Heng, Soong Yin Aiwina and Tan, Andy C. C. and Mathew, Joseph and Yang, Bo-Suk (2007) Machine Prognosis with Full Utilization of Truncated Lifetime Data. In Gelman, L., Eds. *Proceedings 2nd World Congress on Engineering Asset Management and the 4th International Conference on Condition Monitoring*, pages pp. 775-784, Harrogate, UK.

Machine Prognosis with Full Utilization of Truncated Lifetime Data

Aiwina Heng*, Andy C. C. Tan*, Joseph Mathew**, and Bo-Suk Yang
**CRC for Integrated Engineering Asset Management
School of Engineering Systems
*Queensland University of Technology
GPO Box 2434 Brisbane QLD 4001
AUSTRALIA
E-mail: a.heng@qut.edu.au

Abstract

Intelligent machine fault prognostics estimates how soon and likely a failure will occur with little human expert judgement. It minimizes production downtime, spares inventory and maintenance labour costs. Prognostic models, especially probabilistic methods, require numerous historical failure instances. In practice however, industrial and military communities would rarely allow their engineering assets to run to failure. It is only known that the machine component survived up to the time of repair or replacement but there is no information as to when the component would have failed if left undisturbed. Data of this sort are called truncated data. This paper proposes a novel model, the Intelligent Product Limit Estimator (iPLE), which utilizes truncated data to perform adaptive long-range prediction of a machine component's remaining lifetime. It takes advantage of statistical models' ability to provide useful representation of survival probabilities, and of neural networks ability to recognise nonlinear relationships between a machine component's future survival condition and a given series of prognostic data features. Progressive bearing degradation data were simulated and used to train and validate the proposed model. The results support our hypothesis that the iPLE can perform better than similar prognostics models that neglect truncated data.

1. INTRODUCTION

Increasingly complex and refined machines prevalent today require highly sophisticated maintenance. Domestic plants in the United States spent more than $600 billion to maintain their critical plant systems in 1981 and by 1991, the costs had increased to more than $800 billion and topped $1.2 trillion in 2000 (Raytek, 2004). An even more alarming fact is that one third to one half of maintenance expenditure is wasted through ineffective maintenance management methods. It has been argued that this trend is similar in Australia and industry can no longer absorb this incredible level of inefficiency. Therefore, there is a pressing need to continuously develop and improve modern intelligent maintenance systems, in order to compete in the global market. Machine component prognostics determines whether a fault is impending and estimates how soon and likely a fault will occur with little human expert judgement. It minimizes production downtime, spares inventory and maintenance labour costs.

Even though the concept proposed may be applied to the prognosis of various machine components, this work focuses on rolling element bearings for illustration purposes as bearing failure is one of the foremost causes of machinery breakdown. For bearing life prediction, data like vibration signals, oil particle count and temperature are usually collected periodically. Data features like RMS are then extracted and plotted in order to trend the changes in feature values. These changes indicate the progression of fault severity over time. However, there are still remaining problems to be solved in the research area of bearing prognostics.

1.1. Remaining problem: negligence of truncated data

Prognostics models, especially probabilistic models (Sutherland et al., 2002, Vlok et al., 2004, Groer, 2000) and artificial intelligence models (Gebraeel et al., 2004, Huang et al., , Qiu et al., 2003, Wang and Vachtsevanos, 2001) require numerous historical failure instances. In practice however, industrial and military communities would rarely allow their engineering assets to run to failure. Most of the time, once a defect has been detected in a machine component, the component is extracted and replaced before it fails. Therefore the cut-off point at which the component will cease to function is not always known or recorded. It is only known that the machine component survived up to the time of repair or replacement but there is no information as to when the component would have failed if left undisturbed. Data of this sort are called truncated data.

The intelligent machine prognostics models proposed in the literature have not taken this into consideration. When all the component truncation times in historical data are treated as the component failure times, the prognostic model will produce biased estimates (underestimation) of the time to failure. Since in most instances a component is replaced once a fault is detected, treating the replacement times as failure times defeats the purpose of prognostics because it is the duration the failing component can survive beyond this point that is of interest (Figure 1). It is not uncommon that a component's remaining useful life (from the point where a defect is detected) is substantially more than its L10 life. It is a prognostician's goal to recommend a maintenance schedule that does not interrupt production or wastefully replace components that still have useful remaining life. It is the ability to estimate this remaining lifetime that is critical to optimal maintenance scheduling.

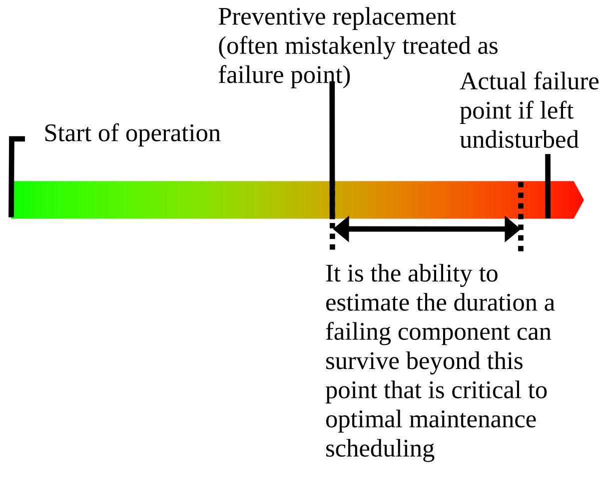

Figure 1 Timeline of a machine component's operational life.

