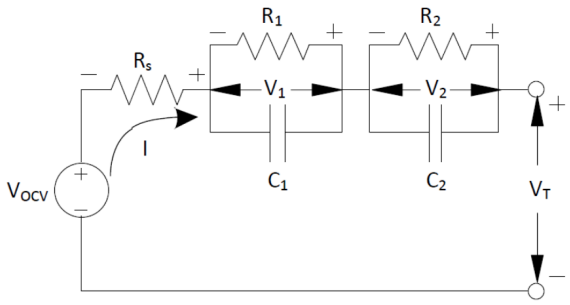


Fig. 1: The three-state equivalent circuit model.

B. Relevant Literature

The electro-thermal dynamic model is adopted from [1]. For observer design, extended Kalman filter (EKF) [2] and ensemble Kalman filter (EnKF) [3] will be explored. For control design, we first try optimal control via dynamic programming [4]. In addition, nonlinear control methods such as feedback linearization and sliding control will also be investigated [5], [6].

C. Focus of this Study

The purpose of this project is to study the observability, non-linear observer design, and control strategies on a Li-ion battery model. We implement observer designs on estimating the core temperature of the battery to prevent overheating. Secondly we design non-linear controlers to control the SOC of the battery while keeping a reasonable charging and discharging rate.

II. Technical Description

A. Modeling

As mentioned earlier, we are using a coupled electro-thermal model. We will discuss a 3-state electrical model, a two-state thermal model, and finally combine them into a five-state coupled model.

*1) **Electrical Model**:* The schematic of an equivalent circuit model (electrical model) [1] is shown in Figure 1.

The state equation for the State of charge (SOC), z, is given by:

$$\dot{z} = -\frac{1}{Q_{bat}} I \tag{1}$$

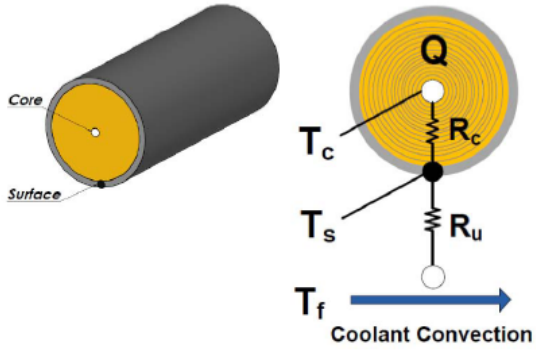


Fig. 2: The two-state thermal model.

where Q is the nominal capacity of battery. The current I is assumed to be positive for discharging. The voltage dynamics of the RC pairs are calculated by

$$\begin{aligned}\dot{V}_1 &= -\frac{1}{R_1 C_1} V_1 + \frac{1}{C_1} I \\ \dot{V}_2 &= -\frac{1}{R_2 C_2} V_2 + \frac{1}{C_2} I\end{aligned} \tag{2}$$

where states V_1 and V_2 are voltages indicated in Figure 1. The measured output is the terminal voltage V_T.

$$\begin{aligned}V_T &= V_{OCV}(z) - V_1 - V_2 - I R_s \\ V_{OCV}(z) &= p_0 + p_1 z + p_2 z^2 + p_3 z^2\end{aligned} \tag{3}$$

where $V_{OCV}(z)$ represents the open circuit voltage and is a function of SOC.

Note that in the coupled electro-thermal model, the parameters R_1, C_1, R_2, C_2, and R_s are functions of SOC or average temperature T_m defined later in the thermal model. Please refer to the Appendix for detail. The battery temperature is driven by the heat generation Q defined as:

$$Q = I(V_{OCV} - V_T) \tag{4}$$

*2) **Thermal Model**:* The schematic of the thermal model [1] is shown in Figure 2. In the model, the input T_f is the inlet air coolant temperature and Q is shown in (4). C_c and C_s are the thermal capacitance of the core and the surface, respectively. R_c and R_s are the conduction resistances between the core and the surface of the cell and around the cell, respectively. The model is described in [1] in more detail.

The dynamics of the thermal model can be defined as the following:

$$\begin{aligned}C_c \dot{T}_c &= Q + \frac{T_s - T_c}{R_c} \\ C_s \dot{T}_s &= \frac{T_f - T_s}{R_u} - \frac{T_s - T_c}{R_c}\end{aligned} \tag{5}$$

where T_c and T_s represent the core and surface temperatures respectively. The measured output is T_s.

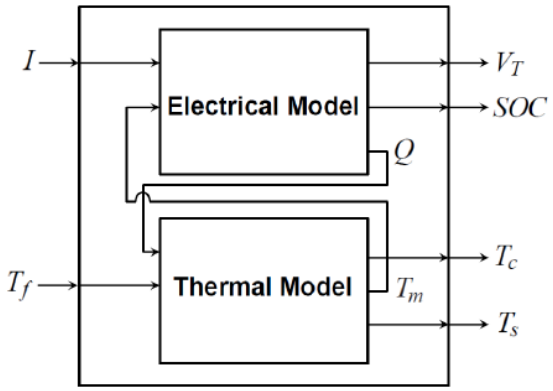


Fig. 3: The five-state electro-thermal model.

