Comparison of Prognostic Algorithms for Estimating Remaining Useful Life of Batteries

Bhaskar Saha[1], Kai Goebel[2] and Jon Christophersen[3]

[1]Research Programmer, Mission Critical Technologies, NASA Ames Research Center, Moffett Field, CA 94035, USA
[2]Senior Scientist, NASA Ames Research Center, Moffett Field, CA 94035, USA
[3]Idaho National Laboratory, P.O. Box 1625, Idaho Falls, ID 83415, USA

The estimation of remaining useful life (RUL) of a faulty component is at the center of system prognostics and health management. It gives operators a potent tool in decision making by quantifying how much time is left until functionality is lost. RUL prediction needs to contend with multiple sources of errors like modeling inconsistencies, system noise and degraded sensor fidelity, which leads to unsatisfactory performance from classical techniques like Autoregressive Integrated Moving Average (ARIMA) and Extended Kalman Filtering (EKF). Bayesian theory of uncertainty management provides a way to contain these problems. The Relevance Vector Machine (RVM), the Bayesian treatment of the well known Support Vector Machine (SVM), a kernel-based regression/classification technique, is used for model development. This model is incorporated into a Particle Filter (PF) framework, where statistical estimates of noise and anticipated operational conditions are used to provide estimates of RUL in the form of a probability density function (PDF). We present here a comparative study of the above mentioned approaches on experimental data collected from Li-ion batteries. Batteries were chosen as an example for a complex system whose internal state variables are either inaccessible to sensors or hard to measure under operational conditions. In addition, battery performance is strongly influenced by ambient environmental and load conditions.

Keywords: Battery prognostics, remaining useful life, uncertainty management, Autoregressive Integrated Moving Average, Extended Kalman Filtering, Relevance Vector Machine, Particle Filter.

1. Introduction

Batteries form a core component of many machines and are often times critical to the well being and functional capabilities of the overall system. Failure of a battery could lead to reduced performance, operational impairment and even catastrophic failure, especially in aerospace systems. A case in point is NASA's Mars Global Surveyor which stopped operating in November 2006. Preliminary investigations revealed that the spacecraft was commanded to go into a safe mode, after which the radiator for the batteries was oriented towards the sun. This increased the temperature of the batteries and they lost their charge capacity in short order. This scenario, although drastic, is not the only one of its kind in aerospace applications. In fact, accurate estimates of the state-of-charge (SOC), the state-of-health (SOH) and state-of-life (SOL) for batteries would provide a significant value addition to the management of any operation involving electrical systems.

The phrase "battery health monitoring" has a wide variety of connotations, ranging from intermittent manual measurements of voltage and electrolyte specific gravity to fully automated online supervision of various measured and estimated battery parameters. In the aerospace application domain, researchers have looked at the various failure

Address for correspondence: Bhaskar Saha, NASA Ames Research Center, MS 269-3, Moffett Field, CA 94035, USA. E-mail: bsaha@email.arc.nasa.gov

modes of the battery subsystems. Different diagnostic methods have been evaluated, like discharge to a fixed cut-off voltage, open circuit voltage, voltage under load and electrochemical impedance spectrometry (EIS) (Vutetakis and Viswanathan, 1995). In the field of telecommunications, people have looked to combine conductance technology with other measured parameters like battery temperature/differential information and the amount of float charge (Cox and Perez-Kite, 2000).

Other works have concentrated more on the prognostic perspective rather than the diagnostic one. Statistical parametric models have been built to predict time to failure (Jaworski, 1999). Electric and hybrid vehicles have been another fertile area for battery health monitoring (Meissner and Richter, 2003). Impedance spectrometry has been used to build battery models for cranking capability prognosis (Blanke et al., 2005). State estimation techniques, like the Extended Kalman Filter (EKF), have been applied for real-time prediction of SOC and SOH of automotive batteries (Bhangu et al., 2005). A decision-level fusion of data-driven algorithms, like Autoregressive Integrated Moving Average (ARIMA) and neural networks, have been investigated for both diagnostics and prognostics (Kozlowski, 2003). As the popular cell chemistries changed from lead acid to nickel metal hydride to lithium ion, cell characterization efforts have kept pace. Dynamic models for the lithium ion batteries that take into consideration nonlinear equilibrium potentials, rate and temperature dependencies, thermal effects and transient power response have been built (Gao, Liu and Dougal, 2002). Not withstanding the body of work done before, it still remains notoriously difficult to accurately predict the end-of-life of a battery from SOC and SOH estimates under environmental and load conditions different from the training data set. This is where advanced regression, classification and state estimation algorithms have an important role to play.

Support Vector Machines (SVMs) (Vapnik, 1995) are a set of related supervised learning methods used for classification and regression that belong to a family of generalized linear classifiers. The Relevance Vector Machine (RVM) (Tipping, 2000) is a Bayesian form representing a generalized linear model of identical functional form of the SVM. Bayesian techniques also provide a general rigorous framework for dynamic state estimation problems. The core idea is to construct a probability density function (PDF) of the state based on all available information. For a linear system with Gaussian noise, the method reduces to the Kalman filter. The state space PDF remains Gaussian at every iteration and the filter equations propagate and update the mean and covariance of the distribution.