On the other hand, prediction models that carefully omit truncated data from the training data sets will worsen the problem of data unavailability in real life. Progressive component degradation data are already scarce due to irregular measurement recording or/and the huge amount of time it takes to accumulate enough sets of failure data. For example, a bearing can last for several years even under harsh operating conditions. Therefore, a good prognostics model must be able to maximize use of available data.

1.2. Problems associated with the existing neural network prognostics models

Current prognostics approaches fall into 3 main categories: statistical approaches, model-based approaches and artificial intelligence approaches.

Statistical approaches (Vlcek et al., 2003, Groer, 2000, Schömig and Rose., 2003) typically involve fitting probabilistic failure distribution to historical data. These approaches are the least complex and may be the only alternative in not-so-critical or low-failure-rate situations. The next logical extension to these statistical models is to correlate failure instances records with more specific health condition data that are directly related to the system being monitored.

Model-based approaches can be the most accurate when a correct and accurate model is available. However, it is very difficult to build mathematical models for complex systems. It requires system-specific mechanistic knowledge. (Jantunen, 2004) stated that the wear of rotating machinery components is still not fully understood today. Most model-based prognostics methods (Li et al., 2000, Qiu et al., 2002) focus on the prediction of crack propagation. However, there is a large variety of other failure modes and the prognostician needs to correctly identify the fault type in question. Even if that has been accomplished, defect growth is not a deterministic process. Virkler et al. (1979) has shown that even under well-controlled experimental conditions, crack growths of a set of identical components are vastly different. It is also difficult to apply crack growth models in practice because they require the knowledge of the exact geometry or/and orientation of the crack, which are usually very irregular and cannot be identified without disassembling the machine component.

Compared to model-based models, artificial intelligence models make much fewer assumptions about the system and its operating conditions. One popular artificial intelligence prognostics technique in the literature is artificial neural networks. Neural networks can be tuned using well-established algorithms to provide desired outputs directly in terms of vibration signals (Roemer et al., 2005). They learn from examples and aim to capture the relationship among data. A neural network consists of a layer of input nodes, one or more layers of hidden nodes, one layer of output nodes and connecting weights. The network learns the unknown function by adjusting its weights with repetitive observations of inputs and outputs.

Neural networks have produced comparable and, in some cases, superior results to standard mechanistic or statistical models in various disciplines (Reibnegger et al., 1991, White, 1989). In recent years, several methods employing neural networks have been proposed for bearing prognosis. Tse and Atherton (1999) have approached bearing prognosis as a time series prediction using a recurrent neural network (RNN). These models perform single-step-ahead predictions to output the predicted vibration signal feature(s) at the next immediate time step. However, single-step predictions rarely in reality raise the bar from diagnostics to prognostics. 1 time step in a plot of vibration feature measurement for prognostics can be only 15 minutes. A prognostics horizon of 15 minutes or even 1 day is not of much help to optimal maintenance scheduling.

Gebraeel et al. (2004) and Huang et al. attempted to predict the actual bearing failure time and based their bearing prognosis on the assumption that all bearing degradation signals possess an inherent exponential growth. They proposed a feed forward neural network model that derives exponential parameters α and β to fit each bearing degradation signal with the best exponential fit of the form $\alpha e \beta t$. Huang et al. built on the above mentioned method and used a feed forward neural network to generate a predicted failure time based on 100 interpolated points which are exhaustively searched from each bearing degradation data set. This model assumes that the bearing degradation data are continuously monitored and the 100 points can be uniformly interpolated at any prediction point. It is unclear that if these seemingly complex neural network models offer analysis more sophisticated than that provided by a simple exponential curve fitting and extrapolation method. Exponential extrapolations often have a large region of uncertainty. Due to the probabilistic nature of bearing integrity and operating condition, defect propagation rates are vastly stochastic. If the growth of a bearing defect differs from the exponential pattern assumed by the above-mentioned models, the bearing degradation will be poorly extrapolated. Even for proper assumptions about the exponential defect growth, the prediction confidence interval of extrapolations often diverges to impossible values. Extrapolating beyond that range can lead to misleading results. Besides, successive prediction outputs may vary vastly and seem confusing. Without an indication of probability distribution, it is rather difficult for maintenance personnel to make maintenance decisions. Lastly, these feed forward neural network models also require knowledge of the current bearing operational age (the time from the start of a bearing operation/degradation to the current prediction point).

2. AN INTELLIGENT PRODUCT-LIMIT ESTIMATOR THAT FULLY UTILIZES TRUNCATED DATA

2.1. Model Architecture

We propose the Intelligent Product Limit Estimator (iPLE) that maximizes the use of available data. While this model also employs a neural network as one of its tools, it differs from the above techniques in several respects. The network's inputs and prediction outputs will be interpreted as probabilities, incorporating statistical reliability or failure distribution analysis. Truncated data sets will be fully utilized and incorporated directly into the training set, not by using an artificial cut-off time, but rather by using the probability that they will not fail before a certain time as the training signal. Also, the network will separate the failure cases into output classes based on their failure time. Therefore the network output will be a predicted survival curve for individual machine components (bearings, in this case). Lastly, in cases where components failure instances (failure times) are recorded but not every failure record comes with trending data (e.g. vibration data), iPLE maximizes use of these failure instances records. Conventional neural network models, on the other hand, can only use the failure records which come with trending data. This is because iPLE's baseline estimation of survival probabilities is based on historical failure instances.

