Health Monitoring Algorithms for Space Application Batteries

Freeman Rufus Jr., *Member, IEEE*, Seungkoo Lee, *Member, IEEE*, Ash Thakker

***Abstract*— Prototype battery health monitoring algorithms (support vector machine, dynamic neural network, confidence prediction neural network, and usage pattern analysis) were developed and tested on the battery data (voltage, current, temperature, etc.) collected from several 4-amp hour lithium ion (Li-ion) battery cells supplied by United Lithium Systems. The battery data was collected under different operating conditions (storage and charge/discharge cycling under room and 50°C temperatures. The results show that the battery health monitoring algorithms is feasible for determining the health state of a Li-ion cell yielding remaining useful life information to the user.**

***Index Terms*—Battery health algorithms; remaining useful life; Li-ion batteries; virtual sensors.**

I. Introduction

ELECTRICAL power is a vital resource on-board many of NASA's and DoD's space systems. Electric power can be generated through the use of solar cells and/or batteries (a type of solar cell called photovoltaic can convert light energy directly to electricity). However, light energy is not always available to spacecraft orbiting the Earth and thus, the stored energy in batteries must be relied upon. Batteries supply electric power to life sustaining systems, communications equipment, experiments, thermal management, etc. aboard many space systems such as the International Space Station (ISS), Crew Exploration Vehicles (CEV), Crew Launch Vehicle (CLV), Mars Land Rovers (MLR), etc. Therefore, batteries play a critical role in most space missions and must provide reliable and predictable electric power.

Battery state-of-health (SOH) is an important quantity to monitor for space applications. SOH is typically characterized by one or more of the following: available capacity, internal resistance/impedance, capacity fade, self-discharge rate. There are several issues related to the health monitoring of batteries, although many different techniques (voltage recovery, impedance measurements, etc.) have been used to estimate their SOH. Voltage recovery is the common technique employed for estimating the SOH of batteries used in back-up environments such as telecommunications, UPS and other storage applications [1][2]. In this approach the voltage depression under load and the temporal recovery of the battery voltage after removal of the load are monitored and used to estimate the internal resistance of the battery. This method is applicable to battery applications having well understood operating modes, each having relatively stable and predictable loads [2]. There is a lack in performance of off-the-shelf impedance-based technology measurement techniques and processing of the information, especially for online use. Usually the battery condition is estimated from the value of the impedance at a single frequency [1]. Recently, Midtronics developed the discrete frequency immitance spectroscopy (DFIS) approach [3] to derive the complete spectrum of the ac impedance using only the battery's ac impedance measured at three different frequencies. Full spectrum impedance measurement are done off-line since they require at least 17 minutes to measure the frequency range of 1 mHz to 1 kHz [4]. Regardless of the approach used to measure the complete spectrum, the computationally intensive approach of complex least squares analysis is typically used to extract equivalent circuit parameters of the battery that are used to evaluate the SOH of the battery. The approach of using Kalman filtering with electrochemical [5] or electrical equivalent-circuit [6] models to estimate SOH of a battery is time-consuming and usually done off-line [7].

A data-driven, diagnostic and prognostic architecture for detection, identification, and prediction of battery failure modes based upon support vector machine and neural network approach is proposed in this work to detect and identify critical incipient battery failure modes and predict the remaining battery life under different power profiles once a failure has been detected. The architecture will utilize a usage pattern analysis module and fuse prognostic information from both failure and usage-pattern based algorithms. This will enhance the reliability of batteries by providing the crew information about the current health state and predicted remaining life of batteries so that timely action can be taken.

II. Battery Health Monitoring Algorithms

A. Architecture

The Battery Diagnostics/Prognostic Architecture for Space

Manuscript received May 23, 2008. This work was supported in part by NASA under Contract # NNX07CA02P.

F. Rufus, Jr. is with Global Technology Connection, Inc., Atlanta, GA 30339 USA (corresponding author; phone: 770-803-3001 x29; fax: 770-234 4148; e-mail: frufus@globaltechinc.com).

S. Lee. and A. Thakker are with Global Technology Connection, Inc., Atlanta, GA 30339 USA (e-mail: slee@globaltechinc.com, athakker@globaltechinc.com).

9778-1-4244-1936-4/08/$25.00 © 2008 IEEE

Applications shown in Figure 1 depicts the basic software modules of the proposed architecture based upon data-driven prognostics [8][9][10]. Pre-processing of the raw sensor data consists of filtering and normalization operations to prevent noise and outliers from creating false positives/negatives failure detections. The pre-processing operation also includes feature extraction operations which map incoming data into a feature space to aid in the diagnostic failure classification and prognostic failure progression predictions. A support vector machine will be used to detect and identify battery failures specific to the space applications. The battery life prognostic module is based on three constructs: 1) a static "virtual sensor" that relates known measurements to battery deterioration (or state-of-health); 2) a dynamic neural network (DNN) predictor which attempts to project the current state of the damaged material into the future thus revealing the time evolution of the damage and allowing the estimation of the battery's calendar life; and 3) a Confidence Prediction Neural Network (CPNN) [11][12] whose task is to account for uncertainty and manage/shrink the prediction bounds. For battery operation other than under fault conditions, a mode estimator keeps track of battery operating modes (high temperatures, discharge rates, duty cycle types, etc.) and usage pattern analysis is performed on the mode sequence to predict battery capacity degradation. Linear opinion pooling is used to fuse the results of the usage pattern and failure based battery remaining useful life prediction.

