Kalman-Filter SOC Estimation for LiPB HEV Cells

Gregory L. Plett

Abstract

This paper reports some results relating to a new method of state-of-charge (SOC) estimation for battery-based systems. The particular application in mind is that of a hybrid-electric vehicle (HEV). Battery use in HEV is characterized by very high rates (up to ±25C) and very dynamic rate profiles, which disqualify some simple SOC estimation methods (based only on Coulomb counting, for example).

Kalman filtering methods have been reported for SOC estimation [1–3]. What sets this new method apart from previous results is that the SOC must explicitly be a state in the system state vector. The advantage of this approach is that not only is SOC estimated, but also dynamic error bounds on the estimate are automatically given—a by-product of the Kalman approach. That is, instead of reporting the SOC to the vehicle controller (at some point in time) to be "about" 55%, the algorithm is able to report that the SOC is 55%±7%, for example.

Linear or nonlinear cell models may be used. We find that linear cell models poorly approximate cell dynamics in an HEV application, so we focus on nonlinear models. (The Kalman filter used to estimate SOC is then an "Extended Kalman Filter"). Several candidate cell models are given in a companion paper to be presented at this conference [4].

The cells in question in this research are lithium-ion polymer battery (LiPB) cells jointly developed by LG Chem Ltd. (Daejeon, Korea) and Compact Power Inc. (Colorado, USA), have a nominal capacity of about 8Ah, and are optimized for power-needy applications.

The Kalman filter requires a cell model of the general form:

$$x_{k+1} = f(x_k, u_k)$$
$$v_k = g(x_k, u_k),$$

where x_k is the system state vector at time k, u_k is the system "input" at time k (which may include measurements of battery-pack current, temperature and so forth) and v_k is the cell's terminal voltage (not OCV). In the equation, $f(\ \ )$ and $g(\ \ )$ are (possibly nonlinear) functions, specified by the particular cell model used.

The Kalman filter is initialized with a-priori state estimates when the vehicle is turned on (based on OCV readings and a look-up table, plus self-discharge rate data from the cell model and the prior SOC when the vehicle was turned off). The algorithm then repeatedly updates the state estimate and state-uncertainty (error bound) estimate with each set of new measurements, as the system runs. Results are presented based on lab tests on real cells, which show that excellent SOC estimation with very tight error bounds are obtained whether or not the initial SOC estimate is accurate. Fast recovery is demonstrated when the initial SOC estimate is off by as much as 30%.

Keywords: Algorithmic, calculation, state of charge, lithium polymer, HEV.

1. Introduction

In a battery-powered application it can be critical to know how much capacity remains to perform useful work. This is indicated by the state-of-charge (SOC) of the system. In a battery-electric vehicle, for example, knowing SOC enables the driver to estimate remaining range. In a hybrid-electric vehicle, it enables the vehicle controller to make full use of the regenerative and discharge capacity of the battery at any point in time, without worry of damaging the cells by over- or under-voltage.

A cell's SOC may be compared to the fuel gauge on a gasoline-powered automobile. However, while the fuel level in a gasoline tank may be easily sensed with a flotation device, no simple physical sensor has been developed to measure a battery cell SOC. Instead, we must rely on algorithmic methods based on voltage and current measurements, for example. As all these methods are by nature estimates, the level of uncertainty in the estimate (error bounds) must also be known, either implicitly or explicitly, so that a cell is not over-charged or over-discharged due to a poor estimate.

This paper presents a method to determine cell SOC using an established technique known as Kalman filtering. Since the cell models we employ are nonlinear, an extended Kalman filter (EKF) must be used. Our method is set apart from other Kalman-based SOC estimators in that SOC is explicitly a member of the battery cell "state", and so the EKF method automatically computes error bounds on the SOC. Both the estimate of SOC and its error bounds are dynamic quantities, updated in real time by the algorithm to be developed.

Some alternate methods from the literature for SOC estimation are outlined in a companion paper to be presented at this conference [4]. The same paper details the cell modeling approach we used, which is also important to this work. We focus on the HEV application, although we believe that the results should generalize to other less strenuous battery uses. To summarize, important aspects of SOC estimation required in an HEV application are:

- The cell model must be accurate for all operating conditions. These include: very high rates (many papers consider rates up to about ±1C for portable electronic applications; we need to consider rates up to about ±25C), temperature variation in the automotive range of –30ºC to 50ºC, very dynamic rates (unlike the more benign portable electronic and battery electric vehicle application). Charging (by the engine providing extra power, or by regenerative braking) must be accounted for.
- We require non-invasive methods using only readily available signals. This requirement is imposed by the HEV environment where the battery management system (BMS) has no direct control over current and voltage experienced by the battery pack—this is in the domain of the vehicle controller and inverter. This requirement implies that we must rely on such measurements as instantaneous cell terminal voltage, cell current and cell external temperature.
- Our cell chemistry also limits the range of approaches we might consider. Techniques specific to lead-acid chemistries, for example, are not appropriate for LiPB cells.

We feel that the method proposed in this paper meets these needs better than any other found in the literature.

This paper is organized as follows: First, Kalman-filter and extended-Kalman-filter theory is reviewed. Next, our cell models are introduced. Third, some results are presented; and finally, conclusions are made.