The network proposed is a feed-forward input-delay network, which has 7 inputs, $x_t, x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, x_{t-5}$ *and* x_{t-6} (the current bearing degradation indicator value and 6 delayed values, which are the inputs from 6 previous time steps). There is 1 hidden layer made up of 10 hidden nodes. For illustration purposes we use 5 output nodes here. However, more outputs can be used especially when more data readings are available. The network is trained using a gradient descent algorithm with momentum back-propagation.

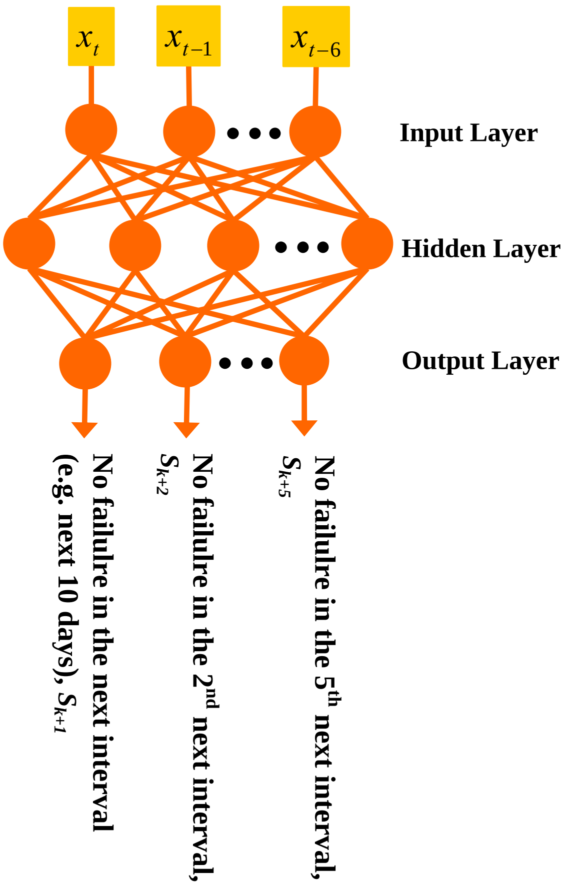

Figure 2 Architecture of the probability neural network used in iPLE

Let k denote the "current" time interval. The first output node represents the probability of the bearing surviving the next time interval (e.g. 1 time interval = 10 days), S_{k+1}; the second node represents the probability of the bearing surviving the 2nd next time interval (between day 10 and day 20), S_{k+2}; and so on, up to the next 5th interval (50 days) (Figure 2) . There will be another class, the 6th class, which represents the class of bearings which may survive beyond day 50. The activations of the output units were trained with and interpreted as the probability that the bearing would survive up to that time.

2.2. Model Training

Representation of training data vectors

The training vector T for each bearing in the training set consists of the survival probability S_{k+n} for each time interval k. In our example of 5 prediction horizons ($n=1,2,\ldots,5$), a training vector for the kth interval would have the format: $T_k = [S_{k+1}; S_{k+2}; S_{k+3}; S_{k+4}; S_{k+5}]$.

Training data vectors for *complete* degradation data sets

Data sets are considered *complete* if the bearing has reached a predetermined failure threshold when removed from the machine. For these complete failure data sets, the network will be trained with values of *1* for outputs up to the last observed survival time interval, and *0* thereafter. For example, a data set of a bearing that fails at day 22 (within the 3rd time interval) would have a training vector $T_k = [1; 1; 0; 0; 0]$.

Training data vectors for *truncated* degradation data sets

Data sets are considered *truncated* if the bearing has not reached the predetermined failure threshold when removed from the machine. For these truncated data sets, the network will similarly be trained with values of *1* only up to the last observed survival time. The outputs for later time intervals will be computed as survival probabilities using a variation of the standard Product Limit Estimation (also known as the Kaplan-Meier estimation) to the true survival rate of the complete data sets. The Kaplan-Meier's 1958 paper (Kaplan and Meier, 1958) is one of the top 5 most cited papers in the field of Sciences. The aim of using this method is to produce the most accurate survival probability possible taking into account all of the information available. In this work, the standard Product Limit Estimator (PLE) which computes survival probability for the *total* set of samples under study is modified to the Intelligent Product Limit Estimator (iPLE) which produces real-time survival probability for *individual* bearing samples.

We define the risk of failure in n next time interval, $risk_{k+n}$, as the conditional probability that a bearing will fail in time interval $k+n$, given that they have not failed up to time interval k-1. Let F_{k+n} denote the number of failures in time interval $k+n$ and R_{k+n} the number of bearings being at risk in time interval $k+n$. Then the risk of a suspended bearing failing in the kth time interval if left undisturbed, is defined as $risk_{kn} = F_{k+n} / R_{k+n}$.