2. Algorithms

2.1 Autoregressive Integrated Moving Average

The Autoregressive Integrated Moving Average (ARIMA) modeling technique is a generalization of Autoregressive Moving Average or ARMA (Box and Jenkins, 1976). These models are fitted to time series data either to better characterize the data or to predict future points in the series. Theoretically, any time series which contains no trend or from which trend has been removed can be represented as consisting of two parts, a self-deterministic part, and a disturbance component. The self-deterministic part of the series can be forecasted from its own past by an autoregressive (AR) model with enough number of terms, p, while the disturbance component (the residuals from the AR model) can be modeled by a moving average (MA) with a large enough number of elements, q. Given a real valued time series x_k where k is an integer index, an ARMA(p,q) model is given by:

$$\left(1 - (\alpha_1 L^1 + \alpha_2 L^2 + \ldots + \alpha_p L^p)\right) x_k = \left(1 + (\beta_1 L^1 + \beta_2 L^2 + \ldots + \beta_q L^q)\right) \rho_k \tag{1}$$

where, $L \mid L^i x_k \equiv x_{k-i}$ is the lag operator, α's are the parameters of the AR part of the model, and β's are the parameters of the MA part. The error terms ρ_k are generally assumed to be independent, identically distributed variables sampled from a normal distribution with zero mean.

ARMA modeling assumes the time series is weakly stationary. It is possible to handle non-stationary data by differencing it to a sufficient degree. If the series is differenced d times to achieve stationarity, the model is classified as ARIMA(p,d,q) and is described as:

$$\left(1 - (\alpha_1 L^1 + \alpha_2 L^2 + \ldots + \alpha_p L^p)\right)(1 - L)^d x_k = \left(1 + (\beta_1 L^1 + \beta_2 L^2 + \ldots + \beta_q L^q)\right) \rho_k \tag{2}$$

The basic steps of ARIMA modeling are:

- *Identification:* Ensure that the series is sufficiently stationary (free of trend and seasonality) by differencing d times, and specify the appropriate number of autoregressive terms, p, and moving average terms, q from the autocorrelation function (acf) and partial autocorrelation function (pacf) correlograms.
- *Estimation:* Estimate the parameters (α's and β's) of AR and MA terms, usually by regression analysis.

- *Verification:* Ascertain whether the model fits the historical data well enough. The model is used to forecast all of the extant values in the series. It is said to fit the series well if the differences between the actual values and the forecasted values are small enough and sufficiently random.

2.2 Relevance Vector Machine

In a given classification problem, the data points may be multidimensional (say n_{dim}). The task is to separate them by an n_{dim}-1 dimensional hyperplane. This is a typical form of linear classifier. There are many linear classifiers that might satisfy this property. However, an optimal classifier would additionally create the maximum separation (margin) between the two classes. Such a hyperplane is known as the maximum-margin hyperplane and such a linear classifier is known as a maximum-margin classifier. Nonlinear kernel functions can be used to create nonlinear classifiers (Boser, Guyon, and Vapnik, 1992). This allows the algorithm to fit the maximum-margin hyperplane in the transformed feature space, though the classifier may be nonlinear in the original input space.

This technique was also extended to regression problems in the form of support vector regression (SVR) (Drucker et al., 1997). Regression can essentially be posed as an inverse classification problem where, instead of searching for a maximum margin classifier, a minimum margin fit needs to be found. Although, SVM is a state-of-the-art technique for classification and regression, it suffers from a number of disadvantages, one of which is the lack of probabilistic outputs that make more sense in health monitoring applications. The RVM attempts to address these very issues in a Bayesian framework. Besides the probabilistic interpretation of its output, it uses a lot fewer kernel functions for comparable generalization performance.

This type of supervised machine learning starts with a set of input vectors $\{\mathbf{t}\}_{n=1\ldots N}$ and their corresponding targets $\{\theta\}_{n=1\ldots N}$. The aim is to learn a model of the dependency of the targets on the inputs in order to make accurate predictions of θ for unseen values of $\mathbf{t}$. Typically, the predictions are based on some function $F(\mathbf{t})$ defined over the input space, and learning is the process of inferring the parameters of this function. In the context of SVM, this function takes the form:

$$F(\mathbf{t};\mathbf{w}) = \sum_{i=1}^{N} w_i K(\mathbf{t},\mathbf{t}_i) + w_0, \tag{3}$$

where, $\mathbf{w} = (w_1, w_2, \ldots, w_M)^{\mathrm{T}}$ is a weight vector and $K(\mathbf{t}, \mathbf{t}_i)$ is a kernel function. In the case of RVM, the targets are assumed to be samples from the model with additive noise:

$$\theta_n = F(\mathbf{t}_n;\mathbf{w}) + \varepsilon_n, \tag{4}$$

where, ε_n are independent samples from some noise process (Gaussian with mean 0 and variance σ^2). Assuming the independence of θ_n, the likelihood of the complete data set can be written as:

$$p(\theta \mid \mathbf{w},\sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{1}{2\sigma^2}\left\|\theta - \Phi\mathbf{w}\right\|^2\right\}, \tag{5}$$

where, $\mathbf{\Phi}$ is the N x (N+1) design matrix with $\mathbf{\Phi} = [\Phi(\mathbf{t}_1), \Phi(\mathbf{t}_2), \ldots, \Phi(\mathbf{t}_n)]^{\mathrm{T}}$, wherein $\Phi(\mathbf{t}_n) = [1, K(\mathbf{t}_n, \mathbf{t}_1), K(\mathbf{t}_n, \mathbf{t}_2), \ldots, K(\mathbf{t}_n, \mathbf{t}_N)]^{\mathrm{T}}$.