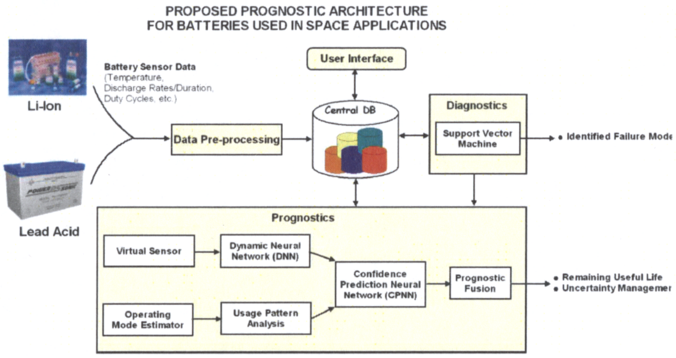


Fig. 1. Overall Diagnostic/Prognostic Architecture for Predicting Remaining Lifetime of Batteries used in Space Applications

B. *Support Vector Machine (SVM) Based Diagnostic Module*

A Support Vector Machine (SVM) [13] is a machine learning algorithm which has roots in the linear classification, however, it has the ability to implicitly create high dimensional feature space mappings (through a kernel) to improve classifier performance. The SVM learning algorithm is driven by a convex objective function which does not have local minima, ensuring efficient training even for large data sets. Thus, a soft margin SVM algorithm [14] will be developed and tested to examine the reliability of the classification of the selected battery related failure modes. The inputs to the SVM will be the heuristically defined features from raw data (which will be further mapped into another feature space via the kernel) and the output will be a sequence of bits (0 or 1) relating to a particular failure mode.

The benefits of using the SVM classifier for battery fault detection are: (1) it finds a unique separating hyperplane which has maximal margin with no local minima during optimization, unlike neural nets; (2) it has implicit feature transformation of input data vectors into feature space via kernel; and (3) efficient optimization is possible using dual form of the optimization problem.

A 1-Norm Soft Margin SVM is a relaxed form of the Maximal Margin SVM which can be used to classify nonlinearly separable data sets. The *primal form* of the 1-Norm Soft Margin SVM optimization problem is:

Given a linearly separable training set $T = \{(\mathbf{x}_1, p_1), (\mathbf{x}_2, p_2), \ldots, (\mathbf{x}_N, p_N)\}$,

$$\text{minimize } \langle \mathbf{w}, \mathbf{w} \rangle + C \sum_{i=1}^{N} \xi_i \,,$$

subject to the constraints

$$p_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b \right) \geq 1 - \xi_i \,,$$

$$\xi_i \geq 0 \,,\; i = 1, 2, \ldots, N$$

where $\mathbf{x}_i \in \Re^M$ are data/feature vectors, $p_i \in \{-1, +1\}$ is the class label, $\mathbf{w} \in \Re^M$ is the weight vector, $\langle \cdot, \cdot \rangle$ is an inner product, $C \in \Re^+$ is a relaxation constant, $\xi_i \in \Re$ are slack variables, and $b \in \Re$ is a bias value. For the purposes of feasibility, the -1 class will represent "*normal life*" while +1 class will mean "*near the end of battery life.*"

The *dual form* of the maximal margin problem is:

Given a linearly separable training set $T = \{(\mathbf{x}_1, p_1), (\mathbf{x}_2, p_2), \ldots, (\mathbf{x}_N, p_N)\}$,

$$\text{maximize } W(\boldsymbol{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} p_i p_j \alpha_i \alpha_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \,,$$

subject to the constraints

$$\sum_{i=1}^{N} p_i \alpha_i = 0 \,,$$

$$0 \leq \alpha_i \leq C \,,\; i = 1, 2, \ldots, N \,.$$

where the α_i terms constitute a dual representation for the weight vector in terms of the training set. The dual form can be solved using gradient descent. Using a kernel, K (e.g. Gaussian), the derivative of W with respect to α_i is:

$$\frac{\partial W}{\partial \alpha_i} = 1 - \frac{1}{2} p_i \sum_{j=1}^{N} p_j \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \qquad (1)$$

This gradient is used to calculate new values of alpha at each training step with a chosen step size, *h*, as:

$$\alpha_i' = \alpha_i + h\frac{\partial W}{\partial \alpha_i} \tag{2}$$

After each update of alpha, the constraint $0 \le \alpha_i \le C$ is enforced (i.e. $\alpha_i' = \min(\max(\alpha_i, 0), C)$). The training algorithm terminates once the evaluated primal and dual objective functions are close enough. Once the α's are known the decision function can be calculated using:

$$y(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i p_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{3}$$

$$p(\mathbf{x}) = \mathrm{sgn}(y(\mathbf{x})) \tag{4}$$

where $K(\mathbf{x}, \mathbf{y})$ is a chosen kernel. A more efficient learning algorithm called Sequential Minimization [15] can also be used.

C. *Virtual Sensor*

The virtual sensor calculates a failure measure indirectly through a neural network mapping of processed sensor data, features, and estimated mode. It is often true that machine or component faults are not directly accessible for monitoring their growth behavioral patterns. Consider, for example, the case of a battery failure due to deep discharge. No direct measurement of the copper solution concentration in the electrolyte is possible when the component is in an operational state. That is, there is no such device as a "fault meter" capable of providing direct measurements of the fault evolution. Examples of a similar nature abound.

