A Similarity-Based Prognostics Approach for Remaining Useful Life Estimation of Engineered Systems

Jay Lee
University of Maryland, College Park
516 PUBLICATIONS **25,247** CITATIONS

A Similarity-Based Prognostics Approach for Remaining Useful Life Estimation of Engineered Systems

Tianyi Wang, Jianbo Yu, David Siegel, and Jay Lee

Abstract—**This paper presents a similarity-based approach for estimating the Remaining Useful Life (RUL) in prognostics. The approach is especially suitable for situations in which abundant run-to-failure data for an engineered system are available. Data from multiple units of the same system are used to create a library of degradation patterns. When estimating the RUL of a test unit, the data from it will be matched to those patterns in the library and the actual life of those matched units will be used as the basis of estimation. This approach is used to tackle the data challenge problem defined by the 2008 PHM Data Challenge Competition, in which, run-to-failure data of an unspecified engineered system are provided and the RUL of a set of test units will be estimated. Results show that the similarity-based approach is very effective in performing RUL estimation.**

Index Terms—**Health management, Performance assessment, Prognostics, Remaining useful life**

I. Introduction

REMAINING Useful Life (RUL) estimation is the most common task in the research field of prognostics and health management. The data-driven approach for RUL estimation normally relies on the availability of run-to-failure data, based on which the RUL can be estimated, either directly through a multivariate pattern matching process, or indirectly through damage estimation followed by extrapolation to the damage progression [1]. In this paper, we present a novel data-driven approach for RUL estimation, which also starts from damage estimation (often referred to as performance assessment), but followed by a similarity-based matching method for RUL determination.

The approach was chosen based on the following assumptions:

i) Run-to-failure historical data from multiple units of a system/component are recorded; (The term *unit* refers to an instance of a system/component.)

ii) The historical data covers a representative set of units of the system/component;

iii) The history of each unit ends when it reaches a failure condition, or a preset threshold of undesirable conditions, after which no more runs will be possible or desirable. (The history can start, however, from a variable degrading condition.)

Then, a library of degradation patterns can be created from these units with complete run-to-failure data (called training units). A unit whose remaining life will be predicted (called a test unit) also has its historical data recorded continuously. Instead of fitting a curve for a test unit and extrapolating it, the data will be matched to a certain life period of certain training units with the best matching scores. Finally, the RUL of the test unit can be estimated by using the real life of the matched training units minus the current life position of the test unit (Fig. 1).

The detailed methodology of the approach employed to tackle the data challenge problem (defined by 2008 PHM Data Challenge Competition [2]) is introduced in Section II. The experimental data used for the competition is described in Section III; the procedures taken to solve the data challenge problem are elaborated in Section IV; the results are further discussed in Section V. Finally, conclusions as well as the future work on the approach are discussed in Section VI.

II. Methodology

The approach consists of two essential procedures: performance assessment and RUL estimation. Feature extraction before performance assessment is optional, since the sensor readings themselves can be considered features in some cases. Sometimes, the data collected covers various operating conditions of the system. Those conditions will be considered separately with local performance assessment models; in this case, operating regime partitioning [3] is necessary before performance assessment is conducted.

Manuscript received on July 18, 2008. This work was conducted primarily for the Data Challenge Competition organized by the 2008 PHM conference. The work is supported by the U.S. National Science Foundation Industry/University Cooperative Research Center for Intelligent Maintenance Systems (NSF I/UCR Center for IMS) at the University of Cincinnati.

Tianyi Wang is with Department of Mechanical Engineering, University of Cincinnati, Cincinnati, OH 45221 USA (phone: 513-556-3412; fax: 513-556-3390; e-mail: wangti@email.uc.edu).

Jianbo Yu is with Shanghai Jiao Tong University, Shanghai, 200240 China. He is now a visiting student at University of Cincinnati, Cincinnati, OH 45220 USA (e-mail: yjb168@sjtu.edu.cn).

David Siegel is with Department of Mechanical Engineering, University of Cincinnati, Cincinnati, OH 45220 USA (e-mail: siegeldn@email.uc.edu).

Jay Lee is with Department of Mechanical Engineering, University of Cincinnati, Cincinnati, OH 45220 USA (e-mail: jay.lee@uc.edu).