As an example, consider a study containing a total of 30 bearings. Suppose that 1 bearing failed in the 1st time interval (time interval $k+1$), 2 more failed and 1 got suspended in time interval $k+2$, and then another 3 failed in interval $k+3$. The training vector in interval k for the suspended bearing would have values of *1* up to interval *k+2* ($T_k = [1; 1; ?; ?; ?]$). The training vector values for the following intervals are computed using the modified PLE method. Firstly, we know that the number of bearings being at risk in the 3rd time interval, $R_{k+3} = 30\text{-}1\text{-}2\text{-}1 = 26$. (Note that only the bearings that are still being monitored are at risk, which means failed bearings and suspended bearings are not considered at risk.). Then, the risk of this suspended bearing failing in the 3rd time interval if left undisturbed, $risk_{k+3} = F_{k+3}\ /_{3k+3} = 3/26 = 0.15$.

Further suppose that 3 more bearings failed and 2 other suspended in the 4th interval. The risk of the suspended bearing failing in the 4th interval if left undisturbed, $risk_{k+4} = F_{k+4} / R_{k+4} = 3/21 = 0.143$. If 4 more failed in the 5th interval, $risk_{k+5} = F_{k+5} / R_{k+5} = 4/18 = 0.222$. The Product Limit estimation of the survival curve, *S*, tracks the cumulative survival probability for any time interval in the study, using the risks in the following fashion:

$$S_{k+n} = \begin{cases} 1, & 0 \le k+n \le L(i) \\ S_{k+n}(1 - risk_{k+n}), & k+n > L(i). \end{cases}$$

where *L(i)* denotes the last observed survival time of the individual bearing *i* (*i = 1, 2, ... , 30*. in this example). Note that we simply use the last observed survival time *L* of each suspended bearing as the starting time, rather than time *0*, to compute appropriate training probabilities. Continuing the above example, $S_{k+1} = 1.0$; $S_{k+2} = 1.0$; $S_{k+3} = 1.0(1\text{-}0.15) = 0.85$; $S_{k+4} = 0.85(1\text{-}0.143) = 0.728$; $S_{k+5} = 0.728(1\text{-}0.222) = 0.566$.

The training vector for that suspended bearing data set in interval *k* will be $T_k = [1; 1; 0.850; 0.728; 0.566]$. For each individual output node *k+n*, this training signal represents the input sample's probability of membership in the class represented by that node, i.e. the probability that the bearing will fail in time interval *k+n*. Collectively, the survival probability values represent an expected survival curve for that bearing at the time of prediction (Figure 3).

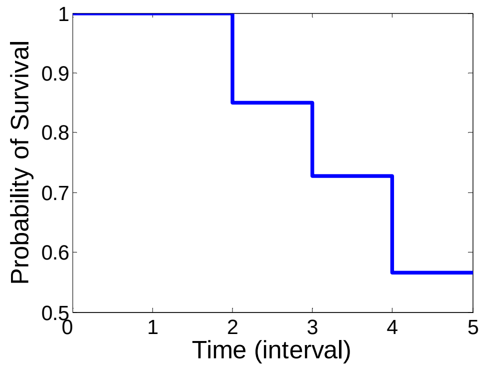

Figure 3 The survival curve represented by the network output node values.

Training the network's predictive power

During training, the training vectors of the training set are repetitively presented to the neural network. The neural network attempts to produce output values that are as close to the target vectors as possible. Connection weights within the network are changed during training using the back propagation of errors algorithm. Throughout training, the connection weight structure of the neural network evolves, and the response to a given input vector changes.

3. VALIDATION OF THE INTELLIGENT PRODUCT-LIMIT ESTIMATOR (iPLE)

After training, the predictive abilities of the neural network can be evaluated using the bearing degradation data from the testing set. When a bearing's degradation indicator values at the current time *t* and 6 previous time steps, x_t, x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, , x_{t-5} x_{t-6} are fed into the input nodes, along with the feedback values from the hidden nodes, the network will produce an output vector Y_t. As the next set of input values becomes available, a new updated output vector will be produced, generating a new survival probability curve. This method continuously tracks the survival probability of the bearing being monitored in the future time intervals, at any time *t*, in the following fashion:

$$S_{t_{k+n}} = \begin{cases} 1, & 0 \le t_{k+n} \le L(i) \\ S_{t_{k+n-1}}(1 - risk_{t_{k+n-1}}), & t_{k+n} > L(i). \end{cases}$$

Each output vector has the format:

$$Y_t = [S_{t_{k+1}}; S_{t_{k+2}}; S_{t_{k+3}}; S_{t_{k+4}}; S_{t_{k+5}}],$$

which can also be plotted as the survival probability curve for that bearing, estimated at time *t*.

3.1. Simulation of Bearing Failure Data Sets

Before sufficient real-life bearing failure data sets become available, the Matlab model for simulating progressive bearing degradation vibration data that the author has developed previously was used to generate bearing inner race degradation data sets for the trial training and testing of the proposed model(Figure 4).