2. Linear Kalman Filtering

The SOC estimation approach presented in this paper is based on Kalman filter theory, which was introduced in 1960 [5]. To illustrate this theory, consider the linear discrete-time system in Figure 1.

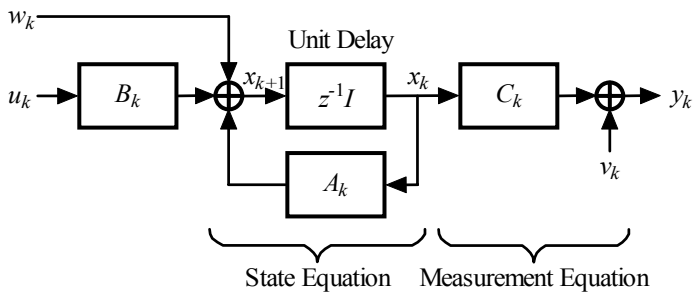

Figure 1: Diagram of system in state-variable form.

A_k, B_k, and C_k are (possibly time-varying) matrices, and $z^{-1}I$ is the unit-delay operator. The deterministic input to the system is u_k, and the output is y_k. There are also two stochastic inputs: process noise w_k (which influences the state x_k) and measurement noise v_k (which does not). As inferred, an internal (hidden) signal is x_k, the system state.

The concept of state is fundamental to Kalman theory. The state of a system is the minimum set of data that is sufficient to uniquely describe its unforced dynamics. Put another way, it is the least amount of information on past behavior required to predict future behavior. Typically, the state is unknown and may not be directly measured. To estimate its value we must use observations on y_k.

We may mathematically describe the system in Figure 1 with two equations. The *state equation* is:

$$x_{k+1} = A_k x_k + B_k u_k + w_k. \tag{1}$$

We assume that w_k is additive, white and Gaussian, with zero mean and covariance matrix:

$$E[w_n w_k^T] = \begin{cases} \Sigma_w, & n = k; \\ 0, & n \neq k. \end{cases}$$

The state equation captures the evolving system dynamics. System stability, dynamic controllability, and sensitivity to disturbance may all be determined from this equation, for example.

The second equation describing the system is the *measurement equation:*

$$y_k = C_k x_k + v_k. \tag{2}$$

The output of the system at discrete-time index k is a linear combination of states, to which is added measurement noise. Again, Kalman filter theory assumes that measurement noise v_k is additive, white and Gaussian with zero mean and covariance matrix:

$$E[v_n v_k^T] = \begin{cases} \Sigma_v, & n = k; \\ 0, & n \neq k. \end{cases}$$

The Kalman filter problem is then: Use the entire observed data $\{u_1, u_2, \cdots, u_k\}$ and $\{y_1, y_2, \cdots, y_k\}$ to find the minimum mean squared error estimate $\hat{x}_k$ of the true state x_k. That is, solve

$$\hat{x}_k = \underset{\hat{x}_k}{\arg\min}\ E[(x_k - \hat{x}_k)^T (x_k - \hat{x}_k) \,|\, u_1, u_2, \cdots, u_k, y_1, y_2, \cdots y_k].$$

The solution to this problem is widely known, and is presented in Table 1. The heart of the solution is a set of recursive relationships that involve both an estimate of the state itself, and also the covariance matrix $\Sigma_{e,k}$ of the state estimate error $x_k - \hat{x}_k$. The covariance matrix indicates the uncertainty of the state estimate, and may be used to generate error bounds on the estimate. A "large" $\Sigma_{e,k}$ (one with large singular values) indicates a high level of uncertainty in the state estimate; a "small" $\Sigma_{e,k}$ (one with small singular values) indicates confidence in the estimate.

Table 1: Summary of the Kalman filter from reference [6].

State-space model:

$$x_{k+1} = A_k x_k + B_k u_k + w_k$$
$$y_k = C_k x_k + v_k,$$

where w_k and v_k are independent, zero-mean, Gaussian noise processes of covariance matrices Σ_w and Σ_v, respectively.

Initialization: For $k = 0$, set

$$\hat{x}_0(+) = E[x_0]$$
$$\Sigma_{x,0}(+) = E[(x_0 - E[x_0])(x_0 - E[x_0])^T].$$

Computation: For $k = 1,2,\ldots,$ compute:

State estimate propagation

$$\hat{x}_k(-) = A_{k-1}\hat{x}_{k-1}(+) + B_{k-1}u_{k-1}.$$

Error covariance propagation

$$\Sigma_{e,k}(-) = A_{k-1}\Sigma_{e,k-1}(+)A_{k-1}^T + \Sigma_w.$$

Kalman gain matrix

$$L_k = \Sigma_{e,k}(-)C_k^T[C_k\Sigma_{e,k}(-)C_k^T + \Sigma_v]^{-1}.$$

State estimate update

$$\hat{x}_k(+) = \hat{x}_k(-) + L_k[y_k - C_k\hat{x}_k(-)].$$

Error covariance update

$$\Sigma_{e,k}(+) = (I - L_k C_k)\Sigma_{e,k}(-).$$

The Kalman filter is initialized with the best available information on the state and error covariance.

$$\hat{x}_0(+) = E[x_0]$$
$$\Sigma_{x,0}(+) = E[(x_0 - E[x_0])(x_0 - E[x_0])^T].$$

Often, this initialization must be performed in an ad-hoc manner, but the Kalman filter quickly converges on the true values as it runs.