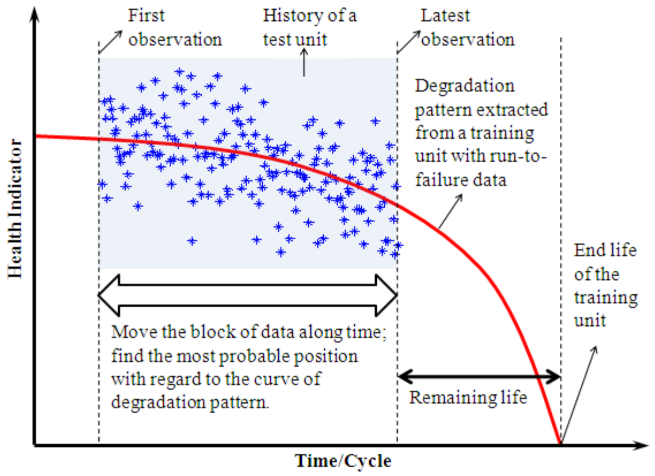
First observation
History of a test unit
Latest observation
Degradation pattern extracted from a training unit with run-to-failure data
End life of the training unit
Health Indicator
Move the block of data along time; find the most probable position with regard to the curve of degradation pattern.
Remaining life
Time/Cycle

Fig. 1. Overview of the RUL estimation strategy. The remaining life of a test unit is estimated based on the actual life of a training unit that has the most similar degradation pattern.

A. *Performance assessment*

The multi-dimensional sensor readings, as well the features extracted from the raw sensor data, are first fused to produce a single Health Indicator (HI). The process to achieve this is called performance assessment. In [4], logistic regression is used to convert the multi-dimensional features into HIs, which can then be used to predict machine performance through ARMA models. However, as found from this study, logistic regression will distort the original degradation pattern (e.g. exponential pattern) of the system, so that curve extrapolation methods based on the output of logistic regression will become less accurate. Specifically, the logistic curve is flat when the value approaches 0 or 1; therefore the HIs produced by logistic regression are less sensitive near the early and end life of the system than in the middle life, which may lead to larger prediction error when extrapolating the HI curve. To preserve the original patterns in the signal/features, a linear regression model is used as performance assessment:

$$y = \alpha + \boldsymbol{\beta}^T \cdot \mathbf{x} + \varepsilon = \alpha + \sum\nolimits_{i=1}^{N} \beta_i x_i + \varepsilon \qquad (1)$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_N)$ is the N dimensional feature vector, y is the health indicator, $(\alpha, \beta) = (\alpha, \beta_1, \beta_2, \ldots, \beta_N)$ is N+1 model parameters, and ε is the noise term. Note that the model is actually the exponential part of a logistic regression model and the training data set can be prepared in the same way as for a logistic regression model, i.e. taking data from healthy and near-failure conditions of the system and assigning the corresponding outputs with 1 and 0 respectively. Linear regression cannot guarantee the transformation to produce a HI within the range of 0 to 1 as logistic regression does; however, this does no harm to RUL estimation.

B. *RUL estimation*

An intuitive method of RUL estimation is to fit a curve of the available data of a testing unit using regression models and extrapolate the curve to certain criteria indicating system failure. However, the available history of a testing unit sometimes may be short; extrapolating the fitted curve may produce large errors whereas the available run-to-failure data are not fully utilized. The same problem exists with the prediction method that employs ARMA models built on the testing data. These methods are suitable when run-to-failure data are unavailable or insufficient, but are not the best choice for the type of problems under investigation in this paper.

Prediction methods based on Neural Networks can take advantage of the run-to-failure data in the model training process. However, these methods lack a systematic approach to select the structure and parameters of the Neural Networks and lack intuition for people to continuously improve their performance.

Since a representative set of units are available (refer to the assumptions in Section I), it is reasonable to first derive multiple representative degradation models from those units, find the models with similar degradation patterns as the test unit and use them as the basis for RUL estimation. The simplest way is, of course, to have one model for each training unit.