D. *Dynamic Neural Network*

Prognosis of the time to failure or the remaining useful lifetime of a component or subsystem involves prediction, i.e. the ability to determine when the component will fail once an incipient failure condition has been detected and identified. It is imperative, therefore, that the prognosticator monitor the time evolution of a failure event, project accurately historical failure data and suggest to the operator or user the most probable (and desirable) time window for maintenance so that equipment uptime is maximized. Signals in such a network configuration can flow not only in the forward direction but also can propagate backwards, in a feedback sense, from the output to the input nodes. A Dynamic Neural Network (DNN) is proposed to address the prediction issues. The basic structure of a DNN is shown in Fig. 2. Delayed versions of the input and output augment now the input feature vector and the resulting construct can be formulated as:

$$Y(t+1) = DNN\big(Y(t), \cdots, Y(t-M), U(t), \cdots, U(t-N)\big) \tag{5}$$

where U is the external input; Y is the output; M is the number of outputs; N is the number of external inputs.

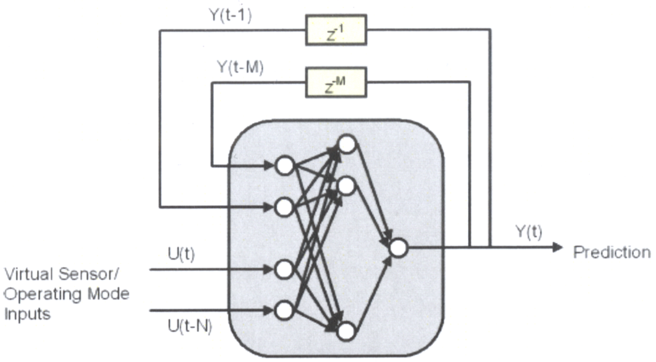


Fig. 2. Structure of dynamic neural network that produces a prediction from delayed feature vectors of input from virtual sensor and operating mode modules.

The *DNN* can be trained in a time-dependent way, using either a gradient-descent technique like the Levenberg-Marquardt algorithm or an evolutionary one such as the genetic algorithm. A virtual sensor fault measure output acts as the input to the *DNN* during training (on- or off-line) and a predicted battery fault measure will be output.

E. *Confidence Prediction Neural Network (CPNN)*

The CPNN module places bounds on the remaining useful lifetime predictions made by the DNN. The following paragraphs describe the CPNN background and function. Based on the basic operation of the General Regression Neural Network (GRNN), the CPNN accomplishes the goal of representing uncertainty in the form of a confidence distribution by employing a confidence distribution approximator node as shown in Fig. 3.

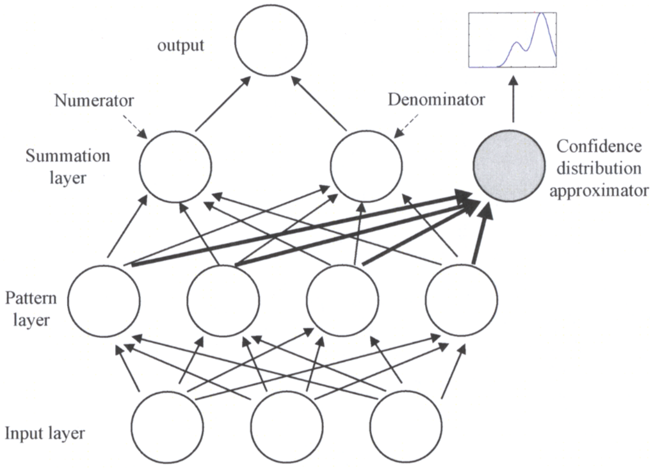


Fig. 3. Structure of the confidence prediction neural network

For 1-step ahead prediction, a buffer of a finite number of recent values, $\boldsymbol{X}$, forms an input vector to the network. The input vector is compared to patterns extracted from historical data. Each comparison will receive a degree of matching $C(\boldsymbol{X}, Y_i)$ and a candidate output y_i. This is exactly the first step of the GRNN, i.e. obtaining an estimate probability density function $\hat{f}(\boldsymbol{X}, Y)$, when it attempts to approximate an output of an input vector. After the whole comparison process is completed, each candidate output will be weighted by its degree of matching to give a final contribution of confidence as a scalar output. However, the purpose of the new methodology is not only to get a single output, but also to obtain the confidence distribution of that output as well. For this purpose, the confidence distribution function $CD(\boldsymbol{X}, Y)$ is defined using an idea from the Parzen estimator, as

$$CD(\boldsymbol{X},Y)=\frac{1}{(2\pi)\sigma_{CD}}\cdot\frac{1}{l}\sum_{i=1}^{l}C(\boldsymbol{X},Y_i)\exp[-\frac{(Y-Y_i)^2}{2\sigma_{CD}^2}] \tag{6}$$

where l is the number of patterns used in the comparison process and σ_{CD} is a scaling parameter for this confidence distribution estimation.

As shown in Fig. 4, a typical prediction output of the CPNN consists of a single prediction value (dashed line) and a confidence distribution. In this case, the highest confidence level is not located at the predicted value and the confidence distribution, in this case, is multi-modal. The point that has the highest confidence level is to the right of the average predicted value. There is also another peak, which has a lesser confidence level and is located to the left. For a longer prediction horizon, the confidence distribution can be shown to spread and grow over time by repetitively applying this technique one step at a time. This indicates that as the number of prediction steps increases, the future uncertainty increases as well. Instead of reducing each step of prediction into a single number, consider each peak that occurs in the multi-modal prediction being taken as a new prediction value for the next iteration. This multi-path of prediction is expected to branch out like a tree into the future. As more data becomes available and time marches on, new confidence limits are derived and the uncertainty bounds shrink through appropriate Q-Learning routines [16].

