



**HAL**  
open science

## Predicting component reliability and level of degradation with complex-valued neural networks

Olga Fink, Enrico Zio, Ulrich Weidmann

► **To cite this version:**

Olga Fink, Enrico Zio, Ulrich Weidmann. Predicting component reliability and level of degradation with complex-valued neural networks. Reliability Engineering and System Safety, 2014, 121, pp.198-206. 10.1016/j.ress.2013.08.004 . hal-00879431

**HAL Id: hal-00879431**

**<https://centralesupelec.hal.science/hal-00879431>**

Submitted on 3 Nov 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Predicting component reliability and level of degradation with complex-valued neural networks

Olga Fink<sup>a,\*</sup>, Enrico Zio<sup>b</sup>, Ulrich Weidmann<sup>a</sup>

<sup>a</sup>*Institute for Transport Planning and Systems, ETH Zurich, Zurich, Switzerland*

<sup>b</sup>*Chair on Systems Science and the Energetic Challenge, European Foundation for New Energy-Electricité de France (EDF) at École Centrale Paris and SUPELEC, France;  
Department of Energy, Politecnico di Milano, Italy*

---

## Abstract

In this paper, multilayer feedforward neural networks based on multi-valued neurons (MLMVN), a specific type of complex valued neural networks, are proposed to be applied to reliability and degradation prediction problems, formulated as time series. MLMVN have demonstrated their ability to extract complex dynamic patterns from time series data for mid- and long-term predictions in several applications and benchmark studies. To the authors' knowledge, it is the first time that MLMVN are applied for reliability and degradation prediction.

MLMVN is applied to a case study of predicting the level of degradation of railway track turnouts using real data. The performance of the algorithms is first evaluated using benchmark study data. The results obtained in the reliability prediction study of the benchmark data show that MLMVN outperforms other machine learning algorithms in terms of prediction precision and is also able to perform multi-step ahead predictions, as opposed to the previously best performing benchmark studies which only performed up to two-step ahead predictions. For the railway turnout application, MLMVN confirm the good performance in the long-term prediction of degradation and does not show accumulating errors for multi-step ahead predictions.

*Keywords:* Neural networks, Complex valued neural networks, Reliability prediction, Level of degradation, Railway turnout system

---

\*Corresponding author

*Email addresses:* ofink@ethz.ch (Olga Fink), enrico.zio@ecp.fr (Enrico Zio), weidmann@ivt.baug.ethz.ch (Ulrich Weidmann)

---

## 1. Introduction

Railways have very high system reliability requirements due to their physical and service characteristics. Key physical characteristics include being track bound and having complex safety systems. Key service characteristics include high customer expectations and operating at high capacity (e.g. very frequent service). The requirements for system reliability and availability are particularly demanding in network segments that are operated close to their capacity limit. These high capacity utilization rates amplify a bottleneck character of these network segments as they cannot be substituted by other segments. When failures occur in a bottleneck they can only partly be compensated for by buffer capacity in the schedule. Therefore, they have a direct impact on network capacity. Anticipating and preventing failures in such crucial links increases railway network stability and service productivity. A good example of a railway network bottleneck is a tunnel on heavily utilised corridors.

Mid- and long-term predictions are essential for systems with tight operating schedules and long lead times for spare parts and/or maintenance scheduling, such as for example railway infrastructure systems with bottleneck character. For such systems, a sufficiently long “anticipation period”, which is the time needed to take the actions necessary to prevent the disruptive event (i.e., the amount of advance warning), is often required. The minimum anticipation period is determined with respect to the criticality, the operating schedule, the lead times needed to obtain spare parts and the spare part policy of the pertinent system [59]. For this reason, advanced system reliability prediction and prognostics methods are needed to provide flexible, precise and reliable predictions of failure and degradation behaviour not only in the short term but also in the mid and long term.

As more data describing the system condition and its influencing parameters, become available, data-based methods are being increasingly applied to predict system behaviour and the reliability of a system. Artificial neural networks and support vector machines are some of the most widely applied data-based algorithms in this field.