The HIs calculated using (1) from each cycle of the training unit form a one-dimensional time series, which can be used to build a model that depicts the pattern of performance degradation from normal to failure. A library of models $\{M_i\}$, each from one training unit, can be established. M_i is usually a deterministic model (e.g. regression models, ARMA models, etc.) that can produce an estimated output y at a given time t:

$$M_i : y = f_i(t), \ -T_i \le t \le 0 \qquad (2)$$

where T_i is the time limit associated with the model. Note that the functions are translated along t so that $t = 0$ corresponds to the last cycle before failure and $t < 0$ to all other cycles in the history. In this paper we only consider discrete time units, or cycles, which have only integer values. Cases in which continuous time units are considered (with variable time intervals between adjacent observations) are very similar.

The form of the function in (2) is application dependent. For example, if the data clearly shows an exponential degradation pattern, an exponential model can be directly used to fit the data; if no apparent patterns can be observed, other models like the ARMA models, can be used.

A certain distance measure between a model M_i and $Y = y_1 \ y_2 \ \ldots \ y_r$, a test unit's HI sequence from r consecutive observations, has to be defined:

$$d(\tau, Y, M_i), \ 0 \le \tau \le T_i - r + 1 \qquad (3)$$

It is a function of τ, the number of cycles that the sequence Y is shifted away from cycle zero of model M_i. The distance measure tells the similarity level that a test unit behaves as model M_i at its history τ. Smaller distance means higher similarity.

The distance function $d\ (\tau, Y, M_i)$ can be defined in various ways. A simple definition can be given by Euclidean distance:

$$d(\tau, Y, M_i) = \sum\nolimits_{j=1}^{r} \left(y_j - f_i(-\tau - r + j)\right)^2 / \sigma_i^2 \qquad (4)$$

where σ_i^2 is the prediction variance given by model M_i.

In fact, M_i can also be a probabilistic model (e.g. Hidden Markov Models) that gives the probability of y at time/cycle t:

$$M_i : p = \Pr(y \mid t, M_i), \ -T_i \le t \le 0 \qquad (5)$$

In this case $d\ (\tau, Y, M_i)$ can be defined as negative logarithm of

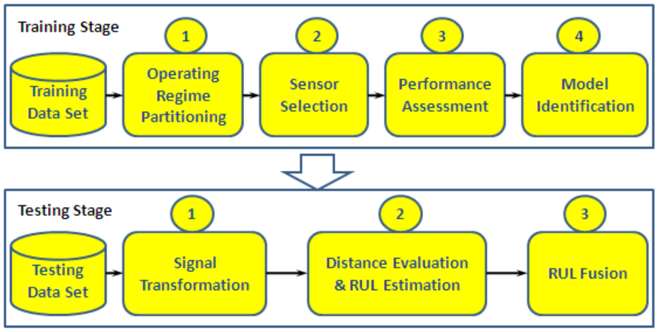

Fig. 2. Procedures of RUL estimation for the Data Challenge problem.

likelihood:

$$d(\tau,Y,M_i) = -\sum\nolimits_{j=1}^{r}\log\left(\Pr(y_j \mid -\tau - r + j, M_i)\right) \tag{6}$$

Once the distance measure is defined, each model M_i in the library can produce one estimated RUL for the test unit:

$$RUL_i = \arg\min_{\tau} d_i(\tau,Y,M_i) \tag{7}$$

At the same time, a distance score can be given to the estimation:

$$D_i = \min_{\tau} d_i(\tau,Y,M_i) \tag{8}$$

The final RUL of the test unit can be estimated through weighted sum of the obtained RULs:

$$RUL = \sum_i w_i \cdot RUL_i\,,\ \sum_i w_i = 1 \tag{9}$$

The weights w_i can be assigned based on the distance score D_i. For example, with the k-nearest-neighbor method, those w_i for the k smallest D_i can be assigned with $1/k$, whereas all other w_i are assigned with 0. In reality, however, the number of nearest neighbors, as well as the way that the weights are assigned to those neighbors, depends highly on the application.

III. Experimental Data

The data set, provided by the 2008 PHM Data Challenge Competition, consists of multivariate time series that are collected from multiple units of an unspecified component. Each time series is from a different instance of the same complex engineered system, e.g., the data might be from a fleet of ships of the same type. There are three operational settings that have a substantial effect on unit performance. The data for each cycle of each unit include the unit ID, cycle index, 3 values for the operational settings and 21 values for 21 sensor measurements. The sensor data are contaminated with noise.