The temperature T_m is the average of T_c and T_s. It is used to parametrize the electrical parameters in the coupled model in the next section.

$$T_m = \frac{T_s + T_c}{2} \tag{6}$$

*3) **Coupled Model**:* The coupled electro-thermal model is shown in Figure 3. The heat generation Q from the electrical model, in equation (4), is an input to the thermal model. The thermal model takes Q and T_f as its input and updates T_c and T_s, as well as T_m dynamically. The electrical model uses T_m and z to update the appropriate electrical parameters (R_1, R_2, R_s, C_1 and C_2) [1]. The measured outputs $y_m = h(\boldsymbol{x})$ are the terminal voltage V_T and the surface temperature T_s. The state information for SOC and core temperature could not be measured and thus could only be estimated. Here is a summary of the state-space model.

$$\dot{x} = f(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} \dot{z} \\ \dot{V}_1 \\ \dot{V}_2 \\ \dot{T}_c \\ \dot{T}_s \end{bmatrix} = \begin{bmatrix} -\frac{1}{Q_{bat}} I \\ -\frac{1}{R_1 C_1} V_1 + \frac{1}{C_1} I \\ -\frac{1}{R_2 C_2} V_2 + \frac{1}{C_2} I \\ \frac{V_1 I + V_2 I + R_s I^2}{C_c} + \frac{T_s - T_c}{R_c C_c} \\ \frac{T_f - T_s}{R_u C_s} - \frac{T_s - T_c}{R_c C_s} \end{bmatrix}$$

$$y_m = h(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} V_T \\ T_s \end{bmatrix} = \begin{bmatrix} V_T = V_{OCV}(z) - V_1 - V_2 - I R_s \\ T_s \end{bmatrix} \tag{7}$$

where

$$\begin{aligned}\boldsymbol{x} &= \begin{bmatrix} z & V_1 & V_2 & T_c & T_s \end{bmatrix}^T \\ \boldsymbol{u} &= \begin{bmatrix} I & T_f \end{bmatrix}^T\end{aligned}$$

B. *Observer Design*

As mentioned before, the goal of the observer design is to enable access to SOC and core temperature, since there are no direct measurements available. To do so, we first studied the observability of the model. Secondly, two nonlinear observers, namely EKF and EnKF, are designed and

implemented. Lastly we briefly discuss a Lyapunov-based nonlinear observer known as the *Thaus method*.

Due to the nonlinear nature of the electro-thermal model, it is hard to verify the correctness of the observer design. Therefore, we designed observers progressively for three models, namely a 3-state electrical model, a 5-state parameter-decoupled model, and a 5-state parameter-coupled model. The non-linearity increases respectively.

The 3-state electrical model is self-explanatory. In the 5-state parameter-decoupled setup, all parameters stay constant. Therefore, the electrical model parameters are not affected by the thermal model. In the 5-state parameter-coupled model, the electrical parameters are temperature (T_m) and SOC dependent.

We implemented EKF on all three models, and EnKF on the 5-state parameter-coupled model.

*1) **Observability Analysis**:* Before designing an observer, we first need to check observability. For a nonlinear model, local observability is checked by calculating the observability matrix O in equation (8).

$$O = \frac{\partial}{\partial x}\begin{bmatrix} L_f^0(V_T) & L_f^0(T_s) \\ L_f^1(V_T) & L_f^1(T_s) \\ L_f^2(V_T) & L_f^2(T_s) \\ L_f^3(V_T) & L_f^3(T_s) \\ L_f^4(V_T) & L_f^4(T_s) \end{bmatrix}_{x=x_0} \tag{8}$$

The observability matrix was evaluated in MuPad at the initial condition $x(0) = x_0$. The resulted matrix is quite lengthy, but the rank of matrix O was found to be 5, which is the same as the number of states. It indicates that the system is locally observable from the initial condition.

The observability is also checked at each time step for the linearized (A, C) pair. The linearized (A, C) pair for the 5-state coupled model are calculated in equation (12).

*2) **Extended Kalman Filter (EKF)**:* EKF is essentially applying a Kalman filter to the linearized model about the current state estimate at every time step. The three battery models mentioned before (3-state, 5-state decoupled and 5-state coupled) were created in MATLAB. We implemented an Extended Kalman Filter (EKF) on each of the three to ensure the correctness of the observer.

Consider a non-linear dynamic system model as in (9),

$$\begin{aligned} \dot{x}(t) &= f(x,u) + w \\ \hat{x}(0) &= \bar{x}_0 \\ y_m(t) &= h(x,u) + n \end{aligned} \tag{9}$$

with w and n being process noise and measurement noise, respectively. Both are stationary Gaussian white noise. In KF, w and n are assumed to be mutually uncorrelated. Their covariance matrices are W and N respectively. W and N in the EKF were tuned to achieve an acceptable convergence.

The general formulation of the EKF is provided in equation (10).

$$\begin{aligned} \dot{\hat{x}}(t) &= f(\hat{x},u) + L(t)\left[y(t) - h(\hat{x},u)\right] \\ \hat{x}(0) &= \bar{x}_0 \end{aligned} \tag{10}$$

The observer gain L is updated by $\Sigma(t)$, which is solved by Riccati equation (11):

$$\begin{aligned} L(t) &= \Sigma(t)C^T(t)N^{-1} \\ \dot{\Sigma}(t) &= \Sigma(t)A^T(t) + A(t)\Sigma(t) + W \\ &\quad - \Sigma(t)C^T(t)N^{-1}C(t)\Sigma(t) \\ \Sigma(0) &= \Sigma_0 \end{aligned} \tag{11}$$

with $A(t) = \frac{\partial f}{\partial x}(\hat{x}(t), u(t))$ and $C(t) = \frac{\partial h}{\partial x}(\hat{x}(t), u(t))$. The A and C matrices for the 5-state coupled model are shown in equation (12). The parameters R_s, R_1,R_2,C_1 and C_2 are functions of either z or T_m. Their parametrization is shown in the Appendix. Note that in the 5-state decoupled model, the parameters are constants.