Artificial neural networks are a generic term subsuming miscellaneous types of self-adaptive algorithms imitating the learning principles of biological neural networks. The application fields of neural networks can be catego-

alized with respect to different criteria, such as industrial application, type of reliability problem, life cycle phase in which the algorithm is predominantly applicable and the type of learning problem.

Because neural networks are a data-based method they are universally applicable to systems from different industrial application fields for system reliability prediction, diagnostics and prognostics. They have been applied among others for applications in nuclear power plants [42], power distribution [54], for different industrial applications of motor bearings [37], [60], [33] and electric machines [50]. The existing research in the field of railway systems focuses among others on condition monitoring, fault detection and fault diagnostics [46], [16], [32], [19].

Neural networks can be, generally, applied to classical reliability problems, such as estimating the failure rate [5], time-to-failure or cumulative time-to-failure [12]. These predictions, then, are valid population wide. For these types of predictions, neural networks are mainly selected due to their non-linearity, self-adaption and the precision of their predictions. However, they can be substituted by other statistical or regression methods.

With the increased availability of monitoring data on the condition of a specific system, neural networks are also increasingly applied in the field of fault detection [53], fault diagnostics [51] and for predicting the residual useful life [55]. All of these analyses and predictions are performed on the level of an individual system or component, as opposed to those valid on population level. In this field, neural networks are mainly competing with other machine learning techniques, such as support vector machines [44], [30], k-nearest neighbors [6] and several other algorithms.

In addition to the approaches in which neural networks are directly applied to reliability problems, they are also applied within meta or surrogate models in which the analyses based on physical models are computationally expensive [26], [20], [45] so that neural networks learn to generalize the input-output mapping from few samples computed by the physical models. Furthermore, neural networks can also be applied in combination with statistical methods, e.g. for estimating parameters of distributions [1], and in combination with model-based methods, e.g. to accelerate optimization computations for very complex reliability block diagrams [48], or to determine the transition probability in homogenous Markov processes [49].

Considering the life cycle phase in which neural networks are predominantly applicable, it appears obvious that they are particularly applicable in the operation phase as reliability monitoring data is recorded in opera-

tion and can be used to predict average or specific behaviour of systems. Therefore, most of the studies using neural networks refer to the operating phase within the life cycle of a product [15], [27]. However, neural networks have also been applied in the design phase to predict the reliability performance based on specific system characteristics, on performed tests or on the performance of previous system designs [36], [35].

With respect to the type of the learning problem, the four major application types are approximation, categorization, association and optimization. Approximation and categorization can be further subdivided in two subcategories. Approximation is distinguished in static and dynamic regression. Both problem types are relevant for reliability prediction problems. Static regression is applied especially for causal relationships [36], while dynamic regression is characterized by a feedback connection of the output to the input of the model [41]. Categorization is distinguished in classification and clustering. In classification, a new data set is assigned to a subpopulation or a class. Classification is especially applied in diagnostics to determine the specific cause of an underlying fault [51]. In clustering, subsets or clusters are identified within the data space based on their similarities within the defined input parameters. Clustering can be applied in the field of fault detection [57], [54], [9] but also to distinguish between the different fault patterns [17]. Association is especially applied to memorization tasks, such as by applying Hopfield networks [7], while optimization is a field of application for which most of the neural network types have not been specifically designed. However, neural networks can also be applied to optimize system design by learning the optimal structure of the network [39], [29].

There are several types of neural networks, showing different characteristics and different performance. Some examples of neural networks applied in the field of reliability prediction, and diagnostics and prognostics are multilayer perceptrons (MLP), recurrent neural networks [25], self-organizing maps (SOM) [31], deep belief networks (DBN) [52] and several other algorithms. Besides the applications in which neural networks are used as stand-alone algorithms there have also been several approaches to combine neural networks with other soft-computing methods, such as for example fuzzy-set theory [37], [31].