Following initialization, the Kalman filter repeatedly performs two steps. First, it *predicts* the value of the next state, system output, and error covariance. Secondly, using a measurement of the system output, it *corrects* the present state estimate. The predicted state and error covariance estimates at time index *k*, but before the measurement, are denoted $\hat{x}_k(-)$ and $\Sigma_{e,k}(-)$, respectively. The corrected estimates following the measurement are denoted $\hat{x}_k(+)$ and $\Sigma_{e,k}(+)$.

The prediction step is accomplished by propagating the system input through the system dynamics:

$$\hat{x}_{k+1}(-) = A_k\hat{x}_k(+) + B_k u_k.$$

The state uncertainty is also updated:

$$\Sigma_{e,k+1}(-) = A_k \Sigma_{e,k}(+) A_k^T + \Sigma_w.$$

If the system is stable, the first term is contractive, reducing uncertainty. The process noise Σ_w term always increases uncertainty.

The state correction step is:

$$\hat{x}_k(+) = \hat{x}_k(-) + L_k[y_k - C_k \hat{x}_k(-)].$$

That is, the new state estimate equals the predicted state estimate plus a weighted correction factor. The term in the square brackets is equal to the measured cell voltage minus the predicted cell voltage. This term may be non-zero due to measurement noise, an incorrect state estimate $\hat{x}_k(-)$, or an inaccurate cell model. It represents the "new information" in the measurement, and is often called the "innovation" process for this reason. If the innovation is large, the state update tends to be large. If the innovation is small, the state update tends to be small.

The innovation is weighted by the *Kalman gain* L_k.

$$L_k = \Sigma_{e,k}(-) C_k^T [C_k \Sigma_{e,k}(-) C_k^T + \Sigma_v]^{-1}.$$

If the present state estimate is very uncertain, $\Sigma_{e,k}$ is large and L_k is large, forcing a large update. If the present state estimate is certain, L_k is small, and the state-estimate update is small. Also, if sensor noise is large, Σ_v is large, causing L_k to be small and the update to be small.

The covariance correction step is:

$$\Sigma_{e,k}(+) = \Sigma_{e,k}(-) - L_k C_k \Sigma_{e,k}(-).$$

The state uncertainty always decreases due to the new information provided by the measurement.

3. Extended Kalman Filtering

The Kalman filter is the optimum state estimator for a linear system with the assumptions as described. If the system is nonlinear, then we may use a linearization process at every time step to approximate the nonlinear system with a linear time-varying (LTV) system. This LTV system is then used in the Kalman filter, resulting in an extended Kalman filter (EKF) on the true nonlinear system. Note that although EKF is not necessarily optimal, it often works very well.

The nonlinear system may be modeled as

$$x_{k+1} = f(x_k, u_k) + w_k \quad (3)$$

$$y_k = g(x_k, u_k) + v_k. \quad (4)$$

As before, w_k and v_k are zero mean white Gaussian stochastic processes with covariance matrices Σ_w and Σ_v, respectively. Now, $f(\ )$ is a nonlinear transition matrix function and $g(\ )$ is a nonlinear measurement matrix.

At each time step, $f(\ )$ and $g(\ )$ are linearized by a Taylor-series expansion.

$$f(x_k, u_k) \approx f(\hat{x}_k, u_k) + \underbrace{\left.\frac{\partial f(x_k, u_k)}{\partial x}\right|_{x_k = \hat{x}_k}}_{\text{Defined as } A_k} (x_k - \hat{x}_k)$$

$$g(x_k, u_k) \approx g(\hat{x}_k, u_k) + \underbrace{\left.\frac{\partial g(x_k, u_k)}{\partial x}\right|_{x_k = \hat{x}_k}}_{\text{Defined as } C_k} (x_k - \hat{x}_k).$$

So, we have the linearized system:

Table 2: Summary of the extended Kalman filter from reference [6].

State-space model:

$$x_{k+1} = f(x_k, u_k) + w_k$$
$$y_k = g(x_k, u_k) + v_k,$$

where w_k and v_k are independent, zero-mean, Gaussian noise processes of covariance matrices Σ_w and Σ_v, respectively.

Definitions:

$$A_k = \left.\frac{\partial f(x_k, u_k)}{\partial x}\right|_{x_k = \hat{x}_k(+)}$$

$$C_k = \left.\frac{\partial g(x_k, u_k)}{\partial x}\right|_{x_k = \hat{x}_k(-)}$$

Initialization: For $k = 0$, set

$$\hat{x}_0(+) = E[x_0]$$
$$\Sigma_{x,0}(+) = E[(x_0 - E[x_0])(x_0 - E[x_0])^T].$$

Computation: For $k = 1,2,\ldots,$ compute:

State estimate propagation

$$\hat{x}_k(-) = f(\hat{x}_{k-1}(+), u_{k-1}).$$

Error covariance propagation

$$\Sigma_{e,k}(-) = A_{k-1}\Sigma_{e,k-1}(+)A_{k-1}^T + \Sigma_w.$$

Kalman gain matrix

$$L_k = \Sigma_{e,k}(-)C_k^T[C_k\Sigma_{e,k}(-)C_k^T + \Sigma_v]^{-1}.$$

State estimate update

$$\hat{x}_k(+) = \hat{x}_k(-) + L_k[y_k - g(\hat{x}_k(-), u_k)].$$

Error covariance update

$$\Sigma_{e,k}(+) = (I - L_k C_k)\Sigma_{e,k}(-).$$

$$x_{k+1} \approx A_k x_k + \underbrace{[f(\hat{x}_k, u_k) - A_k\hat{x}_k]}_{\text{Not a function of } x_k} + w_k$$
$$y_k \approx C_k x_k + \underbrace{[g(\hat{x}_k, u_k) - C_k\hat{x}_k]}_{\text{Not a function of } x_k} + v_k.$$

This leads to the EKF algorithm summarized in Table 2.