To prevent over-fitting a preference for smoother functions is encoded by choosing a zero-mean Gaussian prior distribution $\wp$ over $\mathbf{w}$:

$$p(\mathbf{w} \mid \eta) = \prod_{i=1}^{N} \wp(w_i \mid 0, \eta_i^{-1}), \tag{6}$$

with $\boldsymbol{\eta}$ a vector of $N + 1$ hyperparameters. To complete the specification of this hierarchical prior, we must define hyperpriors over $\boldsymbol{\eta}$, as well as over the noise variance σ^2. Having defined the prior, Bayesian inference proceeds by computing the posterior over all unknowns given the data from Bayes' rule:

$$p(\mathbf{w},\eta,\sigma^2 \mid \theta) = \frac{p(\theta \mid \mathbf{w},\eta,\sigma^2)\, p(\mathbf{w},\eta,\sigma^2)}{p(\theta)}, \tag{7}$$

Since this form is difficult to handle analytically, the hyperpriors over $\boldsymbol{\eta}$ and σ^2 are approximated as delta functions at their most probable values $\boldsymbol{\eta}_{MP}$ and σ^2_{MP}. Predictions for new data are then made according to:

$$p(\theta_* \mid \theta) = \int p(\theta_* \mid \mathbf{w},\sigma^2_{MP})\, p(\mathbf{w} \mid \theta,\eta_{MP},\sigma^2_{MP})\, d\mathbf{w}. \tag{8}$$

2.3 Extended Kalman Filter

For nonlinear systems or non-Gaussian noise, there is no general analytic (closed form) solution for the state space PDF. The extended Kalman filter (EKF) is the most popular solution to the recursive nonlinear state estimation problem (Jazwinski, 1970). In this approach the estimation problem is linearized about the predicted state so that the Kalman filter can be applied. In this case, the desired state PDF is approximated by a Gaussian, which may have significant deviation from the true distribution causing the filter to diverge.

The nonlinear state transition and observation models need to be differentiable functions for the EKF. They can be represented as:

$$\mathbf{x}_k = f(\mathbf{x}_{k\text{-}1}, \mathbf{u}_{\underline{k}}, \boldsymbol{\omega}_k)$$
$$\mathbf{y}_k = h(\mathbf{x}_k, \boldsymbol{\upsilon}_k) \tag{9}$$

where, $\mathbf{x}$ denotes the state, $\mathbf{y}$ is the output or measurements, and $\boldsymbol{\omega}_k$ and $\boldsymbol{\upsilon}_k$ are samples from a zero mean Gaussian noise distribution. The function f can be used to compute the predicted state from the previous estimate and similarly h can be used to compute the predicted measurement from the predicted state. Matrices of partial derivatives (the Jacobian), $\mathbf{F}$ and $\mathbf{H}$ for f and h respectively, are computed at each time step with current predicted states. These matrices are used essentially to linearize the non-linear functions around the current estimate.

$$\mathbf{F}_k = (\partial f/\partial \mathbf{x})|\tilde{\mathbf{x}}_{k\text{-}1|k\text{-}1},\mathbf{u}_k$$
$$\mathbf{H}_k = (\partial h/\partial \mathbf{x})|\tilde{\mathbf{x}}_{k|k\text{-}1} \tag{10}$$

The EKF algorithm proceeds in two steps – the prediction step, in which the state transition model is used to propagate the state vector $\mathbf{x}$ into the next time step, and the update step where the measurement $\mathbf{y}$ is used to correct the prediction. The equations for the first step can be written as:

$$\tilde{\mathbf{x}}_{k|k\text{-}1} = f(\tilde{\mathbf{x}}_{k\text{-}1|k\text{-}1}, \mathbf{u}_{\underline{k}}, 0)$$
$$\mathbf{P}_{k|k\text{-}1} = \mathbf{F}_k\mathbf{P}_{k\text{-}1|k\text{-}1}\mathbf{F}_k^{\mathrm{T}} + \boldsymbol{\Omega}_k \tag{11}$$

where, $\mathbf{P}$ and $\boldsymbol{\Omega}$ are the covariance matrices of the predicted state estimate and the process noise $\boldsymbol{\omega}$. The update step can be expressed as:

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \mathbf{H}_k\tilde{\mathbf{x}}_{k|k\text{-}1}$$
$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k\text{-}1}\mathbf{H}_k^{\mathrm{T}} + \boldsymbol{\Psi}_k$$
$$\mathbf{K}_k = \mathbf{P}_{k|k\text{-}1}\mathbf{H}_k^{\mathrm{T}}\mathrm{S}_k^{\text{-}1}$$
$$\tilde{\mathbf{x}}_{k|k} = \tilde{\mathbf{x}}_{k\text{-}1|k\text{-}1} + \mathbf{K}_k\tilde{\mathbf{y}}_k$$
$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k\text{-}1} \tag{12}$$

where, $\boldsymbol{\Psi}$ is the covariance matrix of the observation noise $\boldsymbol{\upsilon}_k$ and $\mathbf{I}$ is the identity matrix.