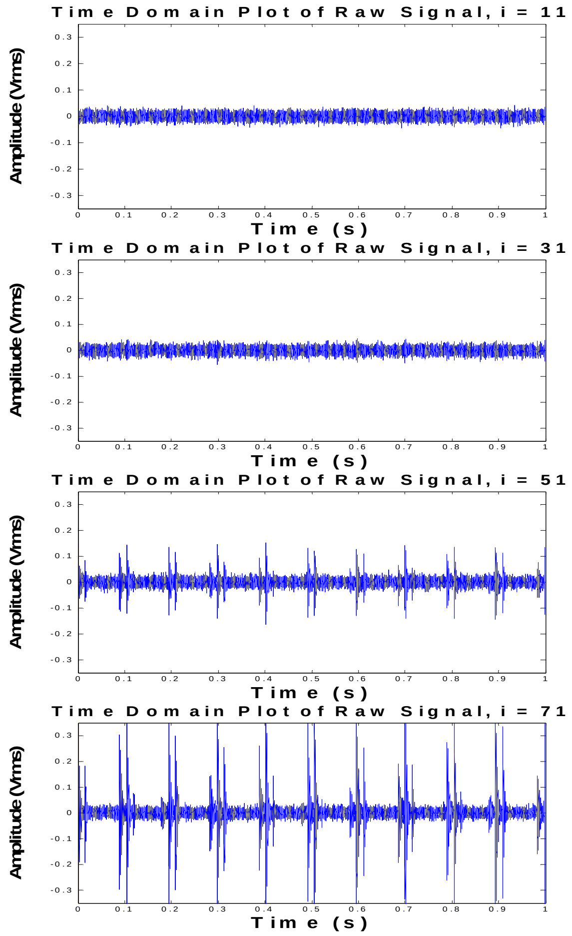

Figure 4 One plot of every 20 data sets generated – with increasing defect impulse.

These simulated signals have defect impulses that increase at different rates and discontinuities. Therefore, the time of reaching the failure threshold are different for each data set, since all data sets were assigned the same level of predetermined failure threshold (Figure 5).

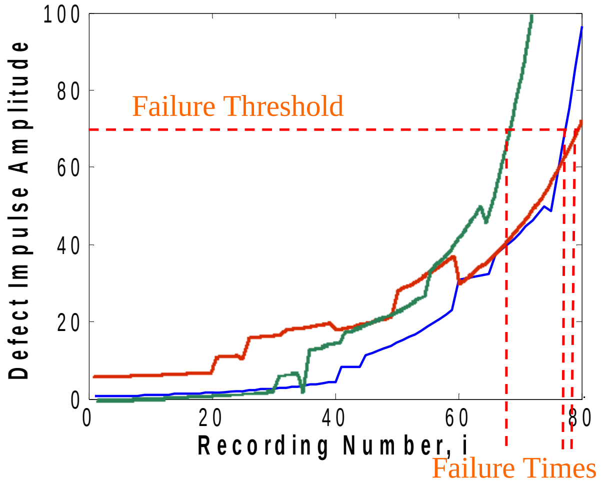

Figure 5 Simulated failure data sets with defect impulses that increase at various rates and discontinuities

The generated signals were high-pass filtered to separate bearing fault frequency signals from dominant high-frequency resonant signals. We used a Self-organizing Map (SOM) to combine 3 extracted features (peak, kurtosis and entropy estimation) into a single degradation indicator (Heng et al., 2006). Even though research has identified that frequency domain features can help reflect the machine condition difference and influence of background noise, we found that for our simulation tests (with a constant level of simulated background noise and operation conditions), the time domain features selected suffice. For more information on simulating these data and using SOM quantization error to combine signal features, please refer to the above-mentioned paper (Heng et al., 2006).

Truncations were imposed on some of the data sets at different points in the simulated bearing lifetime. To do this, the degradation indicator values extracted from suspended data sets were presented to the prognostics model only up to the point of imposed truncation. In this way, the predicted survival probability could be compared to the actual survival time since we actually know when the failure threshold was reached in the simulated data.

40 sets of progressive degradation data were generated by using the simulation model. Each set of data consists of 80 recordings (time steps, *t*). One recording could represent for example, 30 minutes or several days in reality. In our test, 4 time steps are grouped as 1 time interval. Therefore there are a total of 20 intervals in each data set. The prediction horizon is 5 time intervals

(n = *1, 2, …, 5*). One training vector ($T_k = [\, S_{k+1}; S_{k+2}; S_{k+3}; S_{k+4}; S_{k+5}]$) is generated for each interval.

30 of the 40 simulated data sets were assigned for training (Table 1) and the remaining 10 for testing (Table 2). Truncations were randomly imposed on 1/3 of the 30 training data sets.

Data set	Failure time (time step)	Truncation time if any
1	74	
2	62	truncated at 57
3	63	
4	66	truncated at 60
5	65	
6	66	
7	67	
8	59	truncated at 45
9	56	
10	55	
11	64	
12	68	
13	65	truncated at 55
14	64	
15	62	
16	61	
17	63	truncated at 58
18	60	
19	63	truncated at 62
20	57	
21	55	truncated at 49
22	53	
23	50	
24	49	truncated at 44
25	47	
26	71	truncated at 59
27	72	
28	75	
29	77	truncated at 69
30	62	

Table 1 Data sets simulated for model training

Data set	Actual failure time (time step)
31	71
32	56
33	58
34	63
35	62
36	67
37	68
38	46
39	58
40	77

Table 2 Data sets simulated for model testing

We compared the prediction results of the proposed Intelligent Product Limit Estimator (iPLE) with those of:

1. a similar neural network that treats truncation times as failure times (model A)
2. a similar neural network that excludes truncated data sets (model B).

