

## Research Article

# Tuning of Kalman Filter Parameters via Genetic Algorithm for State-of-Charge Estimation in Battery Management System

T. O. Ting,<sup>1</sup> Ka Lok Man,<sup>2,3</sup> Eng Gee Lim,<sup>1</sup> and Mark Leach<sup>1</sup>

<sup>1</sup> Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, No. 111, Ren'ai Road, HET, SIP, Suzhou, Jiangsu 215123, China

<sup>2</sup> Department of Computer Science and Software Engineering, Xi'an Jiaotong-Liverpool University, No. 111, Ren'ai Road, HET, SIP, Suzhou, Jiangsu 215123, China

<sup>3</sup> Department of Computer Science, Yonsei University, 50 Yonsei-ro, Seodaemun-gu, Seoul 120-749, Republic of Korea

Correspondence should be addressed to T. O. Ting; [toting@xjtlu.edu.cn](mailto:toting@xjtlu.edu.cn)

Received 22 April 2014; Revised 3 June 2014; Accepted 4 June 2014; Published 5 August 2014

Academic Editor: Su Fong Chien

Copyright © 2014 T. O. Ting et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this work, a state-space battery model is derived mathematically to estimate the state-of-charge (SoC) of a battery system. Subsequently, Kalman filter (KF) is applied to predict the dynamical behavior of the battery model. Results show an accurate prediction as the accumulated error, in terms of root-mean-square (RMS), is a very small value. From this work, it is found that different sets of  $\mathbf{Q}$  and  $\mathbf{R}$  values (KF's parameters) can be applied for better performance and hence lower RMS error. This is the motivation for the application of a metaheuristic algorithm. Hence, the result is further improved by applying a genetic algorithm (GA) to tune  $\mathbf{Q}$  and  $\mathbf{R}$  parameters of the KF. In an online application, a GA can be applied to obtain the optimal parameters of the KF before its application to a real plant (system). This simply means that the instantaneous response of the KF is not affected by the time consuming GA as this approach is applied only once to obtain the optimal parameters. The relevant workable MATLAB source codes are given in the appendix to ease future work and analysis in this area.

## 1. Introduction

Battery Management System (BMS) [1–3] comprises of mechanism that monitors and controls the normal operation of a battery system so as to ensure its safety while maintaining its State-of-Health (SoH). The BMS, in essence, measures the voltage, current, and temperature of each cell in a battery pack. These data are then analyzed by a management system that guarantees safe and reliable operations. A common example of an independent battery pack is portrayed in Figure 1. The battery is an essential component and should be accurately modeled in order to design an efficient management system [4]. Hence, a generic tool to describe the battery performance under a wide variety of conditions and applications is highly desirable [4]. As such, electrical modeling is necessary to provide such a tool that enables visualization of the processes occurring inside rechargeable batteries. These generic models are necessary for the development of battery management algorithms. These algorithms

control the operation and maintain the performance of battery packs. In short, the ultimate aim of BMS is to prolong battery life, while ensuring reliable operation alongside many applications, especially in photovoltaic systems [5–7].

Battery modeling is performed in many ways depending on the types of battery. In general, the resulting battery model is a mathematical model comprising numerous mathematical descriptions [8]. Ultimately, battery models aim to determine the state-of-charge (SoC) of the battery system. However, the complexity of the nonlinear electrochemical processes has been a great barrier to modeling this dynamic process accurately. The accurate determination of the SoC will enable utilization of the battery for optimal performance and long-life and prevent irreversible physical damage to the battery [9]. The solution to the SoC via neural networks [10] and fuzzy logic [11] has been difficult and costly for online implementation due to the large computation required, causing the battery pack controller to be heavily loaded. This may



FIGURE 1: Typical Lithium-ion battery packs for electric vehicle.

however be a good alternative in the near future as the computational power of processing chips increase alongside their declining cost.

The state-estimation process usually leads to some state variables in a dynamical system. The SoC is a measure of a battery's available power and thus it is important to calculate this value accurately from BMS by the cell voltage, temperature, and polarization effect caused by the electrolyte concentration gradient during high rate charging/discharging cycle [12]. Recently, the battery state-of-charge (SoC) is one of the significant elements attracting significant attention [13, 14]. By definition, SoC is the ratio of remaining capacity to the nominal capacity of the battery. Here, the remaining capacity is the number of ampere-hours (Ah) that can be extracted at normal operating temperature. The mathematical expression for the SoC is given in [13, 14], which is

$$Z(t) = Z(0) + \int_0^t \frac{I_b(\tau)}{C_n} d\tau, \quad (1)$$

where  $t$  is time,  $z(t)$  is battery SoC, in ampere-hours (Ah),  $I_b$  is current flowing through the battery (passing through  $C_{bk}$ ), illustrated in Figure 2, and  $C_n$  is nominal battery capacity. For charging,  $I_b > 0$  and for discharging,  $I_b < 0$ .

From this mathematical expression, it is noted that the SoC cannot be explicitly measured. In the literature, there is a myriad of methods dealing with predicting and estimating of SoC. The most popular of these methods are described in the following paragraphs.

At present, Coulomb-counting [15], also known as charge counting, or current integration is the most commonly used technique; it requires dynamic measurement of battery current. This is an open-loop method; however, it suffers from a reliance on the mathematical integration, and errors (noise, resolution, and rounding) are cumulative, which can lead to large SoC errors at the end of the integration process in (1). On the positive side, if an accurate current sensor is incorporated, the implementation will be much easier.