Each unit starts with different degrees of initial degradation and manufacturing variation which is unknown. This degradation and variation is considered normal. The unit is operating normally at the start of each time series, and develops a fault at some point during the series.

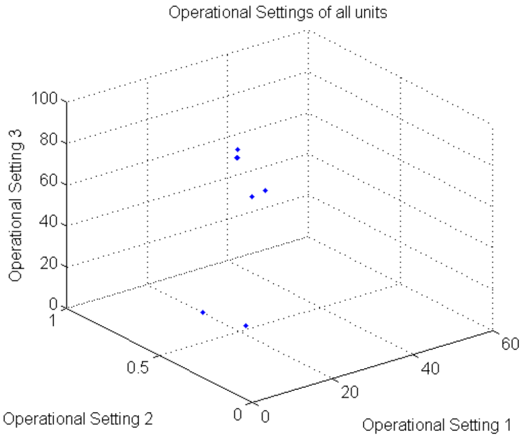

Fig. 3. Operational settings of all units. The six dots are actually six highly concentrated clusters that contain thousands of sample points each. These clusters indicate six discrete operating conditions of the system.

The data set is further divided into training and testing subsets. In the training data set (218 units), the fault grows in magnitude until system failure, at which time, one or more limits for safe operation have been reached, and the unit may not be used for another operational cycle. There is no "hard" failure in the data set; however, the remaining useful life of the last operational cycle of each unit in the training data is considered as zero. In the testing data set, the time series ends some time prior to system failure. The objective of the problem is to predict the number of remaining operational cycles before failure in the testing data set, i.e., the number of operational cycles after the last cycle that the unit will continue to operate. A portion of the testing data set (218 units) is provided first to assist algorithm development and the rest (435 units) is released towards the end of the competition as the validation data set to score the algorithm.

The score for one prediction is defined as the exponential penalty to the prediction error; and the score of an algorithm is defined as the total score S from all the predictions for the K units in the testing data set (defined by the Competition):

$$d_k = \text{estimated } RUL_k - \text{actual } RUL_k$$

$$S_k = \begin{cases} e^{-d_k/13} - 1, d_k \le 0 \\ e^{d_k/10} - 1,\ d_k > 0 \end{cases},\ k = 1,\ldots,K \tag{10}$$

$$S = \sum\nolimits_{k=1}^{K} S_k$$

As we can see, the penalty function is asymmetric, with late predictions penalized more heavily than early predictions. Lower scores are better; a perfect algorithm would score zero.

IV. Procedures

The methodology described in Section II is expanded in more detail when applied to the experimental data. Seven procedures are developed, and are divided into two stages, training (model development) and testing (RUL estimation), as shown in Fig. 2.

A. Training stage

The training stage includes four procedures applied to the training data set.

1) Operating regime partitioning

A quick observation of the sensor data shows that the data exhibit no prominent trend along the life of a unit if the operating settings, indicated by three variables, are not

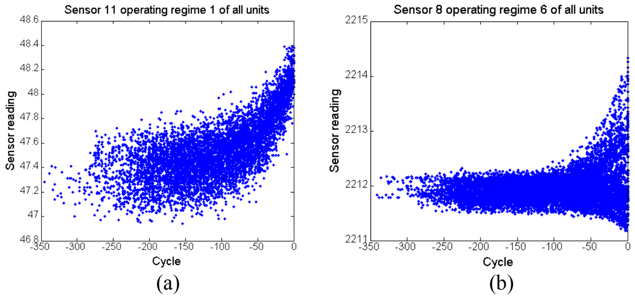

Fig. 4. Selected sensors in selected regimes. (a) A sensor with consistent degradation pattern among all units. (b) A sensor with different degradation patterns among the units.