4. Cell Modeling

To use Kalman filter methods to estimate battery cell SOC, we must create a mathematical state-space model of the cell. Furthermore, if SOC is explicitly a member of the state vector x_k, then both SOC and its uncertainty (error bounds) are automatically estimated by the Kalman filter. To our knowledge, this approach is novel. Other Kalman-filter-based SOC estimation methods [1–3] use circuit models of a cell. A capacitor voltage represents open-circuit voltage (OCV), and SOC is indirectly inferred from OCV. Error bounds are not available.

To include SOC as a model state, we first make some careful definitions:

Definition: The cell high operational voltage limit is called v_h. Here, we may use v_h=4.2V.

Definition: The cell low operational voltage limit is called v_l. Here, we may use v_l=3.0V.

Definition: A cell is *fully charged* when its voltage reaches $v=v_h$ after being charged at infinitesimal current levels.

Definition: A cell is *fully discharged* when its voltage reaches $v=v_l$ after being drained at infinitesimal current levels.

Definition: The *capacity* of a cell is the maximum number of Ampere-hours that can be drawn from the cell before it is fully discharged, at room temperature (25°C), starting with the cell fully charged.

Definition: The *nominal capacity* of the cell is the number of Ampere-hours that can be drawn from the cell at room temperature at the C/40 rate, starting with the cell fully charged.

Definition: The SOC of the cell is the ratio of the remaining capacity to the nominal capacity of the cell, where the remaining capacity is the number of amp-hours that can be drawn from the cell at room temperature at the C/40 rate.

Then, we may derive a recurrent relationship:

$$\mathrm{SOC}_{k+1} = \mathrm{SOC}_k - \frac{\eta(i_k)\, i_k\, \Delta t}{C}, \tag{5}$$

SOC is already in a state-space form, with i_k the model input, $\eta(i_k)$ the Coulombic efficiency, and Δt the sampling period. The minimum requirement to make a full state-space form is an output equation. Three alternate parameterized forms and methods to identify the parameters are described in reference [4]. These are:

Combined one-state model: Commonalities between the "Shepherd", "Unnewehr universal", "Nernst" and a modified Nernst model are exploited to derive a "Combined model". The resulting output equation is:

$$y_k = K_0 - R\, i_k - K_1/\mathrm{SOC}_k - K_2\, \mathrm{SOC}_k + K_3 \ln(\mathrm{SOC}_k) + K_4 \ln(1-\mathrm{SOC}_k). \tag{6}$$

Here, y_k is the cell terminal voltage, R is the cell internal resistance (different values may be used for charge/discharge and at different SOC levels if desired), and K_0 through K_4 are constants chosen to make the model fit the data well. The unknown quantities in (6) may be estimated using any system identification procedure. Equation (5) is used as the state equation.

Filter-state model: The combined model of (5) and (6) may be very quickly identified and implemented. Its serious limitation is that it omits any description of cell relaxation. Since the cell model must accurately predict its behavior in a dynamic HEV environment, we find it is essential to include relaxation effects.

In a state-variable model, dynamics are described by the state equation (1) or (3). Therefore, to include relaxation effects, we must augment the state vector with additional filter states. We choose to implement filtered versions of SOC and the input current. The resulting model is then:

$$x_{k+1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ w_1 & w_2 & 0 & 0 \\ 0 & 0 & w_4 & w_5 \\ 0 & 0 & -w_5 & w_4 \end{bmatrix} x_k + \begin{bmatrix} -1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} I_k^{\mathrm{mod}} \\ w_3 \end{bmatrix}$$

$$y_k = w_6 + w_7 I_k^{\mathrm{mod}} + \frac{w_8}{x_{k,1} + w_9} + \begin{bmatrix} w_{10} & 10 & w_{11} & w_{12} \end{bmatrix} x_k,$$

where $I_k^{\text{mod}} = \eta(i_k)\,|i_k|^n \Delta t/C_p$, n is the Peukert exponent and C_p is the Peukert capacity. The first state of x_k (that is, $x_{k,1}$) is SOC, as before. The output y_k is terminal voltage, as before. The parameters of the model are found by system identification using measured cell data. We found that the model was able to predict cell behavior best when different sets of parameters were used for different levels of input current i_k.

Radial-Basis-Function (RBF) model: Adding linear filter states to the model does improve its ability to predict a cell's behavior. However, as the LiPB cells themselves are nonlinear systems, we can improve even further by considering a fully nonlinear cell dynamic model. For this purpose, we will use radial-basis-function (RBF) networks and a black-box system identification procedure.