Another prominent SoC estimator is the well-known Kalman filter (KF), invented by Kalman in 1960. Although

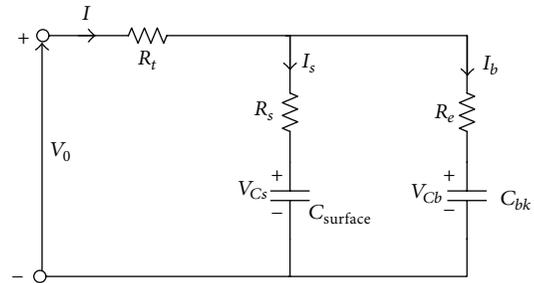


FIGURE 2: Schematic of RC battery model.

his popular work was published almost 54 years ago in [16], it remains as an important citation source in the literature. Readers who are new to this method can refer to an excellent KF tutorial by Faragher in [17]. The KF method is a well-known technique for dynamic state estimation of systems such as target tracking, navigation, and also battery systems [18, 19]. The state-of-the-art method provides recursive solution to linear filtering for state observation and prediction problems. The key advantage of the KF is that it accurately estimates states affected by external disturbances such as noises governed by Gaussian distribution. On the contrary, the disadvantage of KF is that it requires highly complex mathematical calculations. This can be realized by a state-space model, as shown in previous work by the author in [20, 21]. The modeling is a heavy duty task and is also presented in this work to ease the understanding of readers. As such, there exist some possibilities of divergence due to an inaccurate model and complex calculation. In the case of a slow processor, the calculation results may be delayed and exceed the sampling interval, thereby result in an inaccurate tracking.

Various artificial intelligence (AI) methods, mainly the neural networks and fuzzy logic, are being applied in the estimation of battery's SoC [10, 22]. Neural networks are computationally expensive, which can overload the BMS and thus this approach, though theoretically feasible, may not be suitable for online implementation that requires instantaneous feedback and action. Also, neural networks require huge datasets in the time-consuming training process. Other techniques for SoC, include the sliding mode observer, are reported in [12].

In this work, a mathematical derivation leading to a state-space model is presented. The basic schematic model is adopted from [18, 20]. A thorough analysis in the form of state variables with the application of the Kalman filter is presented. The rest of the paper is organized as follows. The mathematical model is derived in Section 2, resulting in a state-space model. Further, in Section 3, the KF is applied to estimate the SoC of a battery system. This is followed by the tuning of KF's parameter by adopting a metaheuristic approach, namely, a genetic algorithm in Section 4. Relevant results are presented in Section 5, and finally the conclusions are derived in Section 6.

## 2. Battery Modeling

Many model-based state-estimations have been proposed in [18, 20, 21, 23]. In [18, 21], the well-known Kalman filter [16] had been applied successfully for both state observation and prediction of the SoC. Work in [24] utilized manufacturers' data in modeling the dynamic behavior of the battery. Several battery models exist from many works over the past few years. Each of these models varies in terms of its complexity and applications. In this work, a dynamical battery model is adopted, consisting of state variable equations, from [20, 21]. The schematic representation of this model is shown in Figure 2. In this model, there exists a bulk capacitor  $C_{bk}$  that acts as an energy storage component in the form of the charge, a capacitor that models the surface capacitance and diffusion effects within the cell  $C_{surface}$ , a terminal resistance  $R_t$ , a surface resistance  $R_s$ , and an end resistance  $R_e$ . The voltages across both capacitors are denoted as  $V_{Cb}$  and  $V_{Cs}$ , respectively.

**2.1. Mathematical Derivations of Battery Model.** In this derivation, we aim to form a state-space model consisting of the state variables  $V_{Cb}$ ,  $V_{Cs}$ , and  $V_0$ . State variables are mathematical descriptions of the "state" of a dynamic system. In practice, the state of a system is used to determine its future behavior. Models that consist of a paired first-order differential equations are in the state-variable form.

Following the voltages and currents illustrated in Figure 2, the terminal voltage  $V_0$  can be expressed as

$$V_0 = IR_t + I_b R_e + V_{Cb}, \quad (2)$$

which is similar to

$$V_0 = IR_t + I_s R_s + V_{Cs}. \quad (3)$$

By (2) and (3), and following straightforward algebraic manipulation,  $V_0$  can be written as

$$I_b R_e = I_s R_s + V_{Cs} - V_{Cb}. \quad (4)$$

From Kirchoff's current law,  $I = I_b + I_s$ ,

$$I_s = I - I_b. \quad (5)$$

Thus, substituting (5) into (4) yields

$$I_b (R_e + R_s) = IR_s + V_{Cs} - V_{Cb}. \quad (6)$$

By assuming a slow varying  $C_{bk}$ , that is,  $I_b = C_{bk} \dot{V}_{Cb}$  (from basic formula of  $i = C(\partial V/\partial t)$ ), and then substituting it into (6), and after rearranging results in

$$\dot{V}_{Cb} = \frac{IR_s}{C_{bk}(R_e + R_s)} + \frac{V_{Cs}}{C_{bk}(R_e + R_s)} - \frac{V_{Cb}}{C_{bk}(R_e + R_s)}. \quad (7)$$

By applying a similar derivation, the rate of change of the surface capacitor voltage  $\dot{V}_{Cs}$ , derived also from (2) and (3), gives

$$\dot{V}_{Cs} = \frac{IR_e}{C_{surface}(R_e + R_s)} - \frac{V_{Cs}}{C_{surface}(R_e + R_s)} + \frac{V_{Cb}}{C_{surface}(R_e + R_s)}. \quad (8)$$

Further, by assuming  $A = 1/C_{bk}(R_e + R_s)$  and  $B = 1/C_{surface}(R_e + R_s)$ , (7) and (8) can be written as

$$\begin{aligned} \dot{V}_{Cb} &= A \cdot I \cdot R_s + A \cdot V_{Cs} - A \cdot V_{Cb}, \\ \dot{V}_{Cs} &= B \cdot I \cdot R_e - B \cdot V_{Cs} + B \cdot V_{Cb}, \end{aligned} \quad (9)$$

respectively. Further, (9) and (10) can be combined to form a state variable relating voltages  $V_{Cs}$  and  $V_{Cb}$  and current flow  $I$ , which is

$$\begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cs} \end{bmatrix} = \begin{bmatrix} -A & A \\ B & -B \end{bmatrix} \begin{bmatrix} V_{Cb} \\ V_{Cs} \end{bmatrix} + \begin{bmatrix} A \cdot R_s \\ B \cdot R_e \end{bmatrix} I. \quad (10)$$