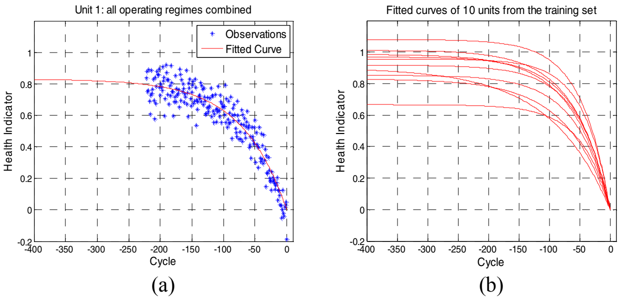

Fig. 5. Curve fitting for training units. (a) One unit (b) Ten units

differentiated. The 3-D plot of the three variables shows that the data concentrated in six clusters, indicating six discrete operating conditions, or regimes (Fig. 3). This observation voids the need of any sophisticated clustering techniques for operating regime partitioning, since the value of the first operational setting is actually enough to distinguish the six operating regimes. Now, each cycle of a unit can be labeled by a regime ID from 1 to 6, replacing the original three variables of operational settings.

2) *Sensor selection*

To be consistent with the models in the form of (2), the cycle indices C of the data for each unit are rearranged: $C^{adj} = C$ – Unit Life. The last cycle of a unit always has the index 0 whereas all the previous cycles have negative cycle indices. In this way, data from all units plotted on a single graph can now show the trend of system degradation (Fig. 4).

Sensor selection starts from observation in each operating regime. A few sensors have single or multiple discrete values, from which it is hard to find trace of system degradation. Most of sensors with continuous values exhibit a monotonic trend during the lifetime of the units. However, some of them show inconsistent end-life trends among the different units in the training data set as shown in Fig. 4-b, which might indicate, for example, different failure modes of the system. It might be possible to first classify the units by failure modes based on these sensors and then process them using different prediction models; this strategy, however, will encounter two challenges. First, the end-life readings of these sensors spread out over a large range, which make it hard to quantize the failure modes without extra information. Second, the failure modes might not be unambiguously identifiable, if not completely indiscernible, at the early age of a unit, and thus might contribute little to RUL estimation when only early history of the unit is available. Therefore, only those continuous-value sensors with a consistent trend (Fig. 4-a) are selected for further processing. Theses sensors are indexed by 2, 3, 4, 7, 11, 12, 15, 20 and 21. Although these nine sensors are selected from observation in this work, it is not hard to define criteria (e.g., using significance test for regression analysis) for an algorithm to select them automatically.

The major challenge, in fact, lies in whether all of the nine sensors selected help to improve the accuracy of RUL estimation, i.e. whether using a subset of those selected sensor can actually improve the accuracy. Some sensors do not show a clear trend as others due to high noise or their low sensitivity to degradation. Including them in the analysis may lower the accuracy of prediction. However, up till the day the results are produced in this study, sensor subset selection is not optimized. Only two combinations from the nine sensors are tested in the experiments: one with three sensors 7, 12 and 15, which show the clearest trend (from observation) in all six regimes, and another one with seven sensors, 2, 3, 4, 7, 11, 12 and 15, leaving out two sensors that exhibit relatively larger variance, and thus, less clearer trend throughout the unit's life. Experiments on the test data sets showed that the choice of seven sensors produced better overall RUL estimation than the one with only three sensors. It is no doubt that the selection of sensors can be further optimized regarding to prediction accuracy.

In the following procedures of this work, the seven sensors, 2, 3, 4, 7, 11, 12 and 15, are used.

3) *Performance assessment*

The selected seven sensors $\mathbf{x} = (x_1, x_2, \ldots, x_7)$ are used to build six linear models in the form of (1), one for each operating regime. The sensor values are used directly without extracting other features. Those models obtained can then transform the sensor data into the HI y; meanwhile, the signals in each regime are scaled to a similar range so that they can be merged again to form a time series for the unit's complete life history.

The sample set $\Omega = \{(\mathbf{x}, y)\}$ required to fit the models consists of selected samples $\mathbf{x}$ from the training data set with purposely assigned y value to them. Those cycles near the end life of all units are selected and assigned with value 0; and those cycles in the early life of the units (which should have a long enough life) are selected and assigned with value 1:

$$\Omega = \{(\mathbf{x}_i, 0) \mid C_i^{adj} > C_{\max}\} \cup \{(\mathbf{x}_i, 1) \mid C_i^{adj} < C_{\min}\} \qquad (11)$$

The selection of C_{max} and C_{min} should create an appropriate size of sample set which contains adequate early-life and end-life samples to train the linear models, and which is also not too large to undermine the representativeness of early-life and end-life properties. In this application, $C_{max} = -5$ and $C_{min} = -300$ (very few units have history longer than 300 cycles) are selected considering that the samples in Ω have to be divided by the six operating regimes to fit six models. Once the model parameters in each regime are obtained, they will be used to transform the complete history of each unit into HI time series (Fig. 5-a), which in turn will be used for the following step.