The training target vectors for the *complete* training data sets are the same for all 3 models. The training target vectors for the *truncated* sets are different for the iPLE and model A and none for model C. For the iPLE, training target vectors were computed using the modified product limit survival estimation method as discussed previously. The standard Product Limit Estimation of survival probability for the entire training data sets is depicted in Figure 6. Model A, on the other hand, was trained based on the false assumption that truncation times were failure times. Model B represents prediction models which only use complete failure data and exclude truncated data.

Our simulation test consists of 3 assessments. In Assessment I, we used all 20 complete data sets and 10 truncated ones to train all three models. In Assessment II, we used only 10 complete training data sets and 10 truncated ones. In the last assessment, we only used the 10 truncated sets for training.

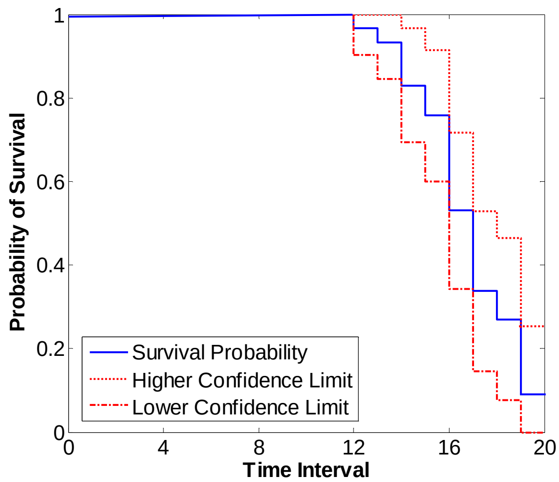

Figure 6 The standard Product Limit Estimation of survival probability for the entire training data sets

4. RESULTS AND DISCUSSION

A significant methodological issue is that of evaluating the trained model. As the prediction outputs of our model are represented by survival probabilities, there is no representation of the exact predicted failure times. Still, there is a well-defined goal: the accurate prediction of individual prognosis. We analysed each of the 10 prediction outputs individually. For evaluation purposes, we identify the predicted failure time merely by noting the first output unit that predicts a survival probability of less than *0.5*. We observed that the predicted survival probabilities closely match the actual failure times. Figure 7 shows one of the sample input data set (Dataset 40 where the actual failure time was simulated at *t=77*) inputted into the iPLE and Figure 7 shows its prediction results. The prediction output at each time step, Y_t, is arranged vertically, with first row value representing the probability that the bearing will survive the immediate next interval, second row value representing the probability that the bearing will survive the 2nd next interval, and so on.

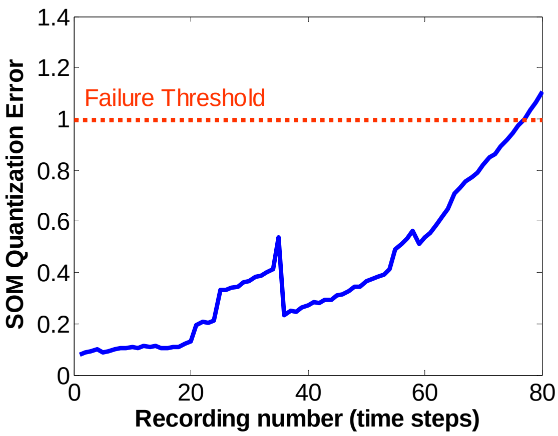

Figure 7 The indication of simulated bearing degradation condition and

Prediction at time step 1 ↓ | Prediction at time step 2 ↓ | Prediction at time step 3 ↓ | Prediction at time step 4 ↓

	Time Interval k=1			
	t=1	t=2	t=3	t=4
Surv. Prob. in next interval, $S_{t_{k+1}}$ →	1.00	1.00	1.00	1.00
Surv. Prob. in 2nd next interval, $S_{t_{k+2}}$ →	1.00	1.00	1.00	1.00
Surv. Prob. in 3rd next interval, $S_{t_{k+3}}$ →	1.00	1.00	1.00	1.00
Surv. Prob. in 4th next interval, $S_{t_{k+3}}$ →	1.00	1.00	1.00	1.00
Surv. Prob. in 5th next interval, $S_{t_{k+3}}$ →	1.00	1.00	1.00	1.00

......