In the EKF context a k-step future prediction is achieved by simply iterating through the prediction equations the requisite number of times. However, it is important to note that the EKF is not an optimal estimator. If the initial estimate of the state is significantly off-target, or if the process is modeled incorrectly, the filter may quickly diverge, owing to its linearization.

2.4 Particle Filter

In the Particle Filter (PF) approach (Arulampalam, 2002; Gordon, Salmond, and Smith, 1993) the state PDF is approximated by a set of particles (points) representing sampled values from the unknown state space, and a set of associated weights denoting discrete probability masses. The particles are generated and recursively updated from a nonlinear process model that describes the evolution in time of the system under analysis, a measurement model, a set of available measurements and an a priori estimate of the state PDF. In other words, PF is a technique for implementing a recursive Bayesian filter using Monte Carlo (MC) simulations, and as such is known as a sequential MC (SMC) method.

Particle methods assume that the state equations can be modeled as a first order Markov process with the outputs being conditionally independent. This can be written as:

$$\mathbf{x}_k = f(\mathbf{x}_{k\text{-}1}) + \boldsymbol{\omega}_k$$
$$\mathbf{y}_k = h(\mathbf{x}_k) + \boldsymbol{\upsilon}_k \tag{13}$$

where, $\mathbf{x}$ denotes the state, $\mathbf{y}$ is the output or measurements, and $\boldsymbol{\omega}_k$ and $\boldsymbol{\upsilon}_k$ are samples from a noise distribution. Sampling importance resampling (SIR) is a very commonly used particle filtering algorithm, which approximates the filtering distribution denoted as $p(\mathbf{x}_k|\mathbf{y}_0,\ldots,\mathbf{y}_k)$ by a set of P weighted particles $\{(w_k^{(i)},\mathbf{x}_k^{(i)}): i=1,\ldots,P\}$. The importance weights $w_k^{(i)}$ are approximations to the relative posterior probabilities of the particles such that

$$\int f(\mathbf{x}_k)p(\mathbf{x}_k \mid \mathbf{y}_0,\ldots,\mathbf{y}_k)dx_k \approx \sum_{i=1}^{P} w_k^{(i)} f(\mathbf{x}_k^{(i)})$$

$$\sum_{i=1}^{P} w_k^{(i)} = 1. \tag{14}$$

The weight update is given by:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{p(\mathbf{y}_k \mid \mathbf{x}_k)p(\mathbf{x}_k \mid \mathbf{x}_{k-1})}{\pi(\mathbf{x}_k \mid \mathbf{x}_{0:k-1},\mathbf{y}_{1:k})}, \tag{15}$$

where, the importance distribution $\pi(\mathbf{x}_k|\mathbf{x}_{0:k-1},\mathbf{y}_{1:k})$ is approximated as $p(\mathbf{x}_k|\mathbf{x}_{k-1})$.

3. Application Domain

The data used in this study had been collected from second generation 18650-size lithium-ion cells (i.e., Gen 2 cells) that were cycle-life tested at the Idaho National Laboratory under the Advanced Technology Development (ATD) Program (Christophersen et al., 2006), initiated in 1998 by the U.S. Department of Energy to find solutions to the barriers that limit the commercialization of high-power lithium-ion batteries. The cells were aged at 60% state-of-charge (SOC) and various temperatures (25°C and 45°C). We use the 25°C data as the baseline for training and the 45°C data as the faulty sequence. This is a key in assessing the robustness of the different prognostic algorithms in the presence of un-modeled effects.

3.1 Model Development

Apart from the purely data-driven technique ARIMA, EKF and PF both require a system model. RVM is used to derive this model from the baseline data. The first step in model development is to extract features from sensor data comprising of voltage, current, power, impedance electro-chemical impedance spectrometry (EIS), frequency and temperature readings. These features are used to estimate the internal parameters of the battery model shown in Figure 1. The parameters of interest are the double layer capacitance C_{DL}, the charge transfer resistance R_{CT}, the Warburg impedance R_W and the electrolyte resistance R_E. In the dataset under study, R_W and C_{DL} showed negligible change over the ageing process and are excluded from further analysis.