4) *Model identification*

In this application, the exponential (nonlinear) regression models are used to describe the relationship between the

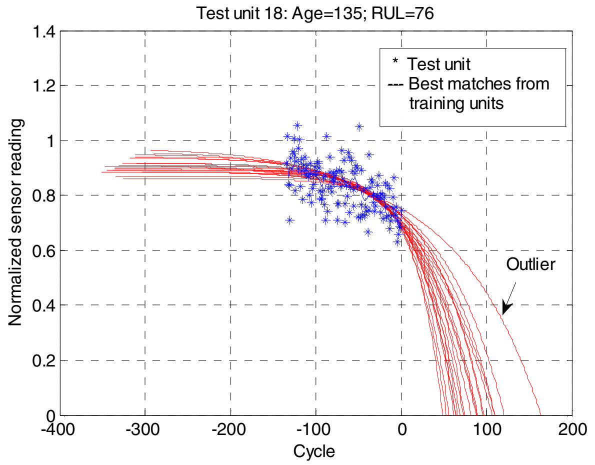

Fig. 6. RUL estimation from the best matched training units that have run-to-failure history. Each curve represents the degradation pattern of one unit. The final RUL of the test unit is estimated based on the RULs given by each matched training unit.

adjusted cycle index C^{adj} and the HI y:

$$y = a(e^{bC^{adj}+c} - e^{c}) + \varepsilon \tag{12}$$

where a, b, and c are the model parameters to be determined, and ε is the noise term. The term e^c is used to force the model to give $y = 0$ when $C^{adj} = 0$. The model for each training unit i will have one set of parameters (a_i, b_i, c_i) as well as the estimated noise variance σ_i^2 that is required by (4).

The functions described in (2) can then be expressed as:

$$M_i : y = f_i(t) = a_i(e^{b_i t + c_i} - e^{c_i}) \tag{13}$$

As found, the units have different life expectancy and wear out at different rates, which are shown in Fig. 5-b.

B. Testing stage

The testing stage includes three procedures applied to each unit in the testing data set.

1) Signal transformation

This procedure utilizes the parameters found from the first three procedures in the training stage. For each unit in the testing data set, the selected sensors' data will be classified by operating regimes, transformed by the linear models for performance assessment obtained during training, and merged to obtain an HI sequence Y. The corresponding cycle indices use the original non-adjusted ones, i.e., 1, 2, …, r.

2) Distance evaluation and RUL estimation

In this study, the HI sequence Y is first filtered using the moving average method. And then the distances between a test unit and each of the models $\{M_i\}$ are evaluated using (4), (13) and (8); the RULs are obtained using (7) and form a RUL pool.

3) RUL fusion

A single, final RUL will be calculated from the RUL pool in three steps: candidate selection, outlier removing, RUL determination.

i) Candidate selection

All the RULs are first sorted by the distance scores given by (8) in ascending order. Top-ranking RULs are selected. A cut-off distance score is set to a 25% increase of the smallest score using constraint $D_i \le 1.25D_1$. If too few RULs remain, a fixed number of top-ranking RULs will be selected.

ii) Outlier removing

RUL estimation for a unit with short history tends to produce great uncertainty or variance, which means unreasonably long or short estimation is likely to appear in this situation. Therefore, those exceptionally long RULs (e.g. larger than 190 cycles) will be removed; those short RULs that make a test unit's total life (RUL + the current life of the test unit) exceptionally short (e.g. 125 cycles) will be removed, too, due to the fact that a unit's life is less likely to go below a certain limit as shown by the statistics of the training units' true life. Those RULs that passed the above criteria are illustrated in Fig. 6. Each curve represents one unit and gives one RUL estimation.