This research study advances the set of neural network types applied for reliability and degradation prediction by applying a powerful method that, to the authors' knowledge, has not yet been applied in this field. Multilayer feedforward networks based on multi-valued neurons (MLMVN), a type of

complex-valued neural networks, have proven to be particularly very good in mid- and long-term time series predictions [4]. Furthermore, multi-valued neurons and MLMVN they have also been successfully applied to associative memory tasks, classification tasks, to signal processing or signal generation [4].

MLMVN have shown a good generalization ability for time series predictions and have shown a very good performance in extracting underlying patterns that could be transferred to value ranges that were not covered by the training data. Additionally, MLMVN do not show an accumulation of errors in multi-step ahead predictions, which, typically, occurs when using predicted values as input for subsequent predictions. MLMVN have been tested on several benchmark studies. In these studies, MLMVN outperformed other machine learning techniques in classification tasks as well as in time series prediction [2],[3]. MLMVN outperformed even some fairly sophisticated methods, such as adaptive-network-based fuzzy inference systems [2]. MLMVN can provide more precise results than other machine learning algorithms because they overcome some of the drawbacks of other methods, such as obtaining locally optimal solutions and poor generalization ability.

The focus of this paper are time series prediction for reliability and degradation problems. Using the prediction of the degradation paths and the defined fault tolerance limits, remaining useful life (RUL) can be estimated [56]. Predicting the remaining useful life based on the degradation prediction is an essential element of the health management of complex systems [56]. For predicting dynamic degradations several approaches have been proposed. Physical models and filtering approaches, such as Kalman filters [56] and Particle filtering [21], [28] belong to state-of-the-art methods applied in degradation modelling. There are some similarities between time series regression and filtering. However, while the goal of regression is to determine a mapping between the input and output, based on a finite number of observed input-output mappings, the goal of filtering is to make sequentially an inference about a dynamic system, based on the evolution of the state in time and a model relating the noisy measurements to the state [14]. Furthermore, several stochastic processes, such as gamma processes [8], [47] have been proposed for predicting degradation processes.

Yet, pure time series predictions are less commonly applied for predicting dynamic degradations, especially due to the accumulating error and the associated increasing uncertainty, when the current prediction is based on previous predictions [10]. If the uncertainty in long-term time series pre-

dictions can be limited, time series predictions can be a powerful tool for predicting degradation paths requiring only the lagged observed measurements, respectively additional influencing parameters. Given the ability of MLMVN algorithms to detect long-term trends and developments without showing a tendency to accumulate errors in multi-step ahead predictions, MLMVN are potentially applicable for long-term multi-step ahead degradation predictions. Furthermore, these long-term degradation predictions could be potentially used for predicting the remaining useful life. This would further complement the existing methods applied in this field [28], [23], [18], [47].

Starting from the strong performance of the MLMVN algorithm in mid- and long-term time series prediction [4], the MLMVN are selected to be applied to the multi-step ahead prediction of the degradation of a railway turnout system. The prediction is based solely on lagged values of the time series, without considering any additional influencing parameters. However, before it is applied to the case study the performance of the MLMVN algorithm is tested on a benchmark study from the field of reliability prediction [58] using two different approaches to compare it to the performance of other algorithms applied to similar types of problems.

The remainder of the paper is organized as follows. The next Section of this paper presents an introduction to complex-valued neural networks and their application. Section 3 describes the application of MLMVN to the reliability benchmark study dataset based on two different approaches. Section 4 describes the application of the MLMVN algorithm to predict the degradation of a railway turnout system. Finally, Section 5 presents the conclusions of this research.

## 2. Complex-valued neural networks

Similarly to biological neural networks, artificial neural networks are made up of neurons. The functionality and limitations of these single neurons is determined mainly by their activation function. Activation functions include threshold functions, sigmoid activation functions and several other functions [22]. In general, the functionality of single neurons is quite limited. However, it can be significantly increased by using complex-valued activation functions (as opposed to the real-valued activation functions).

Multi-valued neurons are a special type of complex-valued neurons. They have a threshold function of multi-valued logic and complex-valued weights.