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ A_{21} & -\frac{1}{R_1C_1} & 0 & A_{24} & A_{25} \\ A_{31} & 0 & -\frac{1}{R_2C_2} & A_{34} & A_{35} \\ 0 & \frac{I}{C_c} & \frac{I}{C_c} & A_{44} & A_{45} \\ 0 & 0 & 0 & \frac{1}{R_cC_s} & -\frac{1}{R_uC_s - R_cC_s} \end{bmatrix}$$

$$C = \begin{bmatrix} C_{11} & -1 & -1 & C_{14} & C_{15} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

where

$$\begin{aligned} A_{21} &= \frac{V_1}{R_1^2C_1}(R_{11*} + 2R_{12*}z)e^{\frac{T_{refR_1*}}{T_m - T_{shiftR_1*}}} \\ &\quad + \frac{(\frac{V_1}{R_1} - I)[C_{11*} + 2C_{12*}z + (C_{14*} + 2C_{15*})T_m]}{C_1^2} \\ A_{24} &= A_{25} = -\frac{V_1T_{refR_1*}}{2R_1C_1(T_m - T_{shiftR_1*})^2} \\ &\quad + \frac{(\frac{V_1}{R_1} - I)(C_{13*} + C_{14*}z + C_{15*}z^2)}{2C_1^2} \\ A_{31} &= \frac{V_2}{R_2^2C_2}(R_{21*} + 2R_{22*}z)e^{\frac{T_{refR_2*}}{T_m}} \\ &\quad + \frac{(\frac{V_2}{R_2} - I)(C_{21*} + 2C_{22*}z + (C_{24*} + 2C_{25*})T_m)}{C_2^2} \\ A_{34} &= A_{35} = -\frac{V_2T_{refR_s*}}{2R_2C_2T_m^2} + \frac{(\frac{V_2}{R_2} - I)(C_{23*} + C_{24*}z + C_{25*}z^2)}{2C_2^2} \\ A_{44} &= -\frac{I^2R_sT_{refR_s*}}{2C_c(T_m - T_{shiftR_s*})^2} - \frac{1}{R_cC_c} \\ A_{45} &= -\frac{I^2R_sT_{refR_s*}}{2C_c(T_m - T_{shiftR_s*})^2} + \frac{1}{R_cC_c} \\ C_{11} &= p_1 + 2p_2z + 3p_3z^2 \\ C_{14} &= C_{15} = \frac{IR_sT_{refR_s*}}{2(T_m - T_{shiftR_s*})^2} \end{aligned}$$

*3) **Ensemble Kalman Filter (EnKF)**:* The EnKF is different from the EKF in the way it generates the covariance matrices. In EnKF, Gaussian random noise is added to the initial condition (IC) and generate an ensemble of N slightly different IC samples. Then all samples go through a prediction

update and a correction update. After that the covariance matrices are calculated from the updated samples. Unlike EKF, EnKF is commonly implemented in discrete time. Equation (13) and (14) shows the general formulation of EnKF.

$$
\begin{aligned}
&prediction\ step:\\
&\hat{x}_i(k+1|k) = f(\hat{x}_i(k|k), u(k))\\
&\bar{x}(k+1|k) = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_i(k+1|k)\\
&\tilde{x}_i(k+1|k) = \hat{x}_i(k+1|k) - \bar{x}_i(k+1|k)\\
&\mathbb{E}_x(k+1) = [\tilde{x}_1(k+1|k)\ ...\ \tilde{x}_N(k+1|k)]\\
&\hat{y}_i(k+1|k) = h(\hat{x}_i(k|k), u(k))\\
&\bar{y}(k+1|k) = \frac{1}{N}\sum_{i=1}^{N}\hat{y}_i(k+1|k)\\
&\tilde{y}_i(k+1|k) = \hat{y}_i(k+1|k) - \bar{y}_i(k+1|k)\\
&\mathbb{E}_y(k+1) = [\tilde{y}_1(k+1|k)\ ...\ \tilde{y}_N(k+1|k)]\\
&\hat{P}_{xy}(k+1) = \frac{1}{N-1}\mathbb{E}_x(k+1)\mathbb{E}_y^T(k+1)\\
&\hat{P}_{yy}(k+1) = \frac{1}{N-1}\mathbb{E}_y(k+1)\mathbb{E}_y^T(k+1)
\end{aligned}
\tag{13}
$$

$$
\begin{aligned}
&correction\ step:\\
&\hat{L}(k+1) = \hat{P}_{xy}(k+1)\hat{P}_{yy}^{-1}(k+1)\\
&\hat{x}_i(k+1|k+1) = \hat{x}_i(k+1|k)\\
&\qquad + \hat{L}(k+1)\left(y_m(k+1) - \hat{h}(\hat{x}_i(k+1|k))\right)\\
&\bar{x}(k+1|k+1) = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_i(k+1|k+1)
\end{aligned}
\tag{14}
$$

*4) **Lyapunov-Based Nonlinear Observer**:* Another nonlinear observer studied was a Lyapunov-based nonlinear observer known as the *Thaus Method*. This is a deterministic approach in contrast to the stochastic EKF. Thaus Method assumes that we can write the dynamic model in three terms $\dot{\boldsymbol{x}} = A\boldsymbol{x} + g(t, \boldsymbol{u}, \boldsymbol{y}) + f(\boldsymbol{x})$, with $\boldsymbol{y} = C\boldsymbol{x}$ being the measurements. This formulation assumes that the measurements are linear and $g(t, u, z)$ is known exactly and thus could be canceled completely in the observer. However after formulating this observer, we observed that $g(t, u, z)$ depends on the state estimates, and therefore we were not able to completely cancel this term. In addition our measurements are non-linear. An effort was made by using the estimated states to evaluate $g(t, u, z)$, as well as linearizing the measurements. However since the conditions on the method are not satisfied, convergence is not guaranteed. This observer design was therefore not satisfactory and is not included in this paper.