Then, more outliers in the RULs, if enough is left, will be removed using constraint ($Q_{0.5} - 3(Q_{0.5} - Q_{0.25}) < RUL_i < Q_{0.5} + 2(Q_{0.75} - Q_{0.5})$), where $Q_{0.25}$, $Q_{0.5}$, and $Q_{0.75}$ are the first, second and third quartiles of the RULs left from the previous outlier-removing process. Note that the outlier condition for larger RULs is stricter than for smaller RULs.

iii) RUL determination

The final RUL is computed from a weighted sum of the RULs that pass the previous step. Considering the scoring method given in (10), averaging all RUL estimations is inappropriate. Instead, the following weighting method is used, which emphasizes on the upper and lower boundary only (considering the exponential penalty to prediction errors).

$$RUL = (13/23)\cdot\min_i(RUL_i) + (10/23)\cdot\max_i(RUL_i) \tag{14}$$

V. Results and Discussions

The RUL estimation method is tested and tuned using the first batch of testing data (218 units) provided by the PHM Competition. The actual remaining life of these units is kept unknown and only the total score of 218 estimated RULs is computed and returned to the user at each submission of results. Therefore, the average prediction error measured in cycles is unavailable. The average prediction error measured in scores defined in (10), however, is not intuitive, because several large errors among the many predictions will completely dominate the final score. For example, a late prediction of 60 cycles will give a penalty score of 402.43, which could happen when only a very short history (e.g. 25 cycles) is available. As comparison, a late prediction of 10 cycles gives only 1.72 penalty score.

As the scoring mechanism defined in (10) doesn't provide a penalty limit for a single prediction, it is rather risky to make a large prediction error for those units with short history. It is also found that large errors of both early prediction and late prediction are possible. The authors believe that, in order to mitigate the risk, a certain risk model (e.g. from Bayesian decision theory) can be derived from the distribution of the actual life of the training units to adjust the RUL estimations made by the algorithms. Till this work is reported, however, the method used for RUL adjustment is rule-based and the parameters are determined from experiments. For example, an

RUL larger than 135 is adjusted to 135 directly to reduce risk.

The RUL adjustment rules, as well as the RUL fusion procedures discussed in a previous section, were developed based on the experiments on the first portion of the testing data set released along with the training data set. Near the end of the competition, the algorithm and rules were applied to the validation data set which contains 435 samples (the validation data set did not allow a trial-and-error type of submission). A total score of 5636.06 is achieved, which is the overall best in the competition.

VI. Conclusion and Future Work

The approach presented in this paper is very effective for the data set provided by the competition, and is expected to demonstrate similar performance for applications under the same assumptions made in Section I.

The approach, however, has great potential of improvement in, for example, automating parameter selection and generalization for other prognostics situations. The following future work could be pursuit:

i) Automate operating regime partitioning. Clustering algorithms, although seemingly unnecessary due to discrete operating conditions, can be used in this application to automate the process; for problems with continuous operating conditions, however, it is a must.

ii) Automate sensor selection. The sensors candidates can be first filtered through unsupervised feature selection methods, e.g. test of significance for fitting desired regression models. Then the optimal sensor subset can be determined through supervised feature selection methods, using the overall prediction score as the selection criteria.

iii) Use analytical methods to determine the parameters for RUL fusion. Some of the parameters, such as the upper and lower limits for candidate RULs, can be determined by investigating the statistics of the training units' actual life. Others can be optimized, either using the overall prediction score as the target function (if possible), or using a risk model built on the penalty of large prediction errors. The risk model can be used to make the adjustment to the final RUL estimation, which is now largely determined through experiment.

iv) Test other models than an exponential model within the framework of the approach. Both deterministic models and probabilistic models can be used. Other data sets that do not exhibit clear exponential degradation patterns can be employed.

References

[1] K. Goebel, B. Saha, and A. Saxena, "A comparison of three data-driven techniques for prognostics," *Failure prevention for system availability, 62th meeting of the MFPT Society* - 2008, pp. 119–131.

[2] 2008 PHM data challenge competition, [Online] http://www.phmconf.org/OCS/index.php/phm/2008/challenge

[3] T. Wang, and J. Lee, "The operating regime approach for precision health prognosis," *Failure Prevention for System Availability, 62th meeting of the MFPT society* - 2008, pp. 87-98.

[4] J. Yan, M. Koc, and J. Lee, "A prognostic algorithm for machine performance assessment and its application," *Production Planning & Control*, Vol. 15, No. 8, 2004, pp. 796–801.