The inputs and outputs of multi-valued neurons lie on the unit circle ( $k^{th}$  roots of unity which corresponds to  $\varepsilon_k = e^{\frac{i2\pi}{k}}$ , whereby  $i$  is a complex number) [4]. The mapping is displayed in Figure 1, where the complex plane is divided into  $k$  equal sections. The multi-valued logic inherent in multi-valued neurons means that they are able to learn problems which are non-linearly separable and which cannot be learned by single real-valued neurons. The XOR problem is an example of a non-linearly separable problem that cannot be learned by a single real-valued neuron but a single MVN can learn to separate this non-linearly separable problem [4].

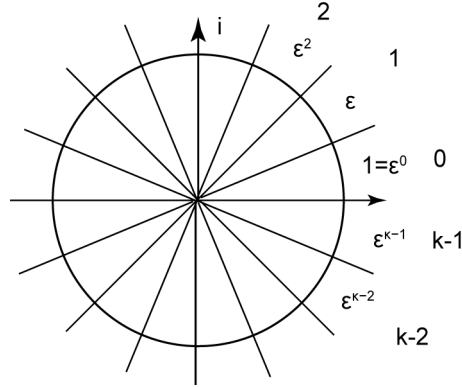


Figure 1: Multi-valued activation function on the unit circle

The learning algorithm of a single multi-valued neuron (MVN) corresponds to the movement along the unit circle. Therefore, the learning algorithm cannot suffer from local minima, which is a major drawback of back-propagation learning algorithms [4]. The learning rule is derivative-free and is based on a simple linear error correction rule (Equation 1).

$$W_{r+1} = W_r + \frac{C_r}{(n+1)|z_r|} (\varepsilon^q - \varepsilon^s) \bar{X}, \quad (1)$$

where  $W$  is the weighting vector,  $r$  is the current iteration,  $r+1$  is the subsequent iteration,  $C_r$  is the learning rate,  $X$  is the input vector and  $\bar{X}$  is a complex conjugated to  $X$ ,  $z$  is the weighted sum:  $z = w_0 + w_1x_1 + \dots + w_nx_n$ ,  $\frac{1}{|z|}$  is the inverse absolute value of the current weighted sum,  $\varepsilon^q$  is the desired output and  $\varepsilon^s$  is the actual output.  $\frac{1}{|z|}$  acts as an adaptable learning rate for datasets with highly non-linear input/output mappings [4]. The learning rule can also be used without this extension.



In general, multi-valued neurons have the advantages of increased functionality, improved performance and reduced computation time due to their activation function and the simplicity of their learning algorithm [4].

While single MVN are able to solve problems that are not solvable by other types of neurons, their functionality can be increased when they are integrated into more complex layered structures. These layered structures are called multilayer networks based on multi-valued neurons (MLMVN), and are feedforward neural networks. The learning algorithms in layered network structures are usually based on derivatives of the activation function. But because the activation functions of MVN are not differentiable, the back-propagation learning rule similar to that of a single MVN is applied for the learning process, which is likewise derivative free [2].

In a layered network structure the following learning rules are applied [4], whereby the learning rules are distinguished between those applied in the output layer, those applied in the first hidden layer and those applied in the hidden layers  $2 - (m-1)$ .

For the output layer neurons:

$$\begin{aligned}\tilde{w}_0^{km} &= w_0^{km} + \frac{C_{km}}{(N_{m-1} + 1)} \delta_{km}; \\ \tilde{w}_i^{km} &= w_i^{km} + \frac{C_{km}}{(N_{m-1} + 1)} \delta_{km} \bar{Y}_{i,m-1}; i = 1, \dots, N_{m-1}\end{aligned}\quad (2)$$

For the neurons from the hidden layers  $2, \dots, m - 1$ :

$$\begin{aligned}\tilde{w}_0^{kj} &= w_0^{kj} + \frac{C_{kj}}{(N_{j-1} + 1)} \delta_{km}; \\ \tilde{w}_i^{kj} &= w_i^{kj} + \frac{C_{kj}}{(N_{j-1} + 1) |z_{kj}|} \delta_{kj} \bar{Y}_{i,j-1}; \\ i &= 1, \dots, N_{j-1}; j = 2, \dots, m - 1\end{aligned}\quad (3)$$