C. Controller Design

Three controllers are implemented for fast-charging. The first approach is optimal control using dynamic programming. The other two approaches are non-linear controllers using feedback linearization and sliding control. The goal is to achieve a constant desired SOC from some initial condition, without overheating the battery core. The parameters are either chosen or controlled to make sure that the core temperature is within the satisfied regime. The following sections will briefly explain the formulation of the controllers.

*1) **Dynamic Programming**:* The optimization problem is written in equation (15). The cost function is the total non-charging time. $\mathbf{1}\left(I(k) = 0\right)$ is an indicator function. It evaluates to 1 when $I(k) = 0$ and 0 otherwise. The reason is to penalize the system when the battery is not charging. The terminal cost $V_N(z)$ is infinity when $z > z_{desired}$ and zero otherwise (17), since we don't want any more current when the desired SOC is reached. $x(k+1) = f_d(x(k), u(k))$ is the discretized dynamics using Runge-Kutta method, which is achieved by calling $ode45$ in MATLAB. The coolant temperature is constant (T_{f0}) throughout. All the other state variables are bounded accordingly.

Although the terminal SOC, $z(N)$, should be equal to $z_{desired}$ eventually, two other constarints may overwrite this constraint if the time horizon N is not long enough. First, the core temperature should not exceed a maximum temperature $T_{c,max}$. Second, the largest current is limited to I_{max}. These two constraints have priority over the $z(N)$ constraint. It could be easily achieved by taking the smallest current magnitude of the three when performing dynamic programming.

$$
\begin{aligned}
\min_{N,z(k),I(k)} J &= \sum_{k=0}^{N-1}\Delta t\ \mathbf{1}\left(I(k) = 0\right) + V_N(z)\\
s.to:\quad x(k+1) &= f_d(x(k), u(k)),\quad k = 0, ..., N-1\\
x(0) &= x_0\\
T_f(k) &= T_{f0}\\
z_{min} \le z(k) &\le z_{max},\quad k = 0, ..., N\\
V_1(k) &\le 0,\quad k = 0, ..., N\\
V_2(k) &\le 0,\quad k = 0, ..., N\\
T_{s,min} \le T_s(k) &\le T_{s,max},\quad k = 0, ..., N\\
T_{c,min} \le T_c(k) &\le T_{c,max},\quad k = 0, ..., N\\
-I_{max} \le I(k) &\le 0,\quad k = 0, ..., N\\
&and\ possibly:\\
z(N) &= z_{desired}
\end{aligned}
\tag{15}
$$

Let $V_k(\boldsymbol{x}(k))$ denote the total non-charging time from step k to the terminal step N, then the principle of optimality equation can be written as:

$$
V_k(\boldsymbol{x}(k)) = \min_{N,z(k),I(k)}\Delta t\ \mathbf{1}\left(I(k) = 0\right) + V_{k+1}(\boldsymbol{x}(k+1))
\tag{16}
$$

with the terminal constraint:

$$
V_N(\boldsymbol{x}(N)) = \begin{cases}\infty, & z(N) \ge z_{desired}\\ 0, & otherwise\end{cases}
\tag{17}
$$

Now, we can solve the DP problem backward in time. Note that since the goal is fast-charging, the time-horizon N is also a minimizer, although there is no direct way to incorporate

this in DP. The best we can to is to choose a long enough time horizon to include all initial conditions. It turns out that this is a time-consuming process. Without reducing the state space, the DP process takes over $14\ hours$ for a time horizon of $200\ steps$.

*2) **Feedback linearization**:* Feedback linearization is an exact state transformation and feedback method. The idea is to transform a non-linear systems dynamics into a fully or partially linear one and use linear methods for control. For our non-linear model, the objective is to control SOC to a desired state. Assume that SOC could be estimated from our observer, then we can differentiate the SOC until the control term appears:

$$\begin{aligned} y &= z \\ \dot{y} = \dot{z} &= -\frac{I}{Q_{bat}} = v \end{aligned} \tag{18}$$

The control term I appears in the first derivative. Therefore the relative degree of output with respect to input is $r = 1$ which equates the order of the system, $n = 1$. Since $n = r$, there are no internal dynamics and the input/output linearization is stable. Defining the desired dynamics for the synthetic input v to be:

$$v = -\lambda(z - z_{des}) \tag{19}$$

Solving for the control law we receive:

$$I = -\lambda Q_{bat}(z - z_{des}) \tag{20}$$

*3) **Sliding Control**:* The second nonlinear control approach is sliding control. This part consists of three sub-sections, namely sliding mode (SM) control on SOC, sliding control on SOC with a smoothing function, and a secondary sliding surface to control both SOC and T_c.

a) *Sliding Mode (SM) Control on SOC:*

Define the sliding surface to be

$$S = \left(\frac{d}{dt} + \lambda\right)^{n-1} (z - z_{des}) \tag{21}$$

which simplifies to $S = (z - z_{des})$. The chosen sliding surface satisfies two conditions:

- If $S = 0$, then the tracking error goes to zero;
- $\dot{S}$ is an explicit function of the control I

Next, we calculate the derivative of the sliding surface and equating it to a switching function.

$$\dot{S} = -\frac{I}{Q_{bat}} = -\eta sgn(S) \tag{22}$$

Solving for controller I we get:

$$I = \eta Q_{bat} sgn(S) \tag{23}$$

b) *Sliding control on SOC with smoothing function:*

An extension to the SMC method is sliding control with a smoothing function. Smoothing is done to prevent chattering in the input I. We replace the signum function, $sgn(S)$, with a saturation function , $sat(S/\phi)$.

$$I = \eta Q_{bat} sat(S/\phi) \tag{24}$$

c) *A secondary sliding surface on controlling both SOC and T_c:*

Next we defined a different sliding surface where it looks at two state errors that we care about the most, SOC as before and additional term for core temperature with a different sliding surface.