An RBF network makes a local approximation of the function it models. It computes its output as a weighted sum of (hyper) Gaussian shapes. Specifically, it computes the function

$$y_k = \sum_{j=1}^{N} w_j \exp\left(-\frac{1}{\sigma_j^2}\left\|u_k - t_j\right\|^2\right) + w_{N+1},$$

where N is the number of bases, w_j is the weight connecting the jth basis function to the output, σ_j is the "standard deviation" or width parameter of the jth basis function, u_k is the vector input to the network, and t_j is the center of the jth basis function. Here, u_k includes the states of the system: *e.g.,* $x_k = \left[y_{k-1}, \text{SOC}_k\right]^T$ as well as the cell current i_k.

In order to use EKF with these three models, we must determine the linearized versions. For the combined one-state model:

$$A_k = \left.\frac{\partial f(x_k, u_k)}{\partial x_k}\right|_{x_k=\hat{x}_k} = 1, \qquad \text{and} \qquad C_k = \left.\frac{\partial g(x_k, u_k)}{\partial x_k}\right|_{x_k=\hat{x}_k} = \frac{K_1}{\hat{x}_k^2} - K_2 + \frac{K_3}{\hat{x}_k} - \frac{K_4}{1-\hat{x}_k}.$$

For the filter-state model,

$$A_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ w_1 & w_2 & 0 & 0 \\ 0 & 0 & w_4 & w_5 \\ 0 & 0 & -w_5 & w_4 \end{bmatrix}, \qquad \text{and} \qquad C_k = \begin{bmatrix} w_{10} - \dfrac{w_8}{(\hat{x}_{k,1} + w_9)^2} & 10 & w_{11} & w_{12} \end{bmatrix}.$$

For the RBF model, if $x_k = \left[y_{k-1}, \text{SOC}_k\right]^T$ and the exogenous input is i_k, then

$$x_{k+1} = \begin{bmatrix} \text{RBF}(x_k, i_k) \\ \begin{bmatrix} 0 & 1 \end{bmatrix} x_k \end{bmatrix} + \begin{bmatrix} 0 \\ -\eta(i_k) i_k \Delta t / C \end{bmatrix}$$

and

$$\left.\frac{\partial f}{\partial x_k}\right|_{x_k=\hat{x}_k} = \begin{bmatrix} \partial \text{RBF}/\partial x_k \,|_{x_k=\hat{x}_k} \\ \begin{bmatrix} 0 & 1 \end{bmatrix} \end{bmatrix}, \qquad \text{and} \qquad \left.\frac{\partial g}{\partial x_k}\right|_{x_k=\hat{x}_k} = \frac{\partial \text{RBF}}{\partial x_k}\,|_{x_k=\hat{x}_k}.$$

So, it only remains to find $\partial \text{RBF}/\partial x_k$. If

$$y_k = \sum_{i=1}^{N} w_i \underbrace{\exp\left(-s_i \sum_{j=1}^{n_x} (u_j - t_{ji})^2\right)}_{\phi_i} + w_0,$$

and if $u_j = \left[x_k^T, i_k\right]^T$, then we may find

$$\frac{\partial y_k}{\partial u_k} = \sum_{i=1}^{N} w_i \phi_i \frac{\partial\left(-s_i \sum_{j=1}^{n_x} (u_j - t_{ji})^2\right)}{\partial u_k} = -2 \sum_{i=1}^{N} w_i \phi_i s_i (u_k - t_{ki}),$$

and $\partial \mathrm{RBF} / \partial x_k$ comprises the top two rows of $\partial y_k / \partial u_k$.

With all of the derivatives determined, the algorithm is nearly complete. It remains only to determine an appropriate method to initialize the state and uncertainties when the vehicle is started. That is, we need a method to propagate SOC and Σ_e across the interval between key-off and key-on. For this we employ the SOC estimate and Σ_e value saved when the vehicle was previously turned off, the period of time the vehicle was off, and a very simple cell self-discharge model. Our empirical data indicates that SOC decays exponentially, allowing us to create a continuous-time state-space model for self discharge. Let $x(t)$ be the state of charge as a function of self-discharge time. Then,

$$\dot{x}(t) = -\alpha x(t) + w(t)$$
$$y(t) = f(x(t)) + v(t),$$

where α is the rate of SOC decay, $f(\ )$ is OCV as a function of SOC, w is small and v depends on the period since key-off. We can form a discrete-time version of this model. Let T be the period that the vehicle is off. Then,

$$x_{k+1} = e^{-\alpha T} x_k + w_k$$
$$y_k = \mathrm{OCV}(x_k) + v_k.$$

Applying the EKF to this system estimates SOC and Σ_e. SOC_k, SOC_{k-1}, and SOC_{k-2} are initialized to the same value, and their corresponding Σ_e entries are initialized to the same value. Filter states are initialized to zero with $\Sigma_e = 0$ and y_{k-1} is initialized to the measured voltage y_k, with $\Sigma_e = \Sigma_v$.

5. Kalman Verification

In order to compare SOC estimation abilities of the proposed models using a Kalman filtering technique, we gathered data from some prototype LiPB cells. We used a Tenny thermal chamber set at 25°C and an Arbin cell cycler. In all cases, the cells were fully charged before the tests began. Pulsed discharge cycles punctuated with rest intervals were followed by pulsed charge cycles, again with rest periods. Data points (including voltage, current Ah discharged and Ah charged) were collected once per second.

