A Bayesian Framework for Remaining Useful Life Estimation

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

[1]Georgia Institute of Technology
777 Atlantic Drive
Atlanta, GA 30332
bsaha@ece.gatech.edu

[2]NASA Ames Research Center
Moffett Field, CA 94035
{goebel,spoll}@email.arc.nasa.gov

[3]Idaho National Laboratory
P.O. Box 1625
Idaho Falls, ID 83415
Jon.Christophersen@inl.gov

Abstract

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. This is especially true for aerospace systems, where unanticipated subsystem downtime may lead to catastrophic failures. RUL prediction needs to contend with multiple sources of error like modeling inconsistencies, system noise and degraded sensor fidelity. Bayesian theory of uncertainty management provides a way to contain these problems by integrating out the nuisance variables. We use the Relevance Vector Machine (RVM), for model development. RVM is a Bayesian treatment of the well known Support Vector Machine (SVM), a kernel-based regression/classification technique. This model is next used in a Particle Filter (PF) framework. Statistical estimates of the noise in the system and anticipated operational conditions are processed to provide estimates of RUL in the form of a probability density function (PDF). Validation of this approach on experimental data collected from Li-ion batteries is presented.

Introduction

The application of artificial intelligence (AI) has successfully been adopted in many engineering systems. This is in part true because researchers have focused not so much recreating human knowledge (and it would be unreasonable to cast hundreds of years of engineering knowledge in abstract logic) but has instead focused on how AI and can support various engineering task. As an indirect result, the expanding research in AI has over the years given rise to a variety of sub-fields like *knowledge engineering*, *expert systems*, *soft computing*, and *automated reasoning*. The field of prognostics is a prime example where AI can make a tremendous impact in support of engineering. At the core of prognostics are complex engineering systems that provide crucial information for the remaining life estimation task. Algorithms – influenced by AI – are then tasked with predicting when a component will fail, sort through possibly many different future use branches, deal with uncertainty from sensors, model, future use, etc., be conscious of computational resources, and do this in a way that can be validated and verified.

Copyright © 2007, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Very early development of AI concentrated on logical systems, which interacted with the world through "if and then" statements. The importance of probabilities rose with the realization that logical systems could not anticipate all possible contingencies. Consequently Bayesian techniques (amongst others) started to be assimilated in the AI domain. Simply put, Bayes' theory defines the concept of probability as the degree of belief that a proposition is true. Furthermore, it also suggests that Bayes' theorem can be used as a rule to infer or update the degree of belief in light of new information or data – the more the data, the better the predictions. An additional advantage is that Bayesian models are self-correcting, meaning that the predictions change with change in data trends.

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.

Methodology

Relevance Vector Machine

In a given classification problem, the data points may be multidimensional (say n). The task is to separate them by a n-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 $\{\mathbf{t}_n\}_{n=1}^N$ and their corresponding targets $\{\theta_n\}_{n=1}^N$. 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:

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

where, $\mathbf{w} = (w_1, w_2, \ldots, w_M)^T$ 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:

$$\theta_n = F(\mathbf{t}_n;\mathbf{w}) + \varepsilon_n, \tag{2}$$

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:

$$p(\theta \mid \mathbf{w},\sigma^2) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{1}{2\sigma^2}\|\theta - \Phi\mathbf{w}\|^2\right\}, \tag{3}$$

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

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

$$p(\mathbf{w} \mid \eta) = \prod_{i=1}^{N} \wp(w_i \mid 0, \eta_i^{-1}), \tag{4}$$

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:

$$p(\mathbf{w},\eta,\sigma^2 \mid \theta) = \frac{p(\theta \mid \mathbf{w},\eta,\sigma^2)\, p(\mathbf{w},\eta,\sigma^2)}{p(\theta)}, \tag{5}$$

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:

$$p(\theta_* \mid \theta) = \int p(\theta_* \mid \mathbf{w},\sigma^2_{MP})\, p(\mathbf{w} \mid \theta,\eta_{MP},\sigma^2_{MP})\, d\mathbf{w}. \tag{6}$$

Particle Filters

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 PDF is approximated by a Gaussian, which may have significant deviation from the true distribution causing the filter to diverge.