Define the sliding surface to be

$$S = \left(\frac{d}{dt} + \lambda\right)^{n-1} (z - z_{des}) + \alpha \left(\frac{d}{dt} + \lambda\right)^{n-1} (T_c - T_{c_{des}}) \tag{25}$$

α is a weighting factor on how much we would like the error term of temperature to affect the sliding surface.

$$\dot{S} = \dot{z} + \alpha \dot{T}_c = -\eta sgn(S) \tag{26}$$

Which after plugging in the values for $\dot{z}$ and $\dot{T}_c$ from equation (7), it could be simplifies to

$$\begin{aligned} \left(\frac{\alpha R_s}{C_c}\right) I^2 + \left(-\frac{1}{Q_{bat}} + \alpha \frac{V_1 + V_2}{C_c}\right) I \\ + \left(\alpha \frac{T_s - T_c}{R_c C_c} + \eta sat\left(\frac{S}{\phi}\right)\right) = 0 \end{aligned} \tag{27}$$

Solving this polynomial for I, we receive two solutions where we choose the negative one for charging.

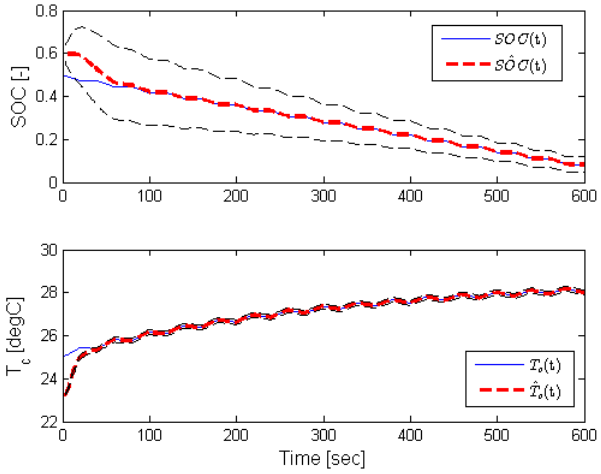
III. Discussion

A. Observer Results

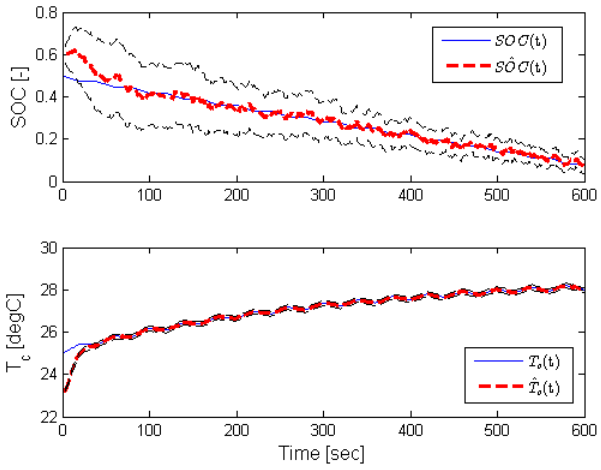
The EKF observers on 3-state model and 5-state decoupled model successfully estimate the states and is satisfactory. Both the estimated SOC and the core temperature T_c converges to the same values as the true model. The observer was also checked with a small amount of measurement noise and the results are still acceptable. Figure 4 shows the 5-state decoupled results. The thin black lines indicates the error bounds within one standard deviation. All time are in second, and temperatures are in degree Celsius.

The results from the 5-state coupled model, however, wasn't as satisfactory (Figure 5). The model tracks the SOC and temperature perfectly with correct initial guess. If the initial estimate is not correct, the SOC estimate exhibits a constant offset and does not converge (Figure 5a). The same conclusion could be made in the case with measurement noise (Figure 5b).

In general, EKF does not guarantee convergence. This is especially true when the model gets too non-linear. The coupled model is highly nonlinear compared to the decoupled model, and therefore the linearized dynamics in the EKF could not represent the model dynamics correctly. Even a little error

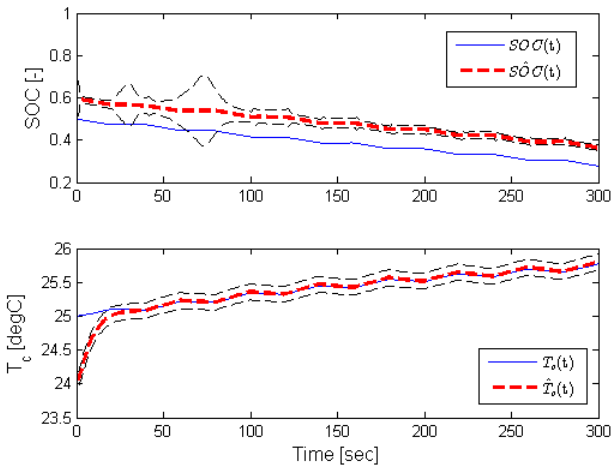


(a) perfect measurement

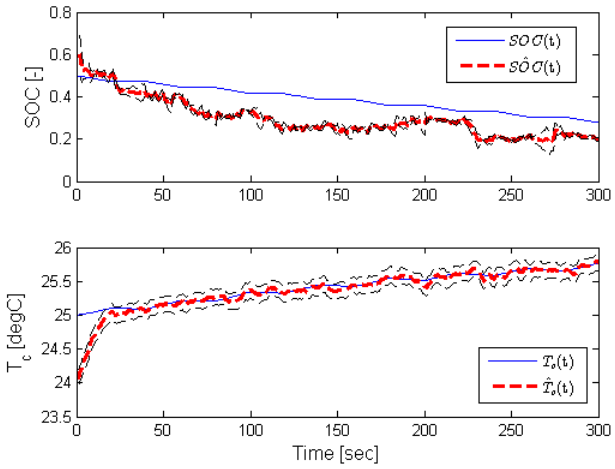


(b) noisy measurement

Fig. 4: EKF results for the 5-state decoupled model.