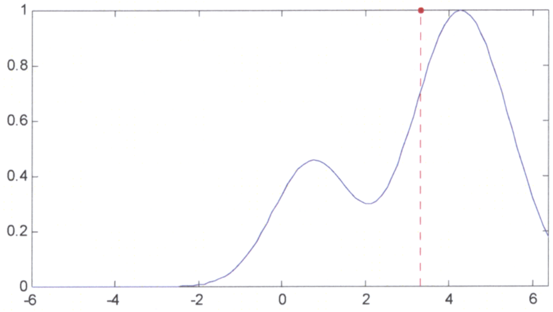


Fig. 4. Output (dashed line) and confidence distribution of the output of the CPNN.

F. Mode Estimation and Usage Pattern Analysis Module

Battery operating modes will be defined such as high/nominal/low temperatures, discharge rates, duty cycles, etc. The mode estimation algorithm will be composed of a fuzzy logic expert (Takagi-Sugeno type) classifier using the fuzzy c-means clustering algorithm taking features extracted from battery sensor measurements and outputting classified operating modes. A usage pattern analysis module will consist of a dynamic neural network (DNN) trained on operating mode/duration inputs and battery life (i.e. capacity reduction) outputs. The CPNN algorithm will be used in conjunction with the output of the usage pattern analysis module to output a remaining useful life (RUL) distribution.

G. Fusion of Prognostics

The RUL distribution results of the CPNN and Usage Pattern Analysis Module will be fused using the concept of linear opinion pooling:

$$RUL_{fused}(t)=\lambda RUL_{failure}(t)+(1-\lambda)RUL_{usage}(t) \tag{7}$$

where, $\lambda \in [0,1]$ is a weighting measure to allow more, equal or less emphasis on the results of the failure RUL results. The prognostic result is a distribution over time for remaining useful battery life.

III. Lithium-ion battery Health Assessment Example

Battery data (voltage, current, temperature, etc.) from several 4-amp hour lithium ion (Li ion) battery cells supplied by United Lithium were collected under different operating conditions (storage and charge/discharge cycling under room and 50° C temperatures). Prototype battery health monitoring algorithms (support vector machine, dynamic neural network, confidence prediction neural network, and usage pattern analysis) were developed and tested on the collected battery data.

A. Data Collection

Charging and discharging was performed on four Li-ion 4-amp hour lithium ion (Li-ion) battery cells at the 1C rate (4 Amps) and either at room temperature or at 50 °C. Current, potential, and temperature were monitored continuously. At approximately every 100 cycles, a set of diagnostics was performed. These included charging and discharging under normal conditions to measure the capacity of the battery, response to step changes in current, and Electrochemical Impedance Spectroscopy (EIS).

Fig. 5 shows the discharge curves and capacity check curves for a lithium ion cell cycled at 50°C. It can be seen that the voltage plateau is at higher potentials while cycling at 50°C. This can be understood from the fact at higher temperatures ionic conductivity increases and also the reaction kinetics tends to be faster. But at higher temperature, degradation mechanisms were also accelerated and so though the apparent performance and the capacity seemed to be higher, the actual capacity available was much lesser. Fig. 6 shows the discharge curves and capacity check curves for a lithium ion cell cycled at room temperature. This figure shows that the discharge curves during cycling at 1C rate and the capacity check curves are closer at room temperature and further supports the argument made above for observations at high temperature cycling experiments.

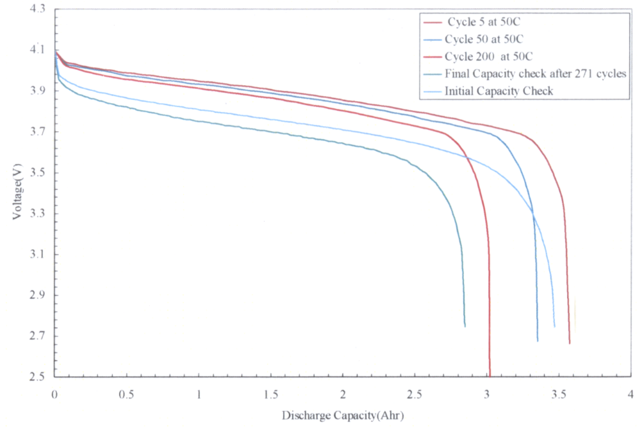


Fig. 5. Discharge and capacity check curves of a lithium ion cell cycled at 50°C

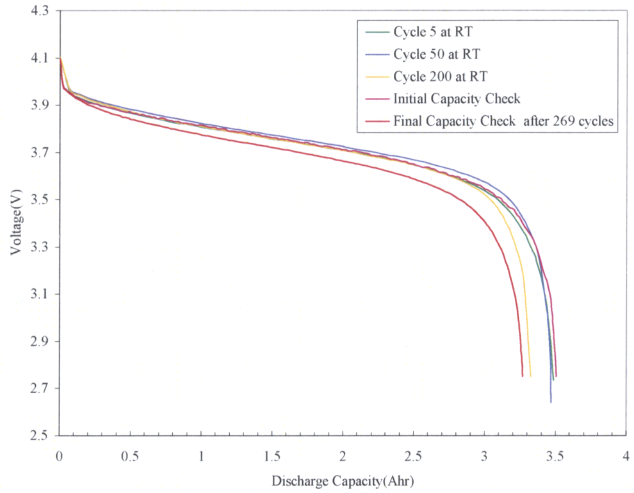


Fig. 6. Discharge and capacity check curves of a lithium ion cell cycled at 50°C

Fig. 7 gives the impedance spectra of a lithium ion cell before and after cycling at 50°C as an illustration. It can be observed that the two semicircles (especially at 100% SOC) are not as distinct after cycling or storage experiments as they were in fresh cells. It is further intriguing to note that the semicircles are smaller after cycling/storage experiments than for fresh cells, which is quite contrary to what is usually observed in literature. This suggests that most of the degradation can be tied to changes in the electrolyte and decreases in low frequency time constants.