In contrast, for the *Particle Filter* (PF) approach (Arulampalam 2002; Gordon, Salmond, and Smith 1993) the 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:

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}) + \vartheta_k \\ \mathbf{y}_k &= h(\mathbf{x}_k) + \omega_k \end{aligned} \tag{7}$$

where, $\mathbf{x}$ denotes the state, $\mathbf{y}$ is the output or measurements, and ϑ_k and ω_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

$$\begin{aligned} &\int f(\mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_0,\ldots,\mathbf{y}_k)\, dx_k \approx \sum_{i=1}^{P} w_k^{(i)} f(\mathbf{x}_k^{(i)}) \\ &\sum_{i=1}^{P} w_k^{(i)} = 1. \end{aligned} \tag{8}$$

The weight update is given by:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{x}_{k-1})}{\pi(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k})}, \qquad (9)$$

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

Application

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 can 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. An efficient method for battery monitoring would greatly improve the reliability of such systems.

Model Development

In order to tie in the above discussed techniques, namely RVM and PF, with the battery health monitoring problem, the process is broken down into an offline and an online part. During offline analysis, the battery/cell operation is expressed in the form of structural and functional models, which aid in the construction of the "physics of failure mechanisms" model. Features extracted from sensor data comprising of voltage, current, power, impedance electro-chemical impedance spectrometry (EIS), frequency and temperature readings, are used to estimate the internal parameters of the battery model shown in Figure 1. The parameters of interest are the double layer capacitance CDL, the charge transfer resistance RCT, the Warburg impedance RW and the electrolyte resistance RE.

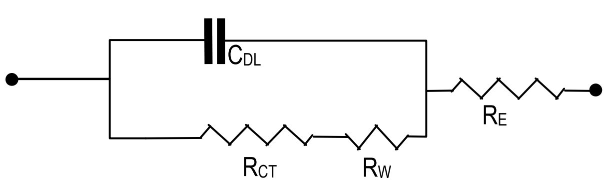

Figure 1. Lumped Parameter Model of a Cell

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 **t** is time, while the target vector θ is given by the inferred parametric values. Exponential growth models, as shown in equation 10, are then fitted on these curves to identify the relevant decay parameters like C and λ:

$$\tilde{\theta} = C \exp(\lambda t), \qquad (10)$$

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.

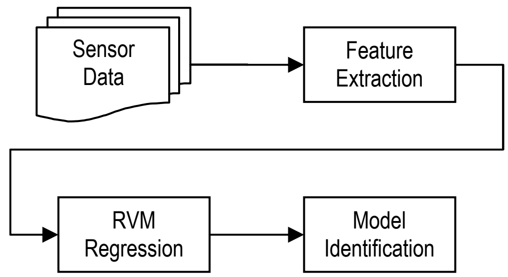

Figure 2. Schematic of Model Development

RUL Estimation

The system description model developed in the offline process is fed into the online process. Data from the system sensors are mapped into system features which are subsequently used to estimate the RUL as explained below. The PF uses the parameterized exponential growth model, described in equation 10, for the propagation of the particles in time. The algorithm incorporates the model parameters C and λ as well as the internal battery parameters R_E and R_{CT} as components of the state vector **x**, and thus, performs parameter identification in parallel with state estimation. The measurement vector **y** is comprised of the battery parameters inferred from measured data. The values of C and λ learnt from the RVM regression are used as initial estimates for the particle filter. 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 $R_{CT}+R_E$ 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.

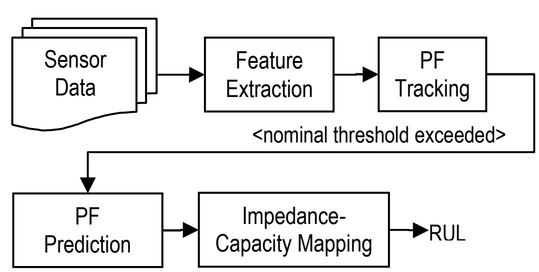

Figure 3. Particle Filter Framework

Results

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, 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).