Next, the output voltage is derived from (2) and (3). By adding both equations, we obtain

$$2V_0 = 2IR_t + I_b R_e + I_s R_s + V_{Cb} + V_{Cs}. \quad (11)$$

Then by substituting  $I_b = R_s/(R_s + R_e) \cdot I$  and  $I_s = R_e/(R_s + R_e) \cdot I$  into (11), it is further simplified as

$$V_0 = \frac{V_{Cb} + V_{Cs}}{2} + \left( R_t + \frac{R_e R_s}{R_e + R_s} \right) I. \quad (12)$$

Taking the time derivative of the output voltage and assuming  $dI/dt \approx 0$  (this simply means that the change rate of terminal current can be ignored when implemented digitally), we obtain

$$\dot{V}_0 = \frac{\dot{V}_{Cb} + \dot{V}_{Cs}}{2}. \quad (13)$$

Substituting the formulae obtained in (9) and (10) into (13) results in

$$2\dot{V}_0 = (-A + B)V_{Cb} + (A - B)V_{Cs} + (AR_s + BR_e)I. \quad (14)$$

Then, solving for  $V_{Cs}$  from (12) we obtain

$$V_{Cs} = 2V_0 - 2 \left( R_t + \frac{R_e R_s}{R_e + R_s} \right) I - V_{Cb}, \quad (15)$$

and after substituting it into (14), it yields

$$\begin{aligned} \dot{V}_0 &= (-A + B)V_{Cb} + (A - B)V_0 \\ &+ [A(0.5R_s + R_t + D) + B(0.5R_e - R_t - D)]I. \end{aligned} \quad (16)$$

Finally, the complete state variable network is obtained by substituting (16) into (10), represented in matrix form as

$$\begin{aligned} \begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cs} \\ \dot{V}_0 \end{bmatrix} &= \begin{bmatrix} -A & A & 0 \\ B & -B & 0 \\ (-A + B) & 0 & (A - B) \end{bmatrix} \cdot \begin{bmatrix} V_{Cb} \\ V_{Cs} \\ V_0 \end{bmatrix} \\ &+ \begin{bmatrix} A \cdot R_s \\ B \cdot R_e \\ A(0.5R_s - R_t - D) + B(0.5R_e + R_t + D) \end{bmatrix} I, \end{aligned} \quad (17)$$

TABLE 1: Parameters for cell model [18, 20].

| $C_{\text{bk}}$ | $C_{\text{surface}}$ | $R_e$            | $R_s$            | $R_t$             |
|-----------------|----------------------|------------------|------------------|-------------------|
| 88372.83 F      | 82.11 F              | 0.00375 $\Omega$ | 0.00375 $\Omega$ | 0.002745 $\Omega$ |

whereby constants  $A$ ,  $B$ , and  $D$  were derived previously but are hereby restated as

$$\begin{bmatrix} A \\ B \\ D \end{bmatrix} = \begin{bmatrix} \frac{1}{C_{\text{bk}}(R_e + R_s)} \\ \frac{1}{C_{\text{surface}}(R_e + R_s)} \\ \frac{R_e R_s}{R_e + R_s} \end{bmatrix}. \quad (18)$$

This completes the initial derivation of the battery model.

**2.2. Numerical Example.** By substituting all capacitor and resistor values from Table 1 into (18), the following values are obtained:

$$\begin{bmatrix} A \\ B \\ D \end{bmatrix} = \begin{bmatrix} 0.001508759347566 \\ 1.623837940973491 \\ 0.001875000000000 \end{bmatrix}. \quad (19)$$

By defining matrix  $\mathbf{A}$ ,

$$\mathbf{A} = \begin{bmatrix} -A & A & 0 \\ B & -B & 0 \\ (-A + B) & 0 & (A - B) \end{bmatrix}, \quad (20)$$

$$\mathbf{B} = \begin{bmatrix} A \cdot R_s \\ B \cdot R_e \\ A(0.5R_s - R_t - D) + B(0.5R_e + R_t + D) \end{bmatrix}. \quad (21)$$

Again by substituting the values from Table 1 to calculate  $A$ ,  $B$ , and  $D$ , we obtain the value of  $\mathbf{A}$  as

$$\mathbf{A} = \begin{bmatrix} -1.51 \times 10^{-3} & 1.51 \times 10^{-3} & 0 \\ 1.6238 & -1.6238 & 0 \\ 1.6223 & 0 & -1.6223 \end{bmatrix} \quad (22)$$

and  $\mathbf{B}$  as

$$\mathbf{B} = \begin{bmatrix} 5.66 \times 10^{-6} \\ 6.08 \times 10^{-3} \\ 1.05 \times 10^{-2} \end{bmatrix}. \quad (23)$$

As such (17) can be rewritten as

$$\begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cs} \\ \dot{V}_0 \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} V_{Cb} \\ V_{Cs} \\ V_0 \end{bmatrix} + \mathbf{B} \cdot I, \quad (24)$$

or numerically as

$$\begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cs} \\ \dot{V}_0 \end{bmatrix} = \begin{bmatrix} -0.0015 & 0.0015 & 0 \\ 1.6238 & -1.6238 & 0 \\ 1.6223 & 0 & -1.6223 \end{bmatrix} \cdot \begin{bmatrix} V_{Cb} \\ V_{Cs} \\ V_0 \end{bmatrix} + \begin{bmatrix} 5.66 \times 10^{-6} \\ 6.08 \times 10^{-3} \\ 1.05 \times 10^{-2} \end{bmatrix} \cdot I. \quad (25)$$