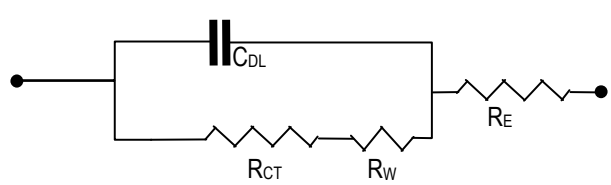

Figure 1 Lumped Parameter Model of a Cell

The values of these internal parameters change with various ageing and fault processes like plate sulfation, passivation and corrosion. RVM regression is performed on parametric data collected from a group of cells over a long period of time so as to find representative ageing curves. Since we want to learn the dependency of the parameters with time, the RVM input vector $\mathbf{t}$ is time, while the target vector θ is given by the inferred parametric values. Exponential growth models, as shown in equation 16, are then fitted on these curves to identify the relevant decay parameters like C and λ:

$$\tilde{\theta} = C\exp(\lambda t), \tag{16}$$

where, $\tilde{\theta}$ is the model predicted value of an internal battery parameter like R_{CT} or R_E. The overall model development scheme is depicted in the flowchart of Figure 2.

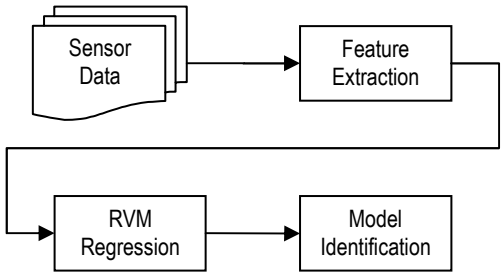

Figure 2 Schematic of Model Development

3.2 RUL Estimation

The system description model developed using RVM is fed into both the EKF and PF algorithms. Data from the system sensors are mapped into system features which are subsequently used to estimate the RUL as explained below. The EKF and the PF use the parameterized exponential growth model, described in equation 16, for the propagation of the state estimates (equation 11) and particles (equation 13) in time, respectively. The EKF maintains C and λ as constant model parameters while the PF algorithm incorporates them as well as the internal battery parameters R_E and R_{CT} as components of the state vector $\mathbf{x}$, and thus, performs parameter identification in parallel with state estimation. The values of C and λ learnt from the RVM regression are used as initial estimates for the particle filter. The measurement vector $\mathbf{y}$ is comprised of the battery parameters inferred from measured data. In the PF algorithm, resampling of the particles is carried out in each iteration so as to reduce the occurrence of degeneracy of particle weights. Taking advantage of the highly linear correlation between R_E+R_{CT} and C/1 capacity (as derived from data), predicted values of the internal battery model parameters are used to calculate expected charge capacities of the battery. The predictions are compared against end-of-life thresholds to derive the RUL estimates. Figure 3 shows a simplified schematic of the process described above. ARIMA simply uses the C/1 time series data to predict future points.

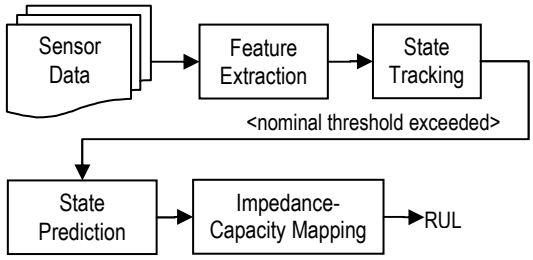

Figure 3 Schematic of RUL Prediction

4. Results

The results for the model development section are presented in the form of four plots. Figure 4 shows the shift in electro-chemical impedance spectrometry (EIS) data of one of the test cells with ageing at 25°C. The nearly vertical left tails of the EIS plots are due to inductances in the battery terminals and connection leads. In some models this distributed inductance is represented in the form of a lumped inductance parameter in series with the electrolyte resistance R_E. The tails on the right side of the curves arise from diffusion based cell transport phenomena. This is modeled as the parameter R_W in Figure 1. Figure 5 shows a zoomed in section of the data presented above in Figure 4 with the battery internal model parameters identified. Since the expected frequency plot of a resistance and a capacitance in parallel is a semicircle, we fit semicircular curves to the central sections of the data in a least-square sense. The left intercept of the semicircles give the R_E values while the diameters of the semicircles give the R_{CT} values. Figure 6 shows the output of the RVM regression along with the exponential growth model fits for R_E and R_{CT}. The use of probabilistic kernels in RVM helps to reject the effects of outliers and the varying number of data points at different time steps, which can bias conventional least-square based model fitting methods. Figure 7 shows the high degree of linear correlation between the C/1 capacity and the internal impedance parameter R_E+R_{CT}. We exploit this relationship to estimate the current and future C/1 capacities.