Fig. 8 shows the discharge voltage vs. depth of discharge for different cycles. Here the curves move in general from the green bolded curve to the red bolded curve as the cycles increase. Thus, one may consider that each curve in this plane represents a battery health state.

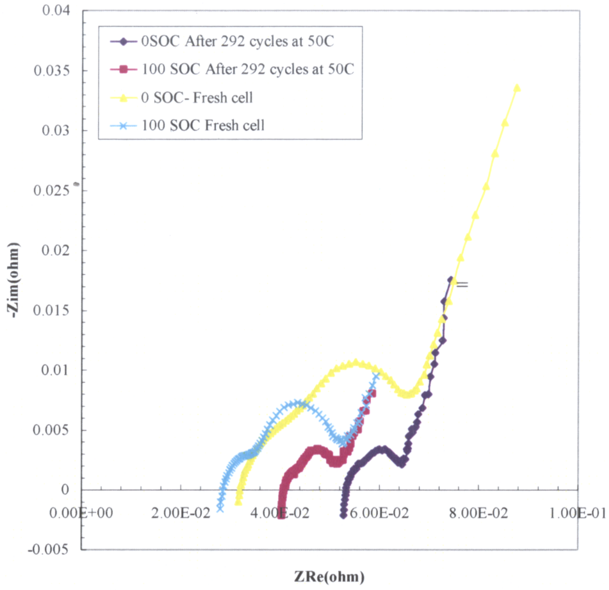


Fig. 7. EIS analysis of lithium ion cells before and after cycling at 50°C.

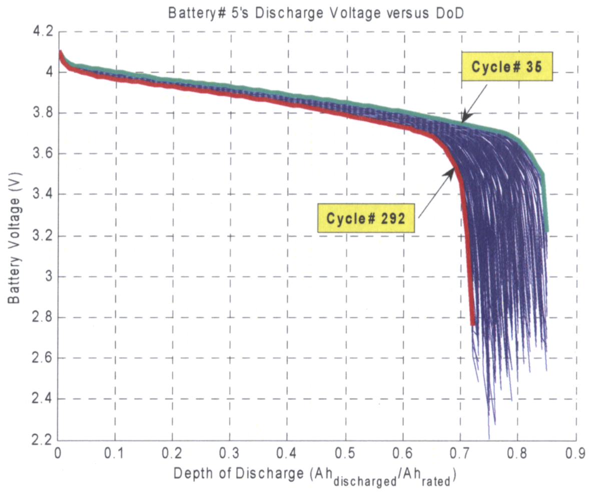
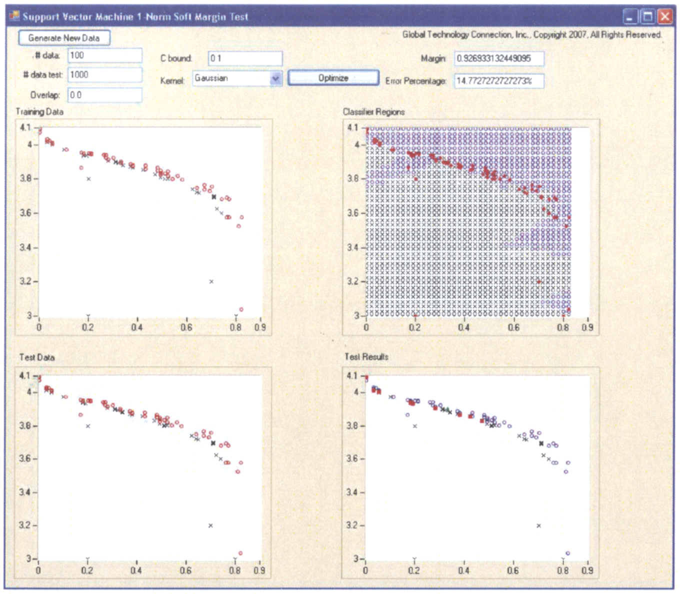


Fig. 8. Battery discharge voltage vs. depth of discharge for various cycles

B. Diagnostic Module: Support Vector Machine (SVM)

A 1-Norm Soft Margin SVM using a Gaussian Kernel was developed as a class library in Visual Studio C++ .NET 2005 to determine if the Li-ion battery was nearing the end of its useful life. Fig. 9 shows a test run of the SVM used to initiate prognostics based on the discharge voltage, depth of discharge, and temperature data. The top left graph of Fig. 9 shows the training data used while the top right shows the resulting classified space.



Support Vector Machine 1-Norm Soft Margin Test

Global Technology Connection, Inc., Copyright 2007, All Rights Reserved.

data: 100

data test: 1000

Overlap: 0.0

C bound: 0.1

Kernel: Gaussian

Margin: 0.926933132449095

Error Percentage: 14.7727272727273%

Fig. 9. Test Run of the 1-Norm Soft Margin Support Vector Machine that maps voltage discharge vs. depth of discharge to healthy or degraded state.

C. *Virtual Sensor*

The Virtual Sensor is realized through a multi-layer, feed-forward, artificial neural network implementation as shown in Fig. 10. The user can specify the number of layers and the number of nodes in the input, output, and each hidden layer. The hidden layers use sigmoid activation functions and the output layer is linear. The neural network is trained using the backpropagation algorithm (gradient descent) to minimize the squared error at the output.