Time Interval k=10			
t=37	t=38	t=39	t=40
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00

Time Interval k=11			
t=41	t=42	t=43	t=44
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
0.97	0.97	0.96	0.96

Time Interval k=12			
t=45	t=46	t=47	t=48
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
0.94	0.94	0.92	0.91

Time Interval k=13			
t=49	t=50	t=51	t=52
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	0.99	0.98
0.91	0.88	0.86	0.85

Time Interval k=14			
t=53	t=54	t=55	t=56
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
1.00	1.00	0.91	0.88
0.96	0.92	0.77	0.73
0.83	0.78	0.63	0.58

Time Interval k=15			
t=57	t=58	t=59	t=60
1.00	1.00	1.00	1.00
1.00	1.00	1.00	1.00
0.84	0.79	0.88	0.84
0.70	0.64	0.74	0.69
0.54	0.48	0.59	0.54

Time Interval k=16			
t=61	t=62	t=63	t=64
1.00	1.00	1.00	0.99
0.97	0.91	0.87	0.81
0.81	0.74	0.70	0.64
0.66	0.59	0.54	0.48
0.50	0.43	0.37	0.31

Time Interval k=17			
t=65	t=66	t=67	t=68
0.88	0.84	0.79	0.76
0.71	0.68	0.64	0.60
0.54	0.51	0.46	0.43
0.37	0.33	0.28	0.25
0.19	0.15	0.10	0.06

Time Interval k=18			
t=69	t=70	t=71	t=72
0.72	0.67	0.61	0.58
0.57	0.53	0.47	0.45
0.40	0.35	0.30	0.27
0.21	0.16	0.10	0.07
0.02	0.00	0.00	0.00

Time Interval k=19			
t=73	t=74	t=75	t=76
0.53	0.48	0.43	0.37
0.40	0.36	0.31	0.26
0.22	0.18	0.13	0.08
0.02	0.00	0.00	0.00
0.00	0.00	0.00	0.00

Time Interval k=20			
t=77	t=78	t=79	t=80
0.29	0.19	0.16	0.11
0.22	0.16	0.11	0.05
0.04	0.00	0.00	0.00
0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00

Figure 8 Results for the test using Data Set 40

As you can see in Figure 8, the survival probability was unity ($S_{t_{k+n}} = 1.00$) when the simulated bearing condition was at healthy state. The probability value began to drop at the end of the 11th interval when the bearing defect impulse started to rise. The survival probability first fell below 0.5 at the 5th column of interval 15's prediction output. This means that the bearing was expected to fail in the 5th following interval, i.e. interval 20. In interval

14, the first output unit that displays a probability of less than 0.5 is at the 4th column, which means the bearing was expected to fail in the 4th following interval, i.e. interval 20. This implication can be observed consistently in intervals 17, 18 and 19. The actual simulated failure is indeed in interval 20. Results of the tests using the other data sets (Datasets 31 to 39) produced similarly promising results.

As mentioned, the probability values of the output nodes can be combined to form an estimated survival curve for an *individual* bearing. Figure 9 depicts the survival curve represented by the output values of iPLE at *t=64*. Note that the the survival probability curve for Dataset 40 (Figure 8) is indeed different from the cumulative survival probability curve for the entire training bearing data sets (Figure 5). This observation proves that iPLE produces customised survival prediction for individual bearings by adapting the survival probabilities with more specific health condition data, rather than only fitting a statistical failure distribution (e.g. Weibull distribution) to failure instances history.

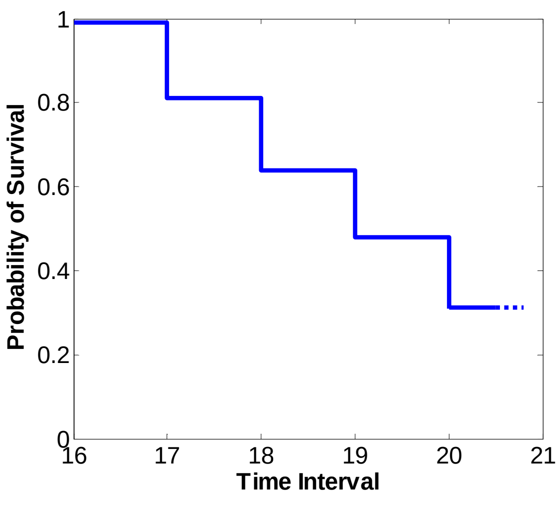

Figure 9 Results for the test using Data Set 40

To compare the effectiveness of the proposed iPLE with that of:

- a simlar neural network that treats suspension times as failure times and
- a similar neural network network that omits suspended data sets,

we calculated the accuracy of each model's prediction outputs. We obtain the prediction accuracy by subtracting the error ratio (the absolute error between the predicted failure interval (k_P) and the actual failure interval (k_A) divided by the actual failure interval) from unity.

$$Accuracy = (1 - \frac{|k_P - k_A|}{k_A}) \times 100\%$$

Figure 10 presents a bar graph of the average accuracy levels of the 3 prediction models in all 3 different assessments. (Note that we only plot the portion beyond 40% for comparison. This is because there is no death before the 12th interval in any of the training or testing datasets, the models generally produce a prediction output of *1.0* survival before the 12th interval (predicted failure interval, $k_P \geq 12$) where accuracy at that point is already at least 40% [$(1 - \frac{|12-20|}{20}) \times 100\%$].) The graph suggests that the iPLE provides more accurate outputs than those of the 2 control models in all assessments. Model B (which excludes truncation data) performs same as well as iPLE when sufficient sets of complete data are available. Its performance drops substantially when there are only a limited number of complete data sets (which is often the case in practice). When only truncated data are available for training, for example, in Assessment III, model B is totally incapable of performing prediction. Model A (which wrongly handles truncated data as failure data) is the weakest model among the 3 models.