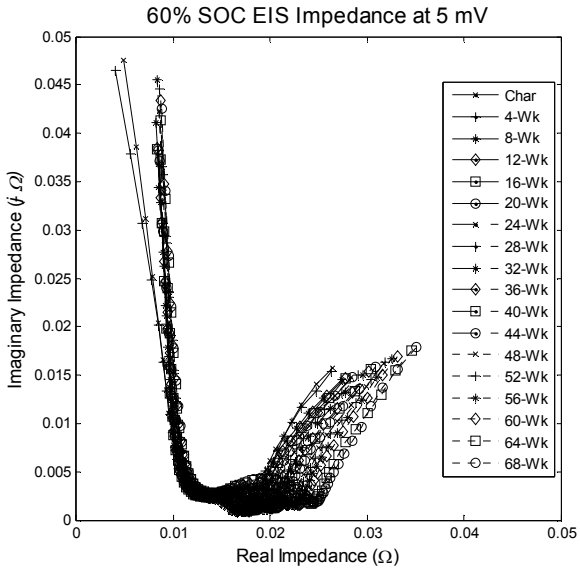

Figure 4 Shift in EIS Data with Ageing

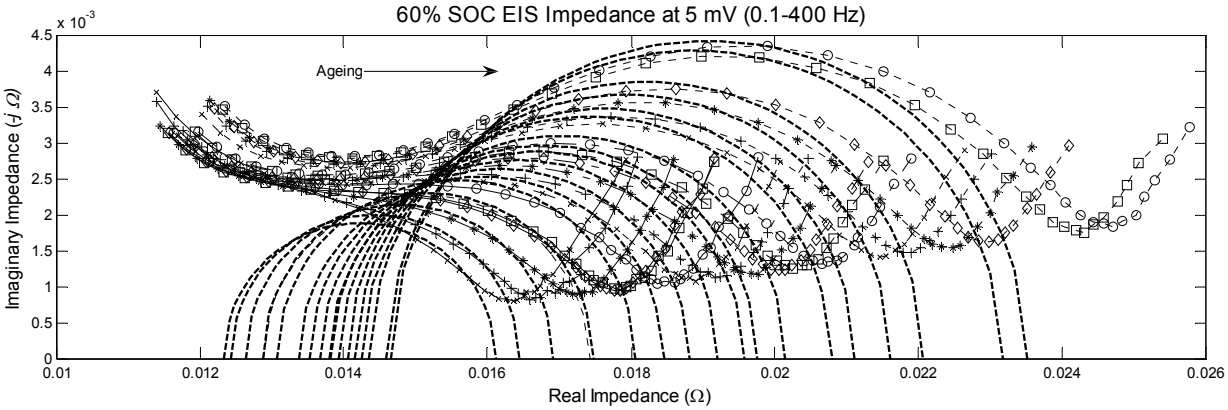

Figure 5 Zoomed EIS Plot with Internal Battery Model Parameter Identification

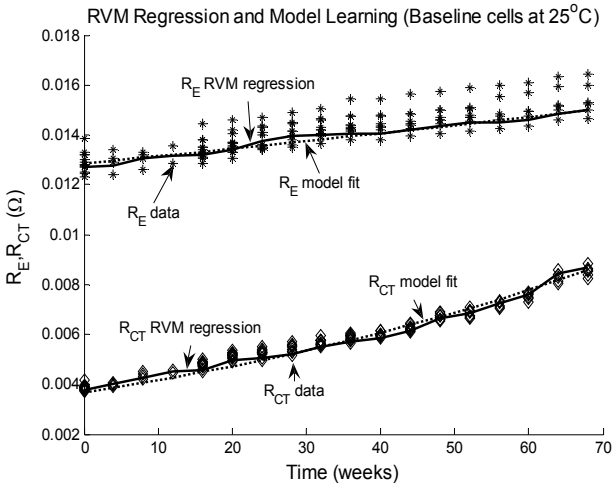

Figure 6 RVM Regression and Growth Model Fit

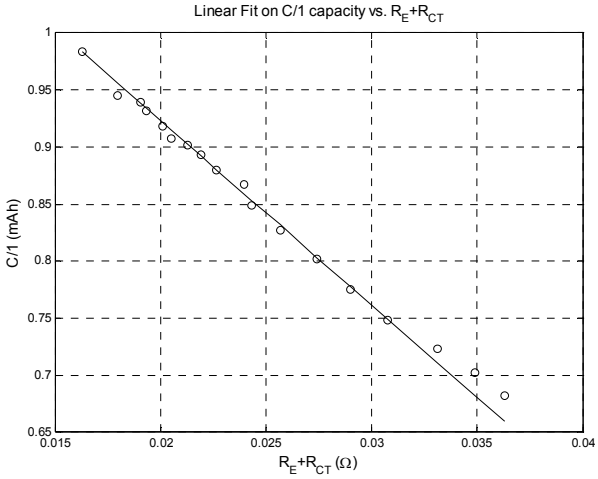

Figure 7 Correlation between Capacity and Impedance Parameters

The implementation of the ARIMA(p,d,q) process begins with determining the values for p, d, and q. The data are roughly exponential in nature; d is chosen to be 2 in order to remove the non-stationarity approximately. Figure 8 shows the autocorrelation and partial autocorrelation plots from which both the p and q values are chosen to be 1. Higher orders do not improve the model fit any further. Figure 9 shows the ARIMA predictions at 36 weeks (squares) and at 52 weeks (circles). Although the RUL estimates are not too far off the real value, the confidence bounds are too wide to have significant practical value. The model-based EKF performs well in tracking the battery capacity despite the application of model parameters derived at 25°C to 45°C data as shown in Figure 10. However, it deviates badly when used for prediction (no measurement update step) while using the baseline model parameters.