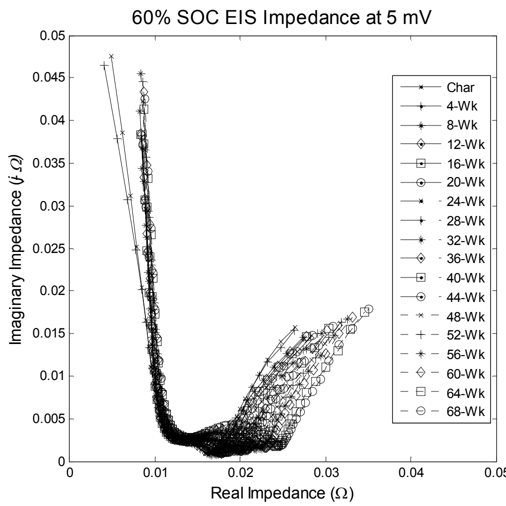

Figure 4. Shift in EIS Data with Ageing

The results for the model development section are presented in the form of 3 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 L 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 6 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. Other internal parameters like R_W and C_{DL} are not plotted since they showed negligible change over the ageing process and are excluded from further analysis.

Figure 5 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.

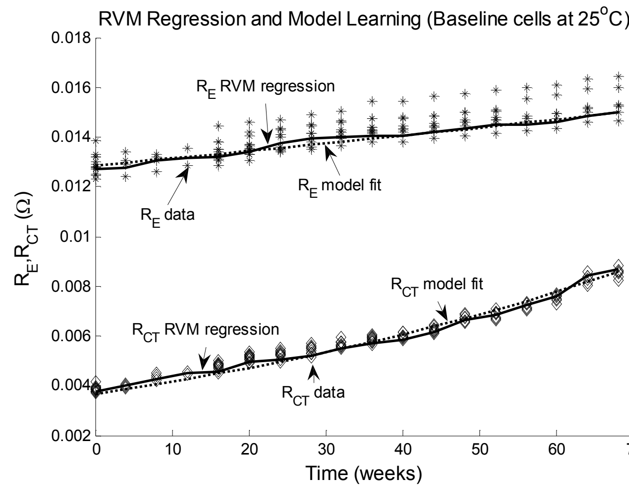

Figure 5. RVM Regression and Growth Model Fit

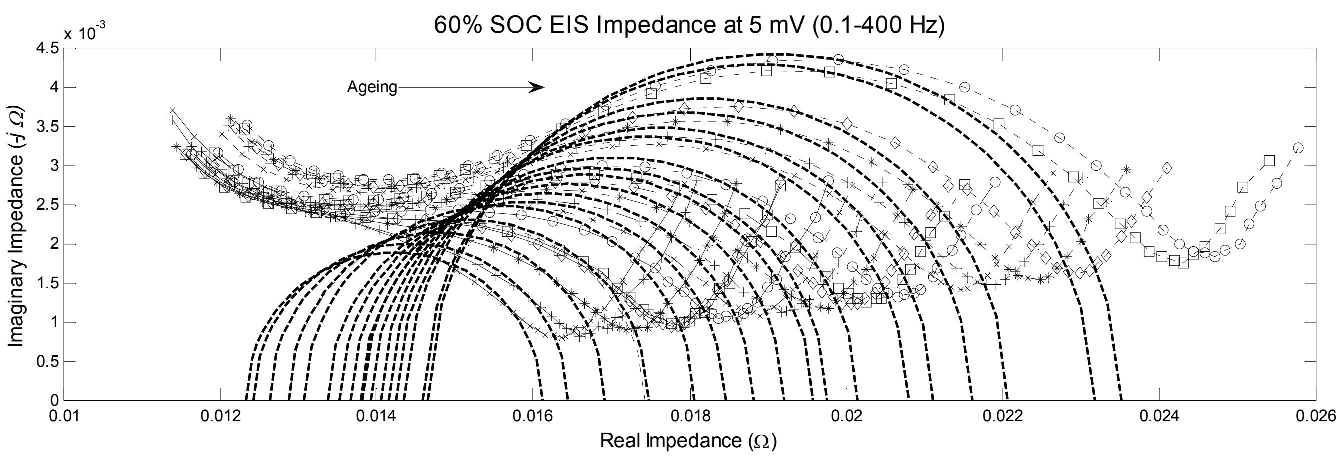

Figure 6. Zoomed EIS Plot with Internal Battery Model Parameter Identification

Figure 7 shows both the state tracking and future state prediction plots for data collected at 45°C. The threshold for fault declaration has been arbitrarily chosen. The estimated λ value for the R_{CT} growth model (equation 10) is considerably larger than of the training data (collected at 25°C). Consequently, the diagnosis is that the cell has undergone rapid passivation due to the elevated temperatures.