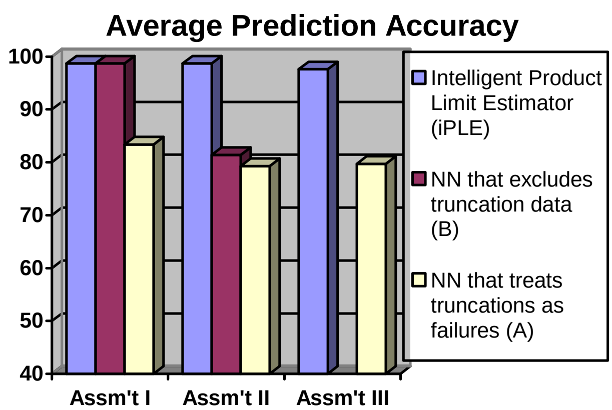

Figure 10 Prediction accuracy comparison between the iPLE and the 2 control models

Even though the simulation test results support our hypothesis that the iPLE can perform better than prognostics models that neglect truncated data, the difference between model A's accuracies in each assessment are smaller than we expected. This discrepancy between the expected and the actual validation results could be because the sample size is small (only 20 time intervals). For example, since the difference between the actual failures and truncations are usually only less than 3 time intervals, the prediction error of model A is also expected to be less than 3 time intervals. However, in real life situations where data may span numerous more intervals, model A's prediction error can be larger and its accuracy can drop to very low values. The other possible reason is that the simulated data are rather predictable. Even without much training, the models are already capable of estimating the time the degradation indicator will exceed the failure threshold. We anticipate seeing a larger difference between the performance of the iPLE and that of the 2 control models when our experimental bearing life test data become available.

5. CONCLUSIONS

The purpose of this study is threefold:

1. To illustrate the potential power of addressing the negligence of suspended lifetime data in machine component prognostics.
2. To enhance the output of a neural network by including survival probability estimation, in order to model, measure and manage risks.
3. To provide real-time long-range prediction, taking advantage of statistical models' ability to provide useful representation of survival probabilities, and of neural network's ability to recognise the nonlinear relationship between a machine component's future survival condition and a given series of prognostic data features.

iPLE is still in its infancy and we are continuing to improve its design and at the same time gather more real-life data for its verification. Despite possible shortcomings, this work still presents a compelling concept for long range machine component prognosis utilizing available information more fully and accurately, as well as for potentially stimulating other more robust variants of the proposed technique.

6. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support provided by the QUT Faculty of Built Environment & Engineering and the Cooperative Research Centre for Integrated Engineering Asset Management (CIEAM).

7. REFERENCES

Gebraeel, N., Lawley, M., Liu, R. and Parmeshwaran, V. (2004) *Residual life predictions from vibration-based degradation signals: a neural network approach*, IEEE Transactions on Industrial Electronics, 51(3), 694- 700.

Groer, P. G. (2000) In *Maintenance and Reliability Conference*, Vol. 2 Knoxville, Tennessee USA, pp. 59.01-59.04.

Heng, A., Tan, A. C. C., Mathew, J. and Kim, Y.-H. (2006) In *International Tribology Conference*Brisbane, Australia.

Huang, R., Xi, L., Li, X., Richard Liu, C., Qiu, H. and Lee, J. *Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods*, Mechanical Systems and Signal Processing, In Press, Corrected Proof(
Jantunen, E. (2004) In *VETOMAC-3 and ACSIM-2004*New Delhi, India, pp. 836-846.

Kaplan, E. L. and Meier, P. (1958) *Nonparametric estimation from incomplete observations*, Journal of the American Statistical Association, 53(457-481.

Li, Y., Kurfess, T. R. and Liang, S. Y. (2000) *Stochastic prognostics for rolling element bearings*, Mechanical Systems and Signal Processing, 14(5), 747-762.

Qiu, H., Lee, J., Lin, J. and Yu, G. (2003) *Robust performance degradation assessment methods for enhanced rolling element bearing prognostics*, Advanced Engineering Informatics: Intelligent Maintenance Systems, 17(3-4), 127-140.

Qiu, J., Set, B. B., Liang, S. Y. and Zhang, C. (2002) *Damage mechanics approach for bearing lifetime prognotics*, Mechanical Systems and Signal Processing, 16(5), 817-829.

Raytek (2004).

Roemer, M. J., Byington, C. S., Kacprzynski, G. J. and Vachtsevanos, G. (2005) In *ISHEM Forum*Napa Valley, CA.

Schömig, A. and Rose., O. (2003) In *Industrial Engineering Research Conference*Portland, Oregon.

Sutherland, H., Repoff, T., House, M. and Flickinger, G. (2002) In *IEEE Aerospace Conference*, Vol. 7-3131, pp. 1-6.

Vlcek, B. L., Hendricks, R. C. and Zaretsky, E. V. (2003) NASA Center for AeroSpace Information, Hanover, MD, pp. 22.

Vlok, P.-J., Wnek, M. and Zygmunt, M. (2004) *Utilising statistical residual life estimates of bearings to quantify the influence of preventive maintenance actions*, Mechanical Systems and Signal Processing, 18(4), 833-847.

Wang, P. and Vachtsevanos, G. (2001) *Fault prognostics using dynamic wavelet neural networks*, AI EDAM, 15(349-365.