**2.3. State-Space Modeling.** Based on control theories, a lumped linear network can be written in the form

$$\begin{aligned} \dot{x}(t) &= \mathbf{A}x(t) + \mathbf{B}u(t), \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}u(t), \end{aligned} \quad (26)$$

where in this work, the state variable  $\dot{x}(t)$  is

$$\dot{x}(t) = \begin{bmatrix} \dot{V}_{Cb} \\ \dot{V}_{Cs} \\ \dot{V}_0 \end{bmatrix}. \quad (27)$$

Obviously,

$$x(t) = \begin{bmatrix} V_{Cb} \\ V_{Cs} \\ V_0 \end{bmatrix}, \quad (28)$$

with

$$u(t) = I. \quad (29)$$

Therefore, by restating the previous calculation values in (21) and (23), we should note that the values of  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$  are as follows:

$$\mathbf{A} = \begin{bmatrix} -1.51 \times 10^{-3} & 1.51 \times 10^{-3} & 0 \\ 1.6238 & -1.6238 & 0 \\ 1.6223 & 0 & -1.6223 \end{bmatrix}, \quad (30)$$

$$\mathbf{B} = \begin{bmatrix} 5.66 \times 10^{-6} \\ 6.08 \times 10^{-3} \\ 1.05 \times 10^{-2} \end{bmatrix}, \quad (31)$$

$$\mathbf{C} = [0 \ 0 \ 1], \quad (32)$$

$$\mathbf{D} = [0], \quad (33)$$

respectively. As such, with (32), the output  $y(t)$  is in fact

$$y(t) = V_0. \quad (34)$$

This means that the output of the system is the open terminal voltage  $V_0$ , as expected. Note that this is an important observation from this state-space modeling.

Further, the above state-space variables are transformed to a transfer function,  $G(s)$ . This is done by using *ss2tf* function in MATLAB, which after factorization yields

$$G(s) = \frac{0.01054s^2 + 0.0171s + 2.981 \times 10^{-5}}{s^3 + 3.248s^2 + 2.637s - 1.144 \times 10^{-18}}. \quad (35)$$

The plot of the unit step response for the gain in (35) is given in Figure 3. Basically, it shows that the open circuit terminal voltage  $V_0$  in Figure 2 increases linearly during the charging operation in a very slow manner after transient behaviour for few seconds.

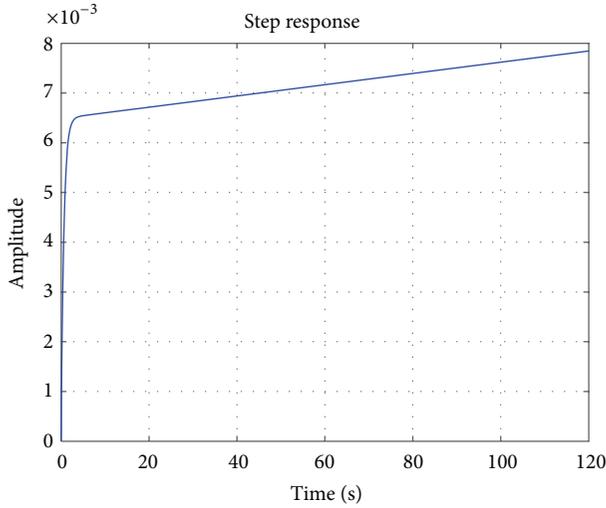


FIGURE 3: Output response of RC model due to constant input.

2.4. *Observability of the RC Battery Model.* In control theory, observability is the degree to which the internal states of a system can be predicted via its external outputs. As such, for an observable system, the behavior of the entire system can be predicted via the system's outputs. On the other hand, if a system is not observable, the current values of some of its states cannot be estimated through the output signal. This means the controller does not know the states' values. In theory, the observability of a system can be determined by constructing an observability matrix  $O_b$ .

$$O_b = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}, \quad (36)$$

and a system is observable if the row rank of  $O_b$  is equal to  $n$  (this is also known as a full rank matrix). The ultimate rationale of such a test is that if  $n$  rows are linearly independent, then each of the  $n$  states is viewable through linear combinations of the output  $y(t)$ .

Further, substituting  $A$  and  $C$  values from (30) and (32) into (36) yields

$$O_b = \begin{bmatrix} 0 & 0 & 1 \\ 1.6223 & 0 & -1.6223 \\ -2.6344 & 0.0024 & 2.6320 \end{bmatrix}. \quad (37)$$

Clearly, in this case  $O_b$  is a full rank matrix, which concludes that this system is observable.

### 3. Kalman Filter for SoC Estimation

A continuous time-invariant linear system can be described in the state variable form as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \quad (38)$$

where  $u(t)$  is the input vector,  $x(t)$  is the state vector,  $y(t)$  is the output vector,  $A$  is the time invariant dynamic matrix,  $B$  is the time invariant input matrix, and  $C$  is the time invariant measurement matrix.

If we assume that the applied input  $u$  is constant during each sampling interval, a discrete-time equivalent model of the system will now be

$$\begin{aligned} x(n+1) &= A_d \cdot x(n) + B_d \cdot u(n), \\ y(n+1) &= C_d \cdot x(n+1), \end{aligned} \quad (39)$$

where

$$A_d \approx I + A \cdot T_c, \quad B_d = B \cdot T_c, \quad C_d = C, \quad (40)$$

whereby  $I$  is the identity matrix and  $T_c$  is the sampling period. As for this system, two independent process noises are present which are additive Gaussian noise,  $w$  vector representing system disturbances and model inaccuracies and  $v$  vector representing the effects of measurement noise.