(a) perfect measurement



(b) noisy measurement

Fig. 5: EKF results for the 5-state coupled model.

in the initial guess results in permanent error residuals in the estimated state trajectory. Our conclusion is that EKF cannot estimate the coupled electro-thermal model correctly.

Then we tried the EnKF. However, the results are not very stable. We tried to increase the number of samples gradually. Good results show up most of the time. Figure 6 shows the best result with 100 Gaussian random samples. The SOC estimate looks better in this case, although the core temperature converges slower compared to EKF. Note that for systems with a large number of states, EnKF is more efficient. However, for our small five-state system, it is much slower than EKF due to the large random samples required for good convergence.

B. Controller Results

Since the observers are not successful in estimating the SOC, we assume that we know all the states when designing the controllers. We successfully implemented all three controllers. The optimal controller is the most effective in

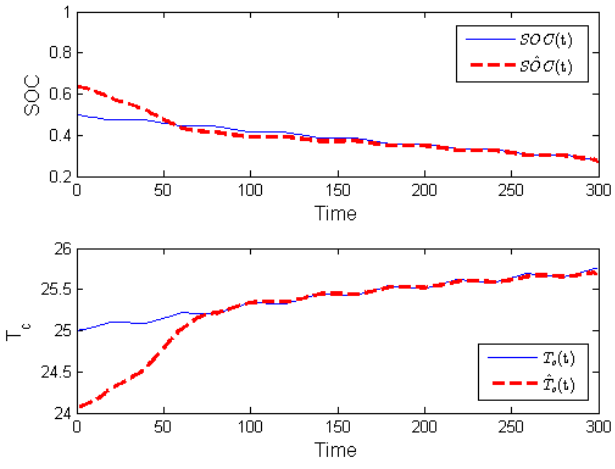


Fig. 6: EnKF results for the 5-state coupled model.

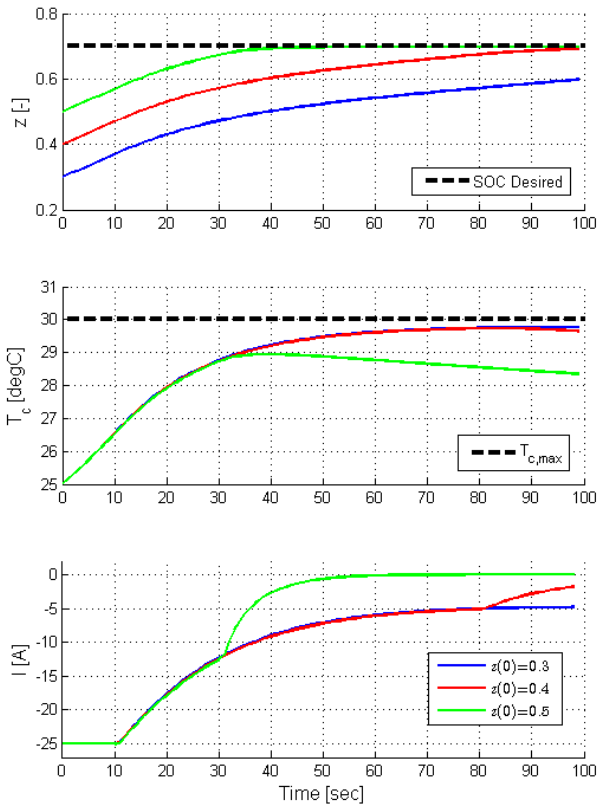


Fig. 7: Fast-charging results from dynamic programming.

constraining battery core overheating. In addition, feedback linearization and sliding control both achieve acceptable SOC tracking, with sub-optimal temperature constraints.

*1) **Dynamic Programming**:* The battery fast-charging results via dynamic programming is shown in Figure 7. Since the simulation is slow, I put an unrealistic maximum current of $25A$ to speed up the process. In addition, states V_1 and V_2 are set to constants ($V_1 = -0.20V$ and $V_2 = -0.06V$) and not gridded when setting up the DP in MATLAB. The black dash line indicates the SOC and T_c thresholds to be 0.7 and $30°C$, respectively.

Three initial SOC's are tested. The blue line indicates $z(0) = 0.3$, the dominant constraint is the core temperature. To prevent overheating, the current is reduced to ensure not exceeding the core temperature threshold, and the desired SOC is not reached within $100sec$. The red line starts at $z(0) = 0.4$ indicates that the system eventually reaches the desired SOC at about $t = 100sec$ without compromising the core temperature threshold. The green line with $z(0) = 0.6$ indicates that once the curre

*2) **Feedback linearization**:* Figure 8 shows the tracking result using feedback linearization.