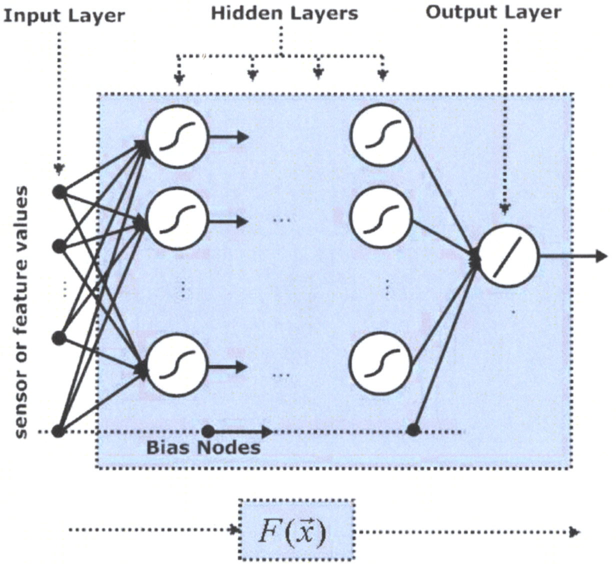


Fig. 10. Virtual Sensor Neural Net Structure

For the purposes of health monitoring, it was noted that each point on the discharge voltage vs. DoD vs. temperature space can be mapped to a state of health value in the interval [0,1] where 0 means very poor health and 1 means good health (similarly for the discharge energy vs. DoD vs. temperature). Therefore, a virtual sensor was used to learn this mapping from the battery data obtained.

Fig. 11 shows the scaled temperature (pink), scaled discharge voltage (light blue), and scaled DoD (green) training data from all batteries, the health estimates, and the learned mapping (red is poor battery health and green is good battery health) for temperatures 50°C. There is a shaded region between the good and bad health regions representing the gradual transition between the two health states. The transition region is defined through the assigned health states from the training data and can have "harder" or "softer" boundaries. Fig. 11 shows a fairly distinct boundary between the two health states. The definition of the assigned health states will depend on the battery application.

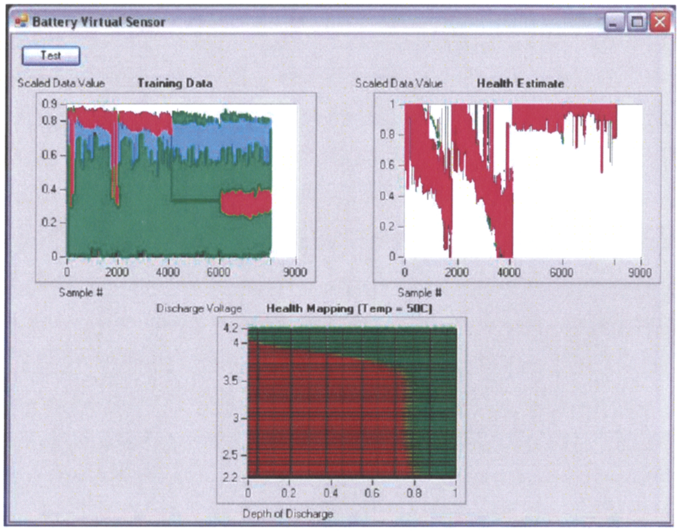


Fig. 11. Virtual sensor mapping of discharge voltage and depth of discharge to a battery state of health measure for cycling at 50°C

D. *Dynamic Neural Network*

Upon each iteration of the DNN algorithm, new data was collected and appended to the historic data, the DNN was trained with the historic data, and then the initial values of the DNN were set with the most recent historic data and a specified number of predictions were made.

Figs. 12 and 13 show the DNN prediction error results for successive predictions using different amounts of discharge amp-hour per cycle training data. As more data was collected and used to train the DNN, the prediction gets better. The dotted red line in each graph represented a predicted path using 100 cycles of training data. The results seem to suggest at least 50 cycles of raw data was necessary to make a good prediction. This can be reduced further if filtering is performed on the data before use in the DNN.

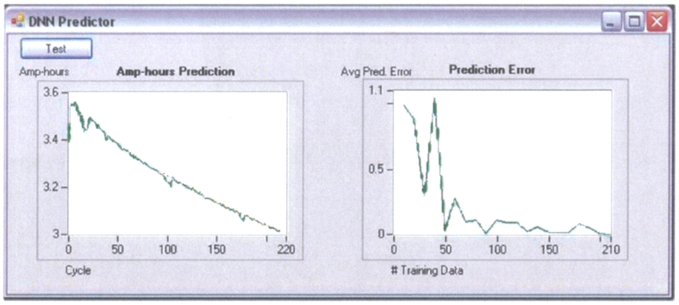


Fig. 12. Dynamic Neural Network predictor for Discharge Amp-hours per Cycle (Temp = 50°C).

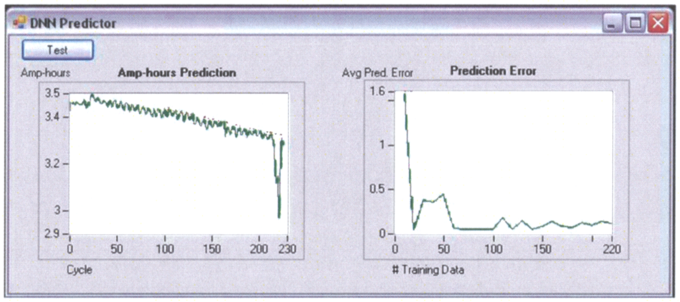
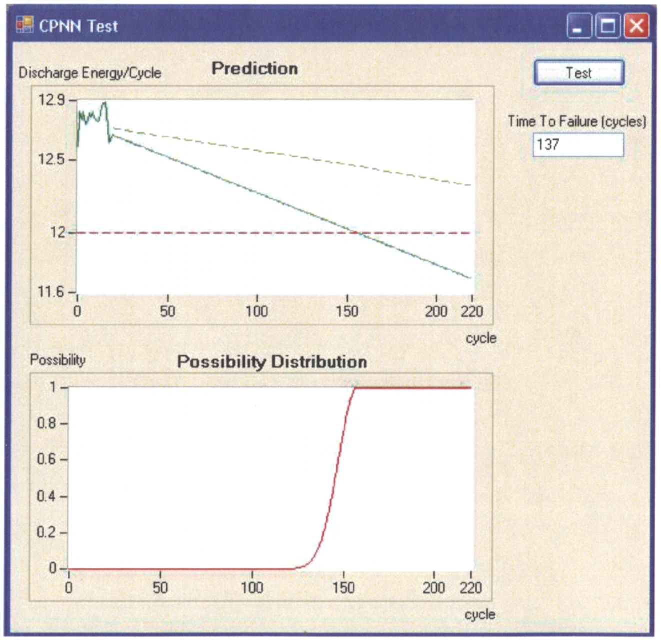