Both  $w$  and  $v$  have a mean value of zero and the following covariance matrices:

$$\begin{aligned} E[w \cdot w^T] &= Q, \\ E[v \cdot v^T] &= R, \end{aligned} \quad (41)$$

where  $E$  denotes the expectation (or mean) operator and superscript  $T$  means the transpose of the respective vectors. In usual case,  $Q$  and  $R$  are normally set to a constant before simulation; in our case both are set to one (see Section 5). Further, a genetic algorithm (GA) is applied in order to obtain a better set of  $Q$  and  $R$  values resulting in lower RMS error from the KF's output.

By inclusion of these noises, the resulting system can now be described by

$$\begin{aligned} x(n+1) &= A_d \cdot x(n) + B_d \cdot u(n) + w, \\ z(n+1) &= C_d \cdot x(n+1) + v, \end{aligned} \quad (42)$$

which is illustrated in Figure 4,

3.1. *Property of Kalman Filter.* An important property of the KF is that it minimizes the sum-of-squared errors between the actual value  $x$  and estimated states  $\hat{x}$ , given as

$$f_{\min}(x) = E([x - \hat{x}] \cdot [x - \hat{x}]^T). \quad (43)$$

To understand the operations of the KF, the meaning of the notation  $\hat{x}(m | n)$  is crucial. Simply stated, it means that the estimate of  $x$  at event  $m$  takes into account all discrete events up to event  $n$ . As such, (43) can include such information, now expanded as

$$f_{\min}(x) = E([x(n) - \hat{x}(n | n)] \cdot [x(n) - \hat{x}(n | n)]^T). \quad (44)$$

In recursive implementation of the KF, the current estimate  $\hat{x}(n | n)$ , together with the input  $u(n)$  and measurement

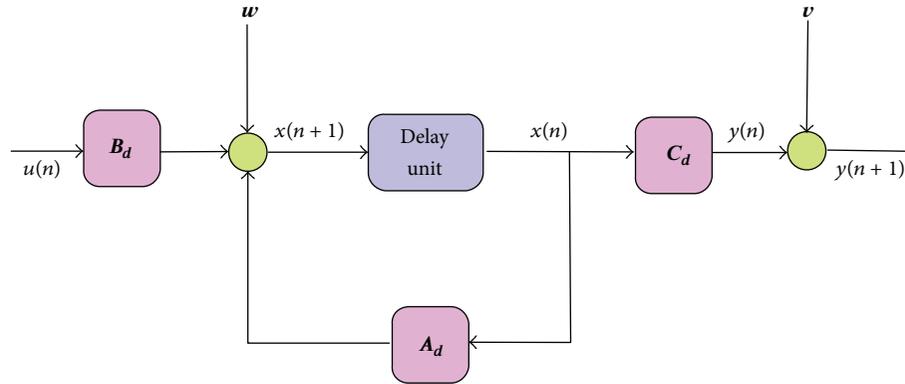


FIGURE 4: Discrete system model with noises  $w$  and  $v$ .

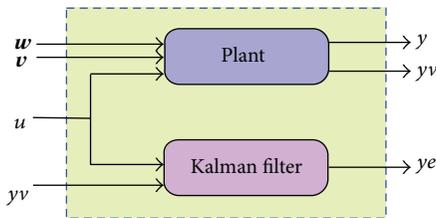


FIGURE 5: Parallel connection of plant and Kalman filter.

```

(1) BEGIN
(2) Initialize population  $P(X)$ 
(3) while  $Terminate = False$  do
(4)  $P(X) \leftarrow f(x)$ : Fitness evaluation,
(5) Selection,
(6) Crossover,
(7) Mutation,
(8) end while
(9) END
    
```

ALGORITHM 1: Pseudocode of GA.

signals  $z$ , is used for further estimating  $\hat{x}(n+1 | n+1)$ . This means that in this discrete system, the input for each sample step will be  $u_1, u_2, u_3, \dots, u_{n+1}$  with respect to the output of  $y_1, y_2, y_3, \dots, y_{n+1}$ . The recursive KF algorithm is obtained with the predictor and corrector stages.

3.2. *KF Online Implementation.* For the case of a battery, it is well understood that only the terminal quantities can be measured (terminal voltage  $V_0$  and current  $I$ ). Assuming that battery parameters are time-invariant quantities, the recursive KF algorithm is applied. By applying (40) into (30)–(32), we obtain the following values of updated matrices, with  $T_c = 1$ :

$$\begin{aligned}
 \mathbf{A}_d &= \begin{bmatrix} 0.9984 & 1.51 \times 10^{-3} & 0 \\ 1.6238 & 0.6238 & 0 \\ 1.6223 & 0 & 0.6223 \end{bmatrix}, \\
 \mathbf{B}_d &= \begin{bmatrix} 5.66 \times 10^{-6} \\ 6.08 \times 10^{-3} \\ 1.05 \times 10^{-2} \end{bmatrix}, \\
 \mathbf{C}_d &= [0 \ 0 \ 1].
 \end{aligned}
 \tag{45}$$

Note that  $\mathbf{B}_d$  and  $\mathbf{C}_d$  remain similar to their previous values, as given in (31) and (32). By utilizing MATLAB’s control toolbox, the KF is placed in parallel to the state-space model, hereby represented by plant in Figure 5. The complete source code is given in the Appendix.

### 4. Genetic Algorithm

The genetic algorithm (GA), introduced by John Holland, is an approach based on biological evolution [25]. The algorithm is developed based on Charles Darwin’s theory of survival of the fittest. The GA has a very powerful encoding mechanism that enables the representation of a solution vector as either a real coded or binary string. Both encodings serve a different purpose in the context of different problem space. GAs are regarded as the global optimizer that often spot or locate the potential area or even accurately obtain the best solution, known as the global minimum [26–28]. Underneath this popular algorithm are the three operators that contribute to its success in performing optimization task.