There are some shortcomings along with this method. Although it achieves the desired SOC, we have no direct control over the core temperature. In this case, the temperature is indirectly bounded by tuning λ. In addition, we don't have

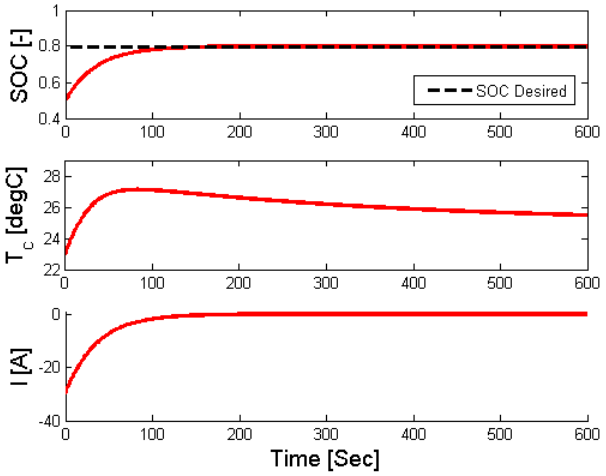


Fig. 8: Fast-charging results from feedback linearization.

control over the maximum current limits. The initial current is huge, about $30A$. Lastly, feedback linearization assumes a perfect model, and thus not robust. Sliding Control, on the other hand, can account for model uncertainties.

*3) **Sliding Control**:* In order to keep the organization in the same order, we will present the result in three subsections as before.

a) *Sliding Mode Control (SMC) on SOC:*

Using sliding mode control, we can guarantee that the sliding surface S goes to zero in finite time. Although we can get better temperature behavior by tuning η , chattering is an issue. Chattering is the discontinuous infinite frequency control signal presented after reaching the sliding surface, due to the discontinuity in the $sgn(S)$ function.

With $\eta = 0.5$, we can observe that the SOC converges to the desired set-point almost instantaneously and then chatters around the desired SOC. Also we can observe the temperature goes much higher than the acceptable range. Decreasing η slows down the convergence to the desired SOC and leads to a much more controlled temperature as seen in Figure 9. It shows the result of a tuned SMC as well as the chattering issue. Again, we don't have direct control over the core temperature.

b) *Sliding Control with smoothing on SOC:*

To resolve the chattering issue, we used a smoothing function, $sat(S/\phi)$ to replace $sgn(S)$. Although chattering is eliminated, we can only guarantee that S converges to a non-zero boundary layer in a finite time. The fast-charging result is shown in red in Figure 10. We can observe that chattering goes away in the current I. However, the core temperature is still out-of-control.

c) *Sliding surfaces on both SOC and T_c:*

To gain control over the core temperature, we combine the SOC sliding surface and a new T_c sliding surface with relative weighing factors. The result is shown in blue in Figure 10. Now, the T_c is well-behaved. The maximum core temperature only exceeds the threshold ($25°C$) by a little. In addition, by a slight tuning, we were able to reduce the large initial current to be less than $10A$. The overall performance is close to DP.

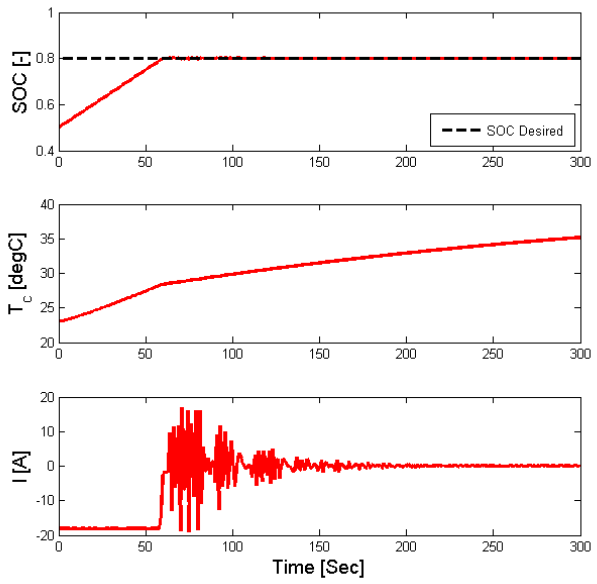


Fig. 9: Fast-charging results from sliding mode control, with a zoom-in on SOC Chattering.

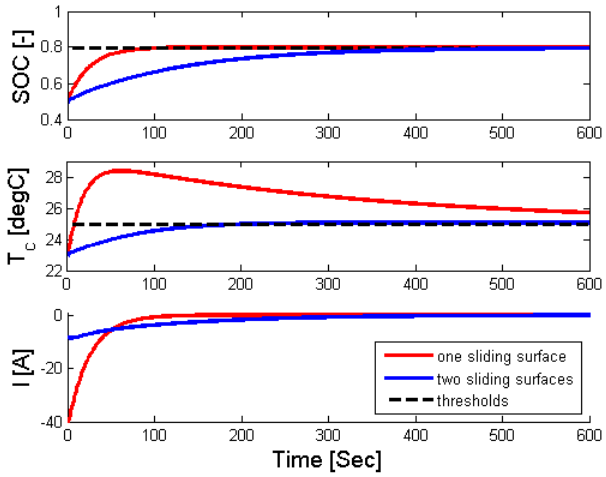


Fig. 10: Fast-charging results from smoothed sliding control, with or without the secondary T_c sliding surface.

IV. Summary

In this project, we examined the observer and controller design of a cylindrical electro-thermal lithium ion battery model.

Observers are designed to estimate the SOC and core temperature. The EKF observer worked well on the 3-state and 5-state decoupled models. However both EKF and EnKF observers were unable to converge on the 5-state coupled model if the initial state estimates were incorrect. EKF failed due to the model non-linearity, and EnKF failed due to its inherent randomness and the system non-linearity. For the EKF observers, we also tested their robustness with noise-contaminated measurements.