For the 1<sup>st</sup> hidden layer neurons:

$$\begin{aligned}\tilde{w}_0^{k1} &= w_0^{k1} + \frac{C_{k1}}{(n + 1)} \delta_{k1}; \\ \tilde{w}_i^{k1} &= w_i^{k1} + \frac{C_{k1}}{(n + 1) |z_{k1}|} \delta_{k1} \bar{x}_i; i = 1, \dots, n,\end{aligned}\quad (4)$$

where  $\bar{Y}_{i,j-1}$  is the updated output of the  $i$ th neuron from the  $(j - 1)^{st}$  layer with  $j = 1, \dots, m - 1$ , that is the  $i$ th input of the  $kj$ th neuron, the bar sign

stands for the complex conjugation;  $z_{kj}$  is the current value of the weighted sum of the  $kj$ th neuron ( $k$ th neuron in the  $j$ th layer);  $x_1, \dots, x_n$  are the network inputs; and  $C_{kj}$  is a learning rate of the  $kj$ th neuron. The learning rate is usually set to 1 for the single neurons.

Within learning rule of the output layer  $\delta_{km}$  is defined by:

$$\delta_{km} = \frac{1}{s_m}(D_{km} - Y_{km}); k = 1, \dots, N_m, \quad (5)$$

where  $D_{km}$  is the desired output of the  $k$ th neuron from  $m$ th layer, which is the output layer,  $Y_{km}$  is the actual output of the  $k$ th neuron from  $m$ th layer,  $s_j$  is the number of neurons in the  $j - 1^{st}$  layer, the preceding layer of the  $j$ th layer ( $s_j = N_{j-1} + 1$ ).

Within learning rule of the  $j$ th layer  $\delta_{kj}$  is defined by:

$$\delta_{kj} = \frac{1}{s_j} \sum_{i=1}^{N_{j+1}} \delta_{i,j+1} (w_k^{i,j+1})^{-1}; k = 1, \dots, N_j; j = 1, \dots, m - 1, \quad (6)$$

whereby  $kj$  is the  $k$ th neuron of the  $j$ th layer.

The learning is performed iteratively, similarly to standard MLP learning procedure, until an error of 0 or the defined error criterion is achieved. The error is usually measured as root mean square error (RMSE) or normalized root mean square error (NRMSE).

The main benefits of complex-valued neural networks are particularly apparent when they are applied to signal processing or signal generation. Signals are naturally located in the frequency domain and therefore can be more easily processed with complex-valued numbers. However, time series data can also be easily transformed into numbers located on the unit circle as shown in Equation 7, whereby  $y_0, y_1, \dots, y_n$  is the original time series and  $x_j$  represents the complex number located on the unit circle. In order to prevent close neighboring of minimal and maximal values of the time series on the unit circle the value range is extended below the minimum and above the maximum observed values, which results in the value range  $[a, b]$  [4]:

$$\begin{aligned} y_j \in [a, b] &\Rightarrow \varphi_j = \frac{y_j - a}{b - a} 2\Pi \in [0, 2\Pi]; \\ x_j &= e^{i\varphi_j}; \\ j &= 0, \dots, n \end{aligned} \quad (7)$$

To set up a MLMVN, the structure of the specific network has to be selected, which is a similar process to that of setting up MLP algorithms. For time series predictions, Aizenberg [4] recommends applying a two-layered neural network with a small number of neurons in the first layer and large number of neurons in the second hidden layer. By so doing, in the first step the input space is condensed to a low dimension that is subsequently expanded significantly in the second layer. The second layer learns especially to discover and to predict specific jumps within the time series, while the first layer learns the average time evolution [4].

The algorithm performance is influenced by both the network structure and the number of lagged time series values used as input to the algorithms. For long-term predictions, the number of lagged time series values should be sufficiently long to enable