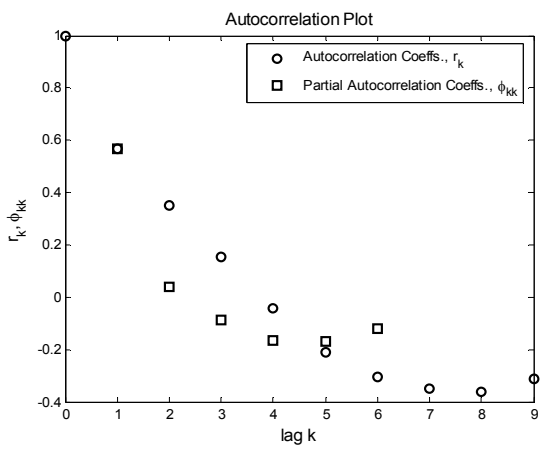

Figure 8 Autocorrelation and Partial Autocorrelation Plots for ARIMA

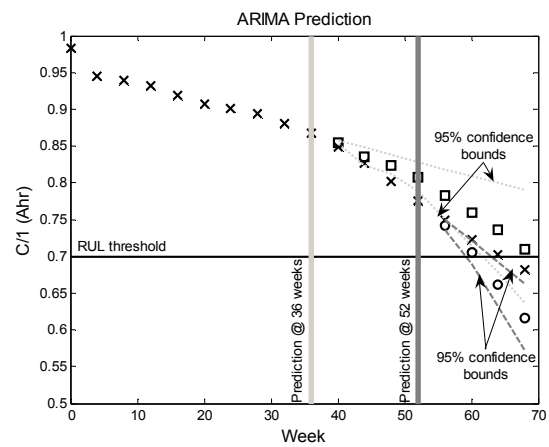

Figure 9 ARIMA Prediction

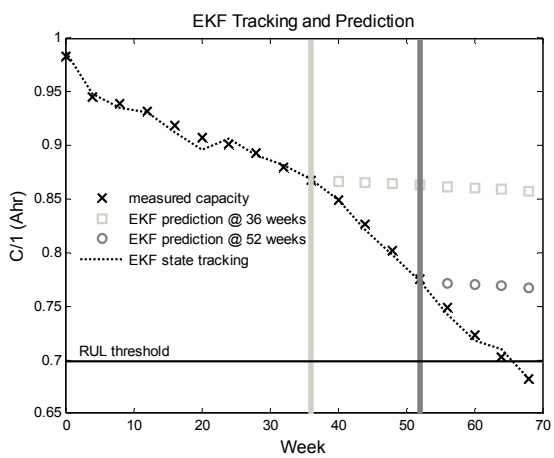

Figure 10 EKF State Tracking and Prediction

Figure 11 shows both the state tracking and future state prediction plots for the Particle Filter applied to the 45°C data. Since the PF also identifies the current parameters, it shows that the estimated λ value for the R_{CT} growth model (equation 16) is considerably larger than of the training data (collected at 25°C), and hence can be used for diagnosis. The threshold for fault declaration in the plot has been arbitrarily chosen. The diagnosis in this case is that the cell has undergone rapid passivation due to the elevated temperatures. Remaining-useful-life (RUL) or time-to-failure (TTF) is derived by projecting out the capacity estimates (derived from the mapping shown in Figure 7) into the future (Figure 12) until expected capacity hits the certain predetermined end-of-life threshold. The particle distribution is used to calculate the RUL PDF by fitting a mixture of Gaussians in a least-squares sense. As shown in Figure 12, the RUL PDF improves in both accuracy (centering of the PDF over the actual failure point) and precision (spread of the PDF over time) with the inclusion of more measurements before prediction.