Fig. 13. Dynamic Neural Network predictor for Discharge Amp-hours per Cycle (Temp = 20°C).

E. Confidence Prediction Neural Network

Fig. 14 shows the results of predicting future values of discharged amp-hours per cycle. The dotted red line represents a user defined threshold hazard line where the battery is declared non-usable (3.0 discharge amp hours per cycle). The green data represents the data used to train the CPNN while the dotted yellow lines represent the uncertainty bounds on the prediction. An estimated time to failure in future cycles is given showing the estimated remaining useful life of the battery under current cycling conditions.



Time To Failure (cycles)
137

Fig. 14. CPNN prediction for discharged Ah per cycle.

F. Usage Pattern Health Prediction

The usage pattern health monitoring algorithm estimates the probability of battery failure with time while the battery is being used in different operating modes. For each operating mode, there exists a Weibull probability density function of the form:

$$p(t)=\frac{\beta}{\eta}\left(\frac{t}{\eta}\right)^{\beta-1} e^{-\left(\frac{t}{\eta}\right)^{\beta}},\ t\ge 0 \tag{8}$$

where, $\beta > 0$ is a shape parameter, $\eta > 0$ is a scale parameter, and *t* is time (*t*=0 is a fresh cell).

The probability that the battery will fail after a time, *T*, is the cumulative distribution:

$$\Pr(0\le t\le T)=\int_0^T p(t)dt=1-e^{-\left(\frac{T}{\eta}\right)^{\beta}}. \tag{9}$$

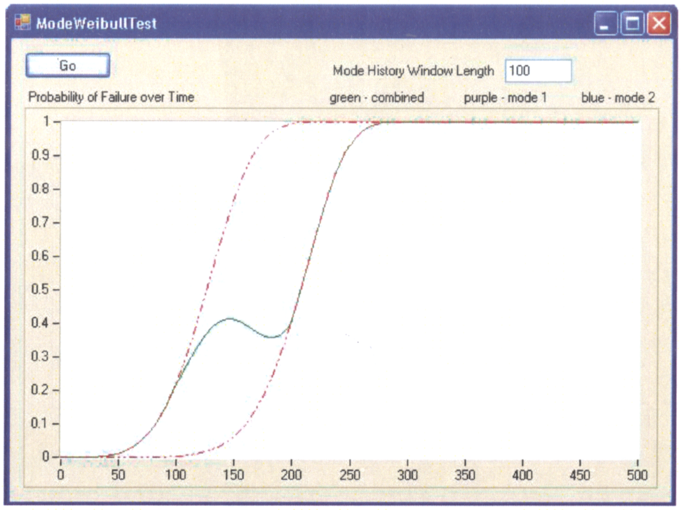
The Weibull cumulative distribution parameters, β and η, can be estimated using known times to failure. Median rank percentage can be estimated using the equation:

$$MR\% \approx \frac{i-0.3}{N+0.4}\cdot 100\% \tag{10}$$

The median rank is the probability of failure for a time-to-failure, and thus the Weibull cumulative distribution is fit to these points to find the parameters. If the times-to-failure are known for batteries operating under different discrete operating modes (temperature, etc.), then a Weibull distribution can be created for each mode. Additionally, if historic operating modes are known, then a probability distribution can be estimated yielding the probability of a battery operating in a mode. Thus an expected probability of failure for a time *T* and historic battery operating mode distribution can be determined via:

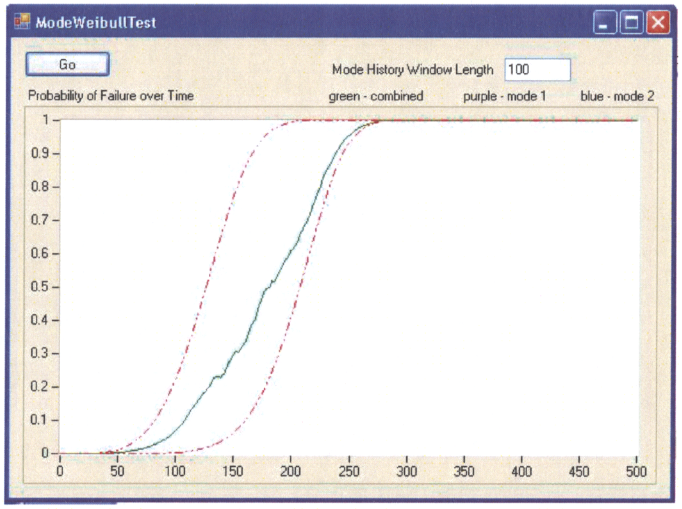
$$\Pr(T\mid \Pr(\text{mode}))=\sum_{k=1}^{M}\Pr(\text{mode}=k)\Pr_k(0<t<T) \tag{11}$$

Figs. 15 and 16 show some usage pattern algorithm testing results using simulated data for two operating modes. Fig. 15 shows that the probability of failure does not have to be monotonically increasing due to the switching of operating modes. Fig. 16 shows the results when the modes are switched between randomly. This probability distribution can be fused with the CPNN possibility distribution using the concept of linear pooling.