- (i) *Selection.* In selection, offsprings with higher fitness have better chance for survival to the following generation in the evolutionary process. This basically is based on the theory of “survival of the fittest.”
- (ii) *Crossover.* The crossover increases and maintains the diversity of the entire population over the entire run. This is due to the fact that a population with higher diversity has the ability to explore a wider range of search space.
- (iii) *Mutation.* This enables chromosomes (potential solutions) to jump to a wider range than crossover. Again, mutation also increases the diversity of the entire population.

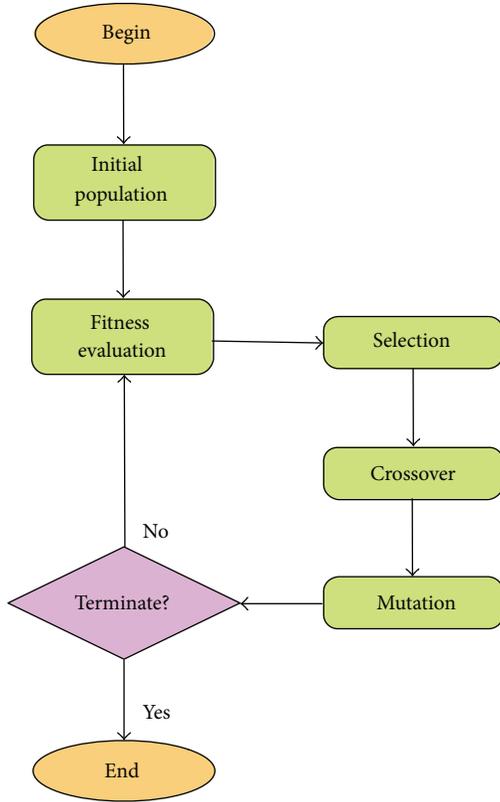


FIGURE 6: GA flowchart.

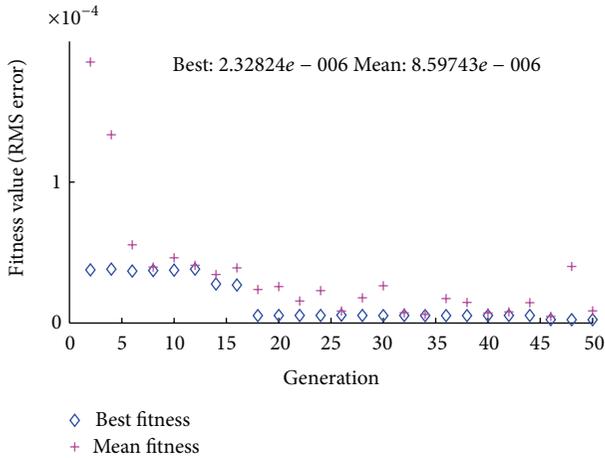


FIGURE 7: The convergence characteristic of GA.

The pseudocode of the GA is presented in Algorithm 1, and relevant figure depicting the algorithm flow is illustrated in Figure 6. To avoid extended discussions on GA, we include here a complete workable source code in the appendix. All parameter settings for the GA are available in the source code.

In the context of Kalman filter, GA is applied to tune the **Q** and **R** parameters, which was explained in detail in Section 3.

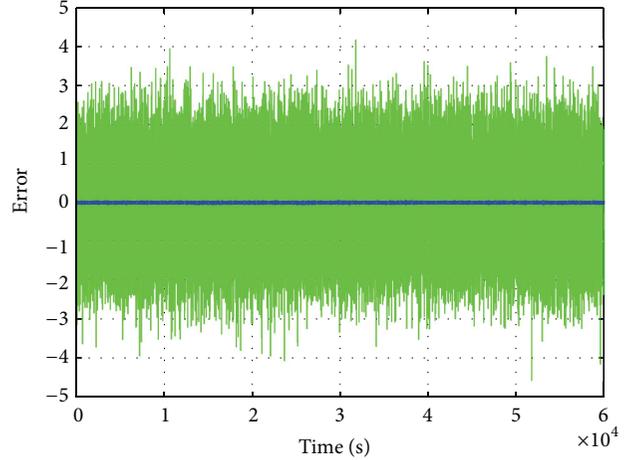


FIGURE 8: The voltage error recorded and depicting measured (green color) and estimated (blue color) errors.

TABLE 2: RMS error recorded during charging operation.

| Output   | <b>Q, R</b>                             | RMS error [V]           |
|--|---|-------------------------|
| Measurement, $y - y_v$                         | 1, 1                                    | 1.0013                  |
| Estimated (KF), $y - y_e$                      | 1, 1                                    | $1.9185 \times 10^{-4}$ |
| Estimated (KF), $y - y_e$<br>(After GA tuning) | 0.012697316315642,<br>2.303940992875865 | $2.3282 \times 10^{-6}$ |

### 5. Results

The program, implemented in MATLAB, is given in the appendix to clarify the results obtained in this work. Take note that the **Q** and **R** mentioned in (41) are both set to a numerical value of one ( $\mathbf{Q} = \mathbf{R} = 1$ ) in the first simulation. The results obtained are tabulated in Table 2. From these results, RMS of the estimated error, which is the error from KF, is far smaller in comparison to the measured error, with values of 1.0013 V and  $1.92 \times 10^{-4}$  V, respectively. This RMS error is further minimized by utilizing  $\mathbf{Q} = 0.012697316315642$  and  $\mathbf{R} = 2.303940992875865$ , found using the GA approach. A graph depicting the convergence characteristic is shown in Figure 7.