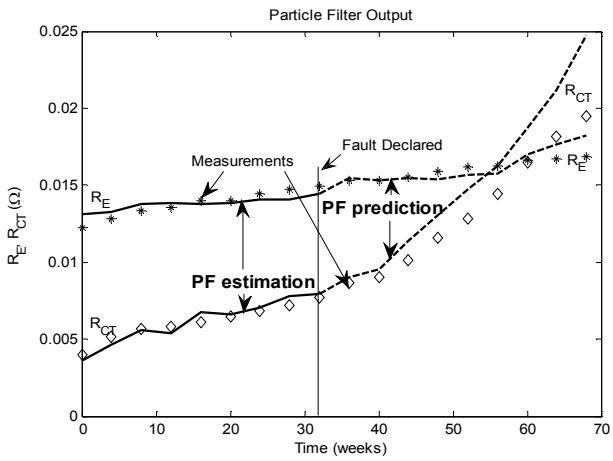

Figure 11 Particle Filter Output

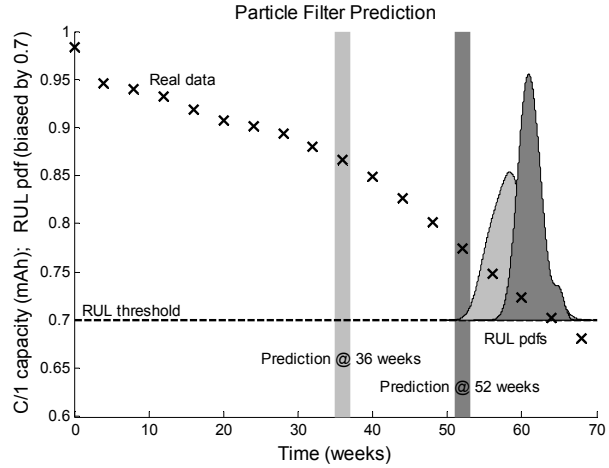

Figure 12 Particle Filter Prediction

5. Conclusions

We were interested here in particular in conditions where un-modeled effects are present as manifested by the different degradation curve at 45°C. Although all algorithms were given the same amount of information to the degree practical, there were considerable differences in performance. Specifically, the combined Bayesian regression-estimation approach implemented as a RVM-PF framework has significant advantages over conventional methods of RUL estimation like ARIMA and EKF. ARIMA, being a purely data-driven method, does not incorporate any physics of the process into the computation, and hence ends up with wide uncertainty margins that make it unsuitable for long-term predictions. Additionally, it may not be possible to eliminate all non-stationarity from a dataset even after repeated differencing, thus adding to prediction inaccuracy. EKF, though robust against non-stationarity, suffers from the inability to accommodate un-modeled effects and can diverge quickly as shown. We did not explore other variations of the Kalman Filter that might provide better performance such as the unscented Kalman Filter. The Bayesian statistical approach, on the other hand, appears to be well suited to handle various sources of uncertainties since it defines probability distributions over both parameters and variables and integrates out the nuisance terms. Also, it does not simply provide a mean estimate of the time-to-failure; rather it generates a probability distribution over time that best encapsulates the uncertainties inherent in the system model and measurements and in the core concept of failure prediction.

References

Arulampalam, S.; Maskell, S.; Gordon, N. J.; and Clapp, T. 2002: A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking. *IEEE Trans. on Signal Processing* 50(2): 174-188.

Bhangu, B. S.; Bentley, P.; Stone, D. A.; and Bingham, C. M. 2005: Nonlinear Observers for Predicting State-of-Charge and State-of-Health of Lead-Acid Batteries for Hybrid-Electric Vehicles. *IEEE Transactions on Vehicular Technology* 54(3):783-794.

Blanke, H.; Bohlen, O.; Buller S.; De Doncker, R. W.; Fricke, B; Hammouche, A; Linzen, D; Thele, M; and Sauer, D. U. 2005: 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* 144(2):418-425.

Boser, B. E.; Guyon, I. M.; and Vapnik, V. N. 1992: A Training Algorithm for Optimal Margin Classifiers. Haussler, D., editor, *5th Annual ACM Workshop on COLT*. Pittsburgh, Penn.: ACM Press, 144-152.

Box, G. E. P., and Jenkins, G. M. 1976: *Time Series Analysis: Forecasting and Control*. San Francisco, Cal.: Holden Day.

Christophersen, J., Bloom, I., Thomas, E., Gering, K., Henriksen, G., Battaglia, V., and Howell, D. 2006: *Advanced Technology Development Program for Lithium-Ion Batteries: Gen 2 Performance Evaluation Final Report*. INL Technical Report INL/EXT-05-00913.

Cox, D. C. and Perez-Kite, R. 2000: Battery State of Health Monitoring, Combining Conductance Technology with other Measurement Parameters for Real-time Battery Performance Analysis. *22nd*

International Telecommunications Energy Conference, INTELEC 342-347.

Drucker, H.; Burges, C. J. C.; Kaufman, L.; Smola, A. J.; and Vapnik, V. 1997: Support Vector Regression Machines. Mozer, M.; Jordan, M.; and Petsche, T., editors, *Advances in Neural Information Processing Systems*. Cambridge, Mass.: MIT Press, 9:155-161.

Gao, L.; Liu, S; and Dougal, R. A. 2002: Dynamic Lithium-Ion Battery Model for System Simulation. *IEEE Transactions on Components and Packaging Technologies* 25(3):495-505.

Gordon, N. J.; Salmond, D. J.; and Smith, A. F. M. 1993: Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation. *Radar and Signal Processing, IEE Proceedings F* 140(2):107-113.

Jaworski, R.K. 1999: Statistical Parameters Model for Predicting Time to Failure of Telecommunications Batteries. *21st International Telecommunications Energy Conference, INTELEC*.

Jazwinski, A. H. 1970: *Stochastic Processes and Filtering Theory*. New York: Academic Press.

Kozlowski, J. D. 2003: Electrochemical Cell Prognostics Using Online Impedance Measurements and Model-based Data Fusion Techniques. *Aerospace Conference 2003, IEEE Proceedings* 7:3257-3270.

Meissner, E., and Richter, G. 2003: Battery Monitoring and Electrical Energy Management - Precondition for Future Vehicle Electric Power Systems. *Journal of Power Sources* 116(1):79-98.

Tipping, M. E. 2000: The Relevance Vector Machine. *Advances in Neural Information Processing Systems*. Cambridge, Mass.: MIT Press, 12:652-658.

Vapnik, V. N. 1995: *The Nature of Statistical Learning*. Berlin: Springer.

Vutetakis, D. G., and Viswanathan, V. V. 1995: Determining the State-of-Health of Maintenance-Free Aircraft Batteries. *Tenth Annual Battery Conference on Applications and Advances, Proceedings* 13-18.