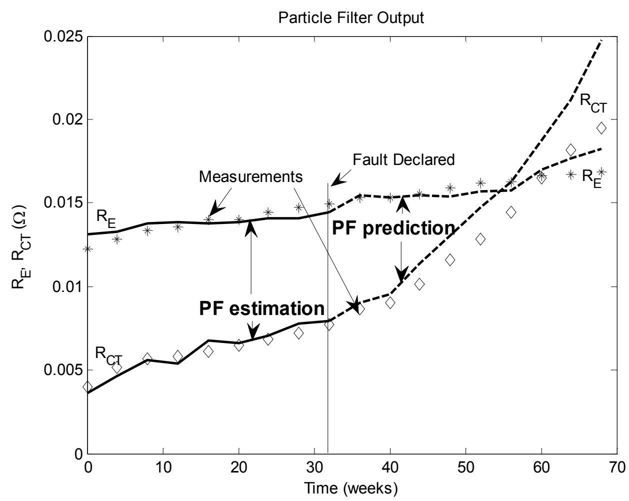

Figure 7. Particle Filter Output

Figure 8 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.

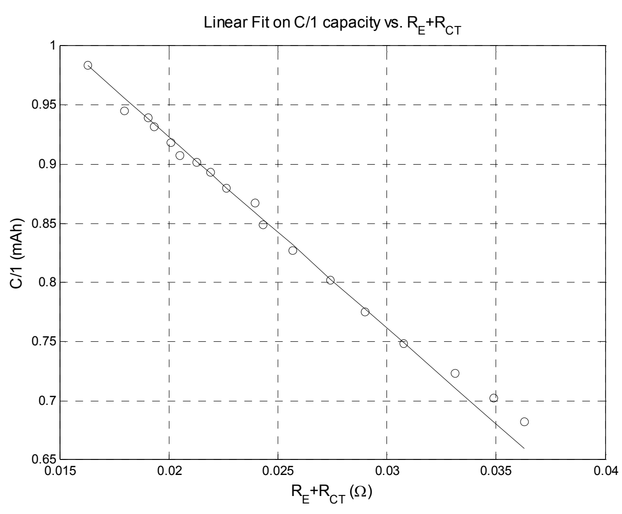

Figure 8. Correlation between Capacity and Impedance Parameters

Remaining-useful-life (RUL) or time-to-failure (TTF) is used as the relevant metric for SOL. This is derived by projecting out the capacity estimates into the future (Figure 9) until expected capacity hits a certain predetermined end-of-life threshold. The particle distribution is used to calculate the RUL probability density (PDF) by fitting a mixture of Gaussians in a least-squares sense. As shown in Figure 9, 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.

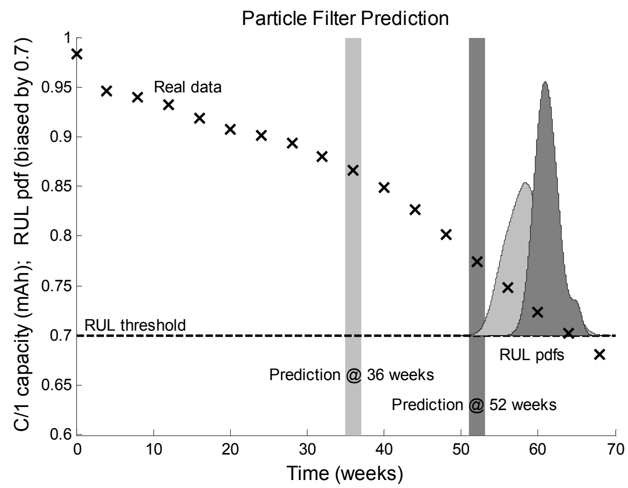

Figure 9. Particle Filter Prediction

Conclusions

The combined Bayesian regression-estimation approach implemented as a RVM-PF framework has significant advantages over conventional methods RUL estimation. A Bayesian statistical approach is very well suited to complex systems whose internal state variables are either inaccessible to sensors or hard to measure under operational conditions, and whose performance is strongly influenced by ambient environmental and load conditions. Additionally, the discussed methodology 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 basic 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.

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

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. eds. *Advances in Neural Information Processing Systems*, 9:155-161. Cambridge, Mass.: MIT Press.

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.

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

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

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