The time plot of the estimated error from 0 s to 60000 s is shown in Figure 8, depicting a very small amplitude, in blue color ( $\approx \pm 0.04$  V) along the timeline. This is observed through the zoomed display of the MATLAB graph. On the contrary, the measurement error, portrayed by green color noise in Figure 8, has an absolute magnitude of  $\approx \pm 2$  V.

5.1. *Charging Behaviour.* The charging characteristic is illustrated in Figure 9 whereby the initial terminal voltage  $V_0$  starts from 0 V and rises up to approximately 0.5 V within 60000 seconds (which is 100 minutes). Charging impulses of 1.53 A are applied in this case as shown in Figure 9. As expected, this is a time consuming process as in general case

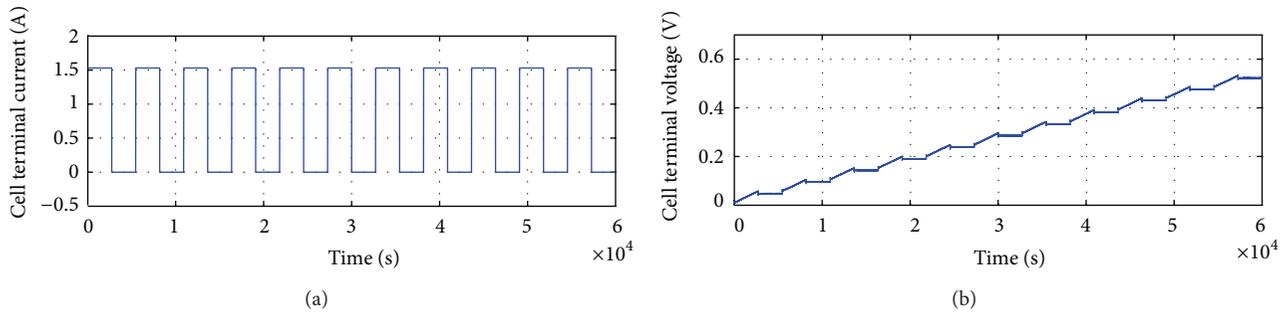


FIGURE 9: Response of RC battery model in terms of  $V_0$  due to charging current  $I = 1.53$  A pulses.

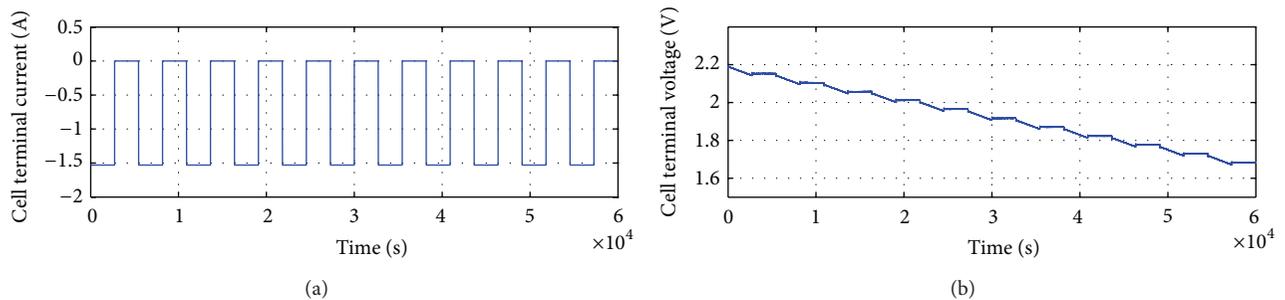


FIGURE 10: Response of RC battery model in terms of  $V_0$  due to discharge current  $I = -1.53$  A pulses.

it may take hours for a car battery (lead acid) to be completely charged.

**5.2. Discharging.** For the discharging process, the initial value of terminal voltage,  $y_0 = V_0$ , is set to 2.2 V in the MATLAB program. Again, in this case, impulses of 1.53 A are applied, but now in negative form. The dynamic behavior of the discharging characteristic is shown in Figure 10. From this figure, it is observed that the discharging process is similar to the charging process, but now with a linearly decreasing terminal voltage slope. The open terminal voltage  $V_0$  drops from 2.2 V to 1.7 V in 60000 seconds (100 minutes); this is similar to the charging process as it takes 100 minutes to reach  $V_0 = 0.5$  V from zero potential.

## 6. Conclusion

In this work, we successfully obtained the state variables of the RC model representing a battery in terms of mathematical derivations. The derivations lead to the conclusion that there exist three state variables relevant to a battery's state-space model. With this state-estimation model, a prominent technique known as the Kalman filter is applied in the aim of estimating state-of-charge for a Battery Management System. From numerical results, the KF is shown to be accurate in predicting the dynamic behavior of the system. This is shown by a very small RMS error of the estimation in comparison to its measurement. The estimated error is further reduced after incorporating the optimized values of  $\mathbf{Q}$  and  $\mathbf{R}$  through the

offline GA approach. As such, the efficacy of such an approach is, thus, validated.

## References

- [1] C. Chen, K. L. Man, T. O. Ting et al., "Design and realization of a smart battery management system," in *Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS '12)*, vol. 2, pp. 1173–1176, March 2012.
- [2] K. L. Man, K. Wan, T. O. Ting et al., "Towards a hybrid approach to SoC estimation for a smart Battery Management System (BMS) and battery supported Cyber-Physical Systems (CPS)," in *Proceedings of the 2nd Baltic Congress on Future Internet Communications (BCFIC '12)*, pp. 113–116, April 2012.
- [3] K. L. Man, C. Chen, T. O. Ting, T. Krilavičius, J. Chang, and S. H. Poon, *Artificial Intelligence Approach to Socestimation for Smart Battery Management System*, 2012.
- [4] P. H. L. Notten and D. Danilov, "From battery modeling to battery management," in *Proceedings of the 33rd International Telecommunications Energy Conference (INTELEC '11)*, pp. 1–8, October 2011.
- [5] D. Benchetrite, F. Mattera, M. Perrin et al., "Optimization of charge parameters for lead acid batteries used in photovoltaic systems," in *Proceedings of the 3rd World Conference on Photovoltaic Energy Conversion*, vol. 2, pp. 2408–2410, Osaka, Japan, May 2003.
- [6] T. O. Ting, K. L. Man, S. Guan, J. K. Seon, T. T. Jeong, and P. W. H. Wong, "Maximum power point tracking (MPPT) via weightless swarm algorithm (WSA) on cloudy days," in *Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS '12)*, pp. 336–339, Kaohsiung, Taiwan, December 2012.