Mode History Window Length 100

Fig. 15. Probability of Failure vs. cycle # when switching from operating mode 1 to operating mode 2 at 100 cycles.



Mode History Window Length 100

Fig. 16. Probability of Failure vs. cycle # when switching between operating mode 1 to operating mode 2 randomly.

IV. Conclusion

Several health monitoring algorithms have been developed, implemented and applied to cycling data collected from Li-ion batteries. Initial testing of the proposed algorithms show that they can be used to provide online estimation of the SOH and remaining useful life of Li-ion batteries while taking into account their usage patterns. The virtual sensor was able to learn the mapping of voltage (or energy), depth of discharge and temperature to poor or good battery health states. The DNN provided accurate predictions of the remaining life of the battery when using at least 50 cycles of raw capacity fade data. As more cycle data was used, the prediction accuracy of the DNN improved. The usage pattern health prediction algorithm was able to estimate the probability of battery failure with time while the battery was being used in different operating modes.

Acknowledgment

Authors thank Dr. Nicholas Propes for the development and implementation of the battery health monitoring algorithms and Dr. Tom Fuller of Georgia Tech Research Institute for collecting discharge/charge cycling and impedance data for several Li-ion batteries.

References

[1] Singh, P. and Reisner, D., "Fuzzy Logic-Based State-of-Health Determination of Lead Acid Batteries," *International Telecommunications Energy Conf. (INTELEC 2002)*, Montreal, Canada, Sep 29 - Oct 3, 2002, pp. 583-590.

[2] Kallfelz, A., "Battery Monitoring Considerations for Hybrid Vehicles and Other Battery Systems With Dynamic Duty Loads", *Battery Power Products & Technology*, Vol. 10, No. 3, May/June 2006.

[3] K. Champlin and K. Bertness, "Results of Discrete Frequency Immitance Measurements on Lead Acid Batteries" *International Telecommunications Energy Conf. (INTELEC 2001)*, Edinburgh, Scotland, 2001, October 14 - 18, 2001, pp. 433-440.

[4] Yoon, C.-U., Barsukov, Y., and Kim, J.-H., "Laplace Transfom Impedance Spectrometer and Its Measurement Method", US Patent 6,687,631 B1, February 3, 2004.

[5] Smith, K. A., Wang, C. Y., and Rahn, C. D., "1-D electrochemical lithium-ion battery model for real-time application", *AABC Conference* 2006.

[6] Bhangu, B. S., Bentley, P., A Stone, D., and Bingham, C. M., "Nonlinear observers for predicting state-of-charge and state-of-health of lead-acid batteries for hybrid-electric vehicles", *IEEE Trans. on Vehicular Technology*, vol. 54, no. 3, pp. 783-794, May 2005.

[7] Urbain, M., Rael, S., Davat, B., and Desprez, P., "State Estimation of a Lithium-Ion Battery Through Kalman Filter", *Power Electronics Specialists Conference (PESC 2007)*, June 17-21, 2007, pp. 2804 - 2810

[8] Hadden, G., Bergstrom, P., Bennett, B., Vachtsevanos, G. and Van Dyke, J. "Shipboard Machinery Diagnostics and Prognostics/Condition Based Maintenance: A Progress Report", MARCON 99, *Maintenance and Reliability Conference*, pp. 73.01-73.16, Gatlinburg, TN, May 9-12, 1999.

[9] Vachtsevanos, G., Wang, P., and Khiripet, N., "Prognostication: Algorithms and Performance Assessment Methodologies", Proceedings of MARCON 2000, *Maintenance and Reliability Conference*, May 8-10, 2000, Knoxville, Tennessee, pp. 8.01-8.12, 2000.

[10] Vachtsevanos, G., Wang, P., Khiripet, N., Thakker, A., and Galie, T., "An Intelligent Approach to Prognostic Enhancements of Diagnostic Systems", *Proceedings of SPIE 15th Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls*, Orlando, Florida, April 16-20, 2001.

[11] Khiripet, N., Vachtsevanos, G., Thakker, A., and Galie, T., "A New Confidence Prediction Neural Network for Machine Failure Prognosis", *Proceedings of Intelligent Ships Symposium IV*, Philadelphia, P.A., April 2-3, 2001.

[12] Khiripet, N., Vachtseanos, G., DeLaurentis, D., Mavris, D., and Patel, C. "A Forecasting Methodology with Uncertainties Representation and Causal Adjustment", *Second International Conference on Intelligent Technologies (InTech, 2001)*, November 27-29, 2001, Bangkok, Thailand.

[13] Cristianini, N. and Shawe-Taylor, J., "An Introduction to Support Vector Machines and other kernel-based learning methods", *Cambridge University Press*, New York, NY, 2005.

[14] Cortes, C. and Vapnik, V., "Support-Vector Networks", *Machine Learning*, 20, 1995.

[15] Sun, J., Zheng, N., and Zhang, Z., "An Improved Sequential Minimization Optimization Algorithm for Support Vector Machine Training", *Journal of Software*, vol. 13, 2002, pp. 2007-2013

[16] C.J.C.H. Watins and P. Dayan, "Q-Learning", *Machine Learning*, vol. 8, no. 3, pp. 279-292, 1992.