The data was used to identify parameters of the three cell models. Then, the models were used with the Kalman filter to estimate SOC. Figures 2–4 show a comparison between measured SOC (computed using the Ah discharged and Ah charged reported by the Arbin) and estimated SOC for three representative tests: pulsed ±1C rates, pulsed ±2C rates and pulsed ±4C rates. In all plots, the red line is the true SOC, the blue line is the model's estimate, and the green dotted lines delineate the confidence region of the estimate. Plots are given for the case when the Kalman filter is initialized with the true initial SOC (left column) and when the filter is initialized with an incorrect estimate (right column).

Figure 2 shows results of the combined one-state model. Cell relaxation effects are not captured by this model, and this translates into poor SOC estimation. Error bounds (computed under the assumption of a perfect cell model) do not accurately describe the SOC estimation error. However, if the initial SOC estimate is inaccurate, the Kalman filter corrects it almost instantly.

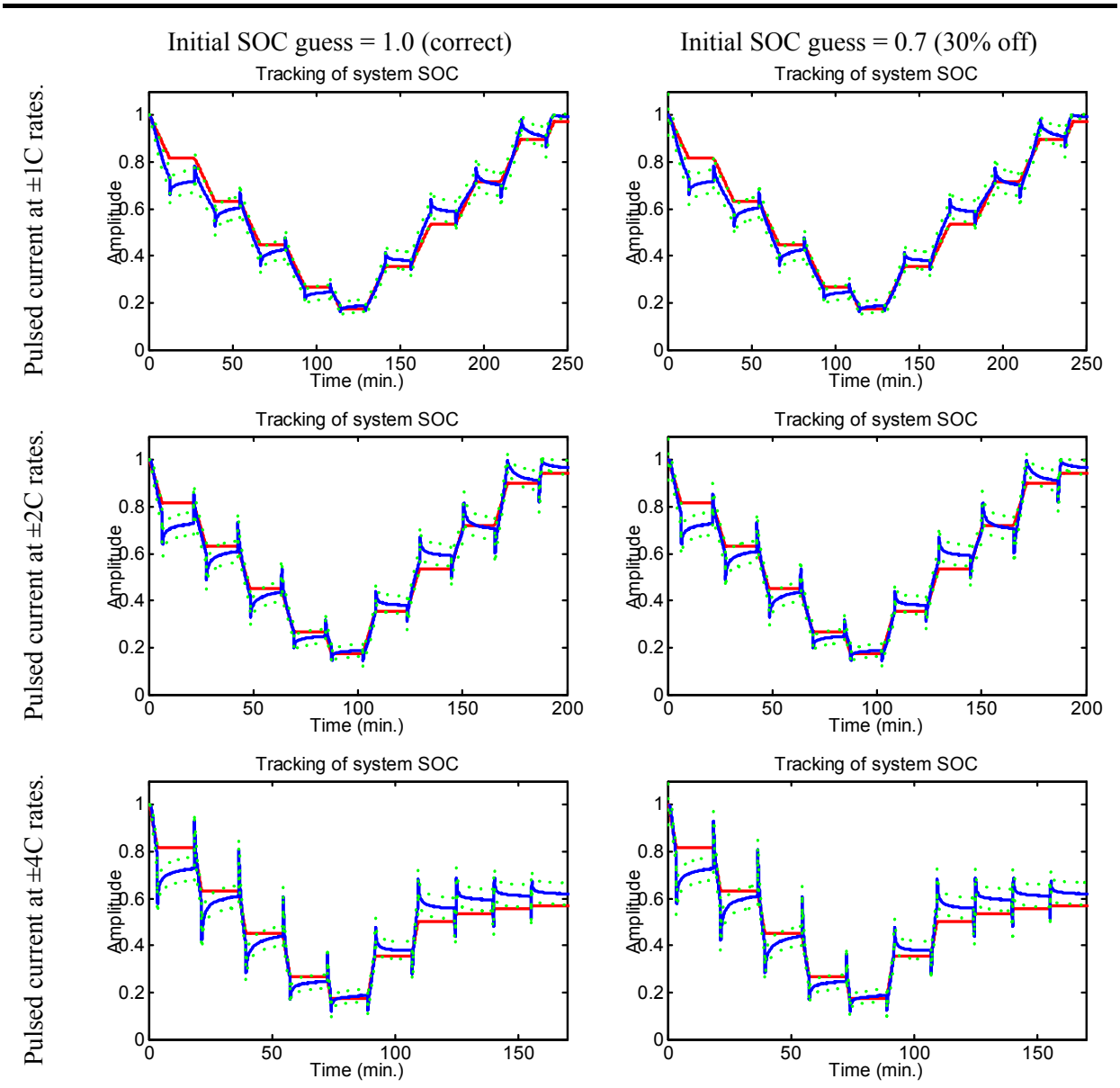

Figure 2: One-state tracking of SOC for correct initial estimate and incorrect initial estimate.

Figure 3 shows results of the filter-state model. SOC is very accurately estimated if the initial estimate is accurate. However, a poor initial guess is not quickly corrected. Error bounds are accurate only when the initial SOC estimate is precise.