- [7] J. Ma, T. O. Ting, K. L. Man, N. Zhang, S. U. Guan, and P. W. H. Wong, "Parameter estimation of photovoltaic models via cuckoo search," *Journal of Applied Mathematics*, vol. 2013, Article ID 362619, 8 pages, 2013.
- [8] H. Gu, "Mathematical modeling in lead-acid battery development," in *Proceedings of the 6th Annual Battery Conference on Applications and Advances*, pp. 47–56, 1991.
- [9] C. S. Moo, K. S. Ng, Y. P. Chen, and Y. C. Hsieh, "State-of-charge estimation with open-circuit-voltage for lead-acid batteries," in *Proceedings of the 4th Power Conversion Conference (PCC '07)*, pp. 758–762, Nagoya, Japan, April 2007.
- [10] C. C. Chan, E. W. C. Lo, and S. Weixiang, "Available capacity computation model based on artificial neural network for lead-acid batteries in electric vehicles," *Journal of Power Sources*, vol. 87, no. 1, pp. 201–204, 2000.
- [11] P. Singh, C. Fennie Jr., and D. Reisner, "Fuzzy logic modelling of state-of-charge and available capacity of nickel/metal hydride batteries," *Journal of Power Sources*, vol. 136, no. 2, pp. 322–333, 2004.
- [12] I.-S. Kim, "Nonlinear state of charge estimator for hybrid electric vehicle battery," *IEEE Transactions on Power Electronics*, vol. 23, no. 4, pp. 2027–2034, 2008.
- [13] R. Restaino and W. Zamboni, "Comparing particle filter and extended Kalman Filter for battery state-of-charge estimation," in *Proceedings of the 38th Annual Conference on IEEE Industrial Electronics Society (IECON '12)*, pp. 4018–4023, Montreal, Canada, October 2012.
- [14] L. Y. Wang, M. P. Polis, G. G. Yin, W. Chen, Y. Fu, and C. C. Mi, "Battery cell identification and soc estimation using string terminal voltage measurements," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 2925–2935, 2012.
- [15] H. Blanke, O. Bohlen, S. Buller et al., "Impedance measurements on lead-acid batteries for state-of-charge, state-of-health and cranking capability prognosis in electric and hybrid electric vehicles," *Journal of Power Sources*, vol. 144, no. 2, pp. 418–425, 2005.
- [16] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Fluids Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
- [17] R. Faragher, "Understanding the basis of the kalman filter via a simple and intuitive derivation," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 128–132, 2012.
- [18] B. S. Bhangu, P. Bentley, D. A. Stone, and C. M. Bingham, "Nonlinear observers for predicting state-of-charge and state-of-health of lead-acid batteries for hybrid-electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 783–794, 2005.
- [19] G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs—part 1. Background," *Journal of Power Sources*, vol. 134, no. 2, pp. 252–261, 2004.
- [20] T. O. Ting, K. L. Man, N. Zhang, C.-U. Lei, and C. Lu, "State-space battery modeling for smart battery management system," in *Proceedings of The International MultiConference of Engineers and Computer Scientists (IMECS '14)*, Lecture Notes in Engineering and Computer Science, pp. 866–869, Kowloon, Hong Kong, March 2014.
- [21] T. O. Ting, K. L. Man, C.-U. Lei, and C. Lu, "State-of-charge for battery management system via kalman filter," *IAENG Engineering Letters*, vol. 22, no. 2, 2014.
- [22] A. J. Salkind, C. Fennie, P. Singh, T. Atwater, and D. E. Reisner, "Determination of state-of-charge and state-of-health of batteries by fuzzy logic methodology," *Journal of Power Sources*, vol. 80, no. 1, pp. 293–300, 1999.
- [23] H. Saberi and F. R. Salmasi, "Genetic optimization of charging current for lead-acid batteries in hybrid electric vehicles," in *Proceedings of the International Conference on Electrical Machines and Systems (ICEMS '07)*, pp. 2028–2032, October 2007.
- [24] N. K. Medora and A. Kusko, "Dynamic battery modeling of lead-acid batteries using manufacturers' data," in *Proceedings of the 27th International Telecommunication Energy Conference (INTELEC '05)*, pp. 227–232, Berlin, Germany, September 2005.
- [25] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, vol. 412, Addison-Wesley Reading, Menlo Park, Calif, USA, 1989.
- [26] T. O. Ting, S. F. Chien, X.-S. Yang, and N. Ramli, "Resource allocation schemes in energy efficient OFDMA system via genetic algorithm," in *Proceedings of the 19th Asia-Pacific Conference on Communications (APCC '13)*, pp. 723–727, 2013.
- [27] S. F. Chien, T. O. Ting, X.-S. Yang, A. K. N. Ting, and D. W. Holtby, "Optimizing energy efficiency in multi-user ofdma systems with genetic algorithm," in *Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI '13)*, pp. 1330–1334, Greater Noida, India, September 2013.
- [28] T. O. Ting, K. Wan, K. L. Man, and S. Lee, "Space exploration of multi-agent robotics via genetic algorithm," in *Network and Parallel Computing*, vol. 7513 of *Lecture Notes in Computer Science*, pp. 500–507, Springer, 2012.