In addition, three controllers were implemented. An optimal control strategy via dynamic programming was used to efficiently perform fast-charging while preventing overheating. Feedback linearization on SOC was successful, but it has issues on robustness and temperature control. Lastly, multiple sliding control strategies were discusesed. A sliding control method, with two weighted sliding surfaces taking both the core temperature and the SOC into account, was proven successful. MATLAB simulations are provided for the non-linear model, the observer design, and the controller design.

Given the unsuccessful state estimates, it is essential to simplify the existing non-linear model to achieve full-state feedback control.

References

[1] X. Lin, H. E. Perez, S. Mohan, J. B. Siegel, A. G. Stefanopoulou, Y. Ding, and M. P. Castanier, "A lumped-parameter electro-thermal model for cylindrical batteries," *Journal of Power Sources*, vol. 257, pp. 1–11, 2014.

[2] A. Rauh, S. S. Butt, and H. Aschemann, "Nonlinear state observers and extended kalman filters for battery systems," *International Journal of Applied Mathematics and Computer Science*, vol. 23, no. 3, pp. 539–556, 2013.

[3] S. Gillijns, O. B. Mendoza, J. Chandrasekar, B. De Moor, D. Bernstein, and A. Ridley, "What is the ensemble kalman filter and how well does it work?" in *American Control Conference, 2006*. IEEE, 2006, pp. 6–pp.

[4] F. L. Lewis and V. L. Syrmos, *Optimal control*. John Wiley & Sons, 1995.

[5] H. K. Khalil and J. Grizzle, *Nonlinear systems*. Prentice hall New Jersey, 1996, vol. 3.

[6] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice-Hall Englewood Cliffs, NJ, 1991, vol. 60.

V. Appendix

A. Temperature-Varying Parameters for the Electrical Model

$$R_s = \begin{cases} R_{sd}, & I \geq 0 \ (discharge) \\ R_{sc}, & I < 0 \ (charge) \end{cases}$$

$$R_{s*} = R_{s0*} exp\left(\frac{T_{refR_{s}*}}{T_m - T_{shiftR_{s}*}}\right)$$

R_{s0d}	R_{s0c}	T_{refR_sd}	T_{refR_sc}	T_{shiftR_sd}	T_{shiftR_sc}
0.0048	0.0055	31.0494	22.2477	−15.3253	−11.5943

TABLE I: Parametric R_s function parameters

$$R_1 = \begin{cases} R_{1d}, & I \geq 0 \ (discharge) \\ R_{1c}, & I < 0 \ (charge) \end{cases}$$

$$R_{1*} = \left(R_{10*} + R_{11*}(SOC) + R_{12*}(SOC)^2\right) exp\left(\frac{T_{refR_{1}*}}{T_m - T_{shiftR_{1}*}}\right)$$

R_{10d}	R_{10c}	R_{11d}	R_{11c}	R_{12d}
$7.1135e{-4}$	0.0016	$-4.3865e{-4}$	−0.0032	$2.3788e{-4}$
R_{12c}	T_{refR_1d}	T_{refR_1c}	T_{shiftR_1d}	T_{shiftR_1c}
0.0045	347.4707	159.2819	−79.5816	−41.4548

TABLE II: Parametric R_1 function parameters

$$R_2 = \begin{cases} R_{2d},\ I \geq 0\ (discharge) \\ R_{2c},\ \ I < 0\ (charge) \end{cases}$$
$$R_{2*} = \left(R_{20*} + R_{21*}(SOC) + R_{22*}(SOC)^2\right) exp\left(\frac{T_{refR_2*}}{T_m}\right)$$

R_{20d}	R_{20c}	R_{21d}	R_{21c}
0.0288	0.0113	-0.073	-0.027
R_{22d}	R_{22c}	T_{refR_2d}	T_{refR_2c}
0.0605	0.0339	16.6712	17.0224

TABLE III: Parametric R_2 function parameters

$$C_1 = \begin{cases} C_{1d},\ I \geq 0\ (discharge) \\ C_{1c},\ \ I < 0\ (charge) \end{cases}$$
$$C_{1*} = \left(C_{10*} + C_{11*}(SOC) + C_{12*}(SOC)^2\right) \left(C_{13*} + C_{14*}(SOC) + C_{15*}(SOC)^2\right) T_m$$

C_{10d}	C_{10c}	C_{11d}	C_{11c}
335.4518	523.215	$3.1712e{+}3$	$6.4171e{+}3$
C_{12d}	C_{12c}	C_{13d}	C_{13c}
$-1.3214e{+}3$	$-7.5555e{+}3$	53.2138	50.7107
C_{14d}	C_{14c}	C_{15d}	C_{15c}
-65.4786	-131.2298	44.3761	162.4688

TABLE IV: Parametric C_1 function parameters

$$C_2 = \begin{cases} C_{2d},\ I \geq 0\ (discharge) \\ C_{2c},\ \ I < 0\ (charge) \end{cases}$$
$$C_{2*} = \left(C_{20*} + C_{21*}(SOC) + C_{22*}(SOC)^2\right) \left(C_{23*} + C_{24*}(SOC) + C_{25*}(SOC)^2\right) T_m$$

C_{10d}	C_{20c}	C_{21d}	C_{21c}
$3.1887e{+}4$	$6.2449e{+}4$	$-1.1593e{+}5$	$-1.055e{+}5$
C_{22d}	C_{22c}	C_{23d}	C_{23c}
$1.0493e{+}5$	$4.4432e{+}4$	60.3114	198.9753
C_{24d}	C_{24c}	C_{25d}	C_{25c}
$1.0175e{+}4$	$7.5621e{+}3$	$-9.5924e{+}3$	$-6.9365e{+}3$

TABLE V: Parametric C_2 function parameters