Figure 4 shows results using an RBF model. SOC is very accurately estimated both when the initial estimate is accurate and when it is not. The error bounds are accurate in both cases. Space does not permit a detailed presentation of results for more dynamic tests, but our experience is that the RBF model accurately estimates SOC even in these cases.

6. Conclusion

This paper has presented an SOC estimation technique based on extended Kalman filtering. A key feature of the method is that both an estimate of SOC and error bounds of the estimate are predicted. This can allow more complete use of the battery pack, without fear of over- or under-charging cells.

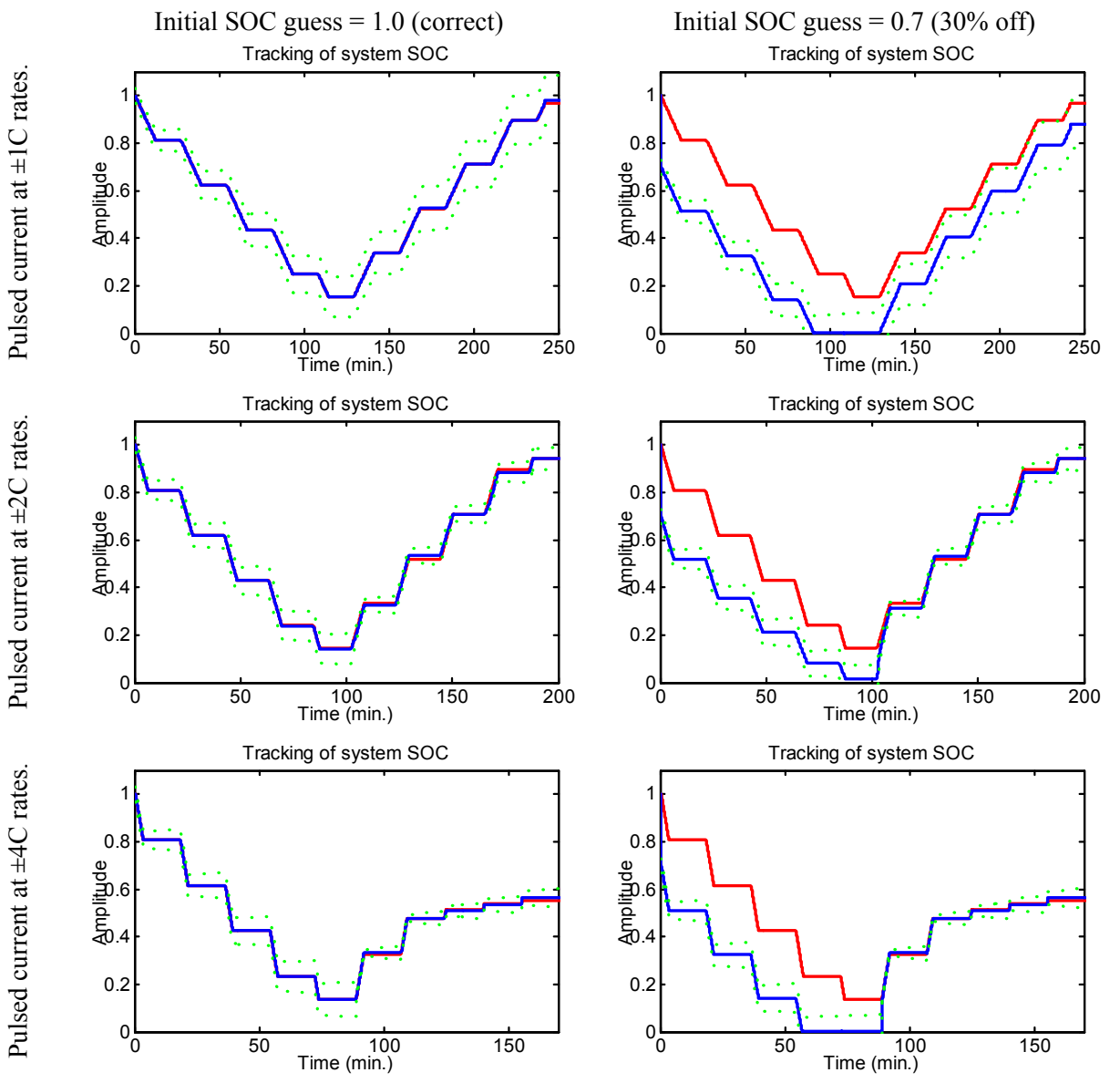

Figure 3: Filter-state tracking of SOC for correct initial estimate and incorrect initial estimate.

Some cell models from reference [4] were used as a basis for this prediction. It is demonstrated that very accurate SOC estimation may be achieved, whether or not the initial state estimate is accurate.

7. References

[1] S. Piller, M. Perrin and A. Jossen, *Methods for state-of-charge determination and their applications*, Journal of Power Sources, vol. 96 (2001), pp. 113–120.

[2] C. Barbier, H. Meyer, B. Nogarede and S. Bensaoud, *A battery state of charge indicator for electric vehicle*, Proceedings of the Institution of Mechanical Engineers. Automotive Electronics. International Conference, 17–19 May 1994, pp. 29–34, London, UK.

[3] W. Steffens and P. Lürkens, *Ladezustandsschätzung von bleibatterien mit hilfe des Kalman-filters*, etzArchiv, vol. 8, no. 7, 1986, pp. 231–6. (In German. English title: *State of charge estimation of lead-acid batteries using a Kalman filtering technique*).

[4] G. Plett, *LiPB dynamic cell models for Kalman-filter SOC estimation*, Proceedings of the 19th International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium & Exhibition (EVS19), 19–23 October 2002, Busan, Korea.

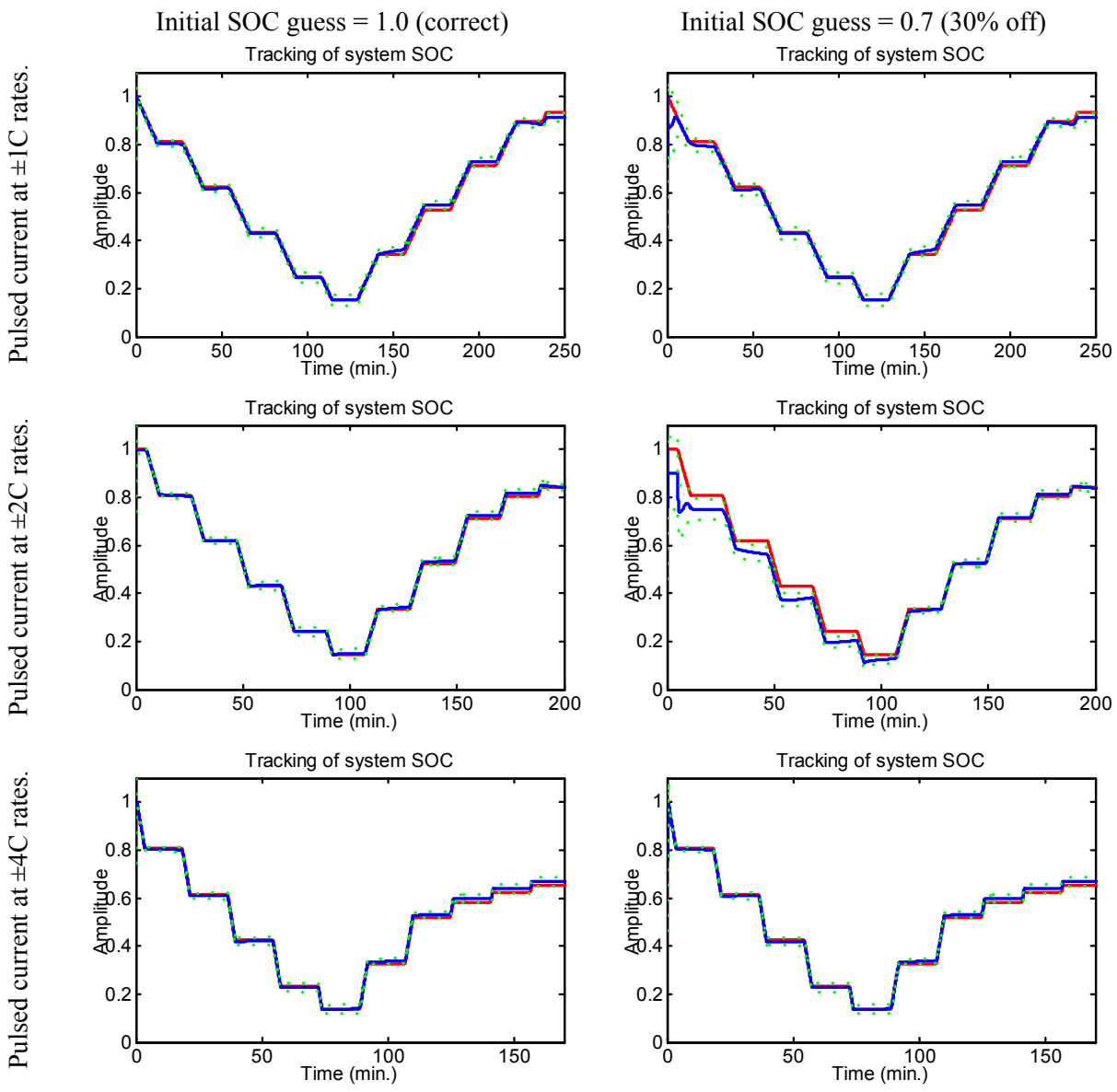

Figure 4: RBF tracking of SOC for correct initial estimate and incorrect initial estimate.

[5] R. Kalman, *A new approach to linear filtering and prediction problems*, Transactions of the ASME—Journal of Basic Engineering, vol. 82 (Series D): March 1960, pp. 35–45.

[6] S. Haykin (Ed.), *Kalman Filtering and Neural Networks*, (Wiley Inter-Science: New York), 2001.

8. Affiliation

Dr. Gregory L. Plett, *Assistant Professor,*
Dept. of Electrical and Computer Engineering, University of Colorado at Colorado Springs,
1420 Austin Bluffs Parkway, P.O. Box 7150, Colorado Springs, CO 80933–7150 USA
Tel: +1–719–262–3468, Fax: +1–719–262–3589, E-mail: glp@eas.uccs.edu,
URL: http://mocha-java.uccs.edu, *and consultant to*
Compact Power Inc., 1200 S. Synthes Ave., Monument, CO 80132 USA
Tel: +1–719–488–1600x134, Fax: +1–719–487–9485, E-mail: gplett@compactpower.com.
URL: http://www.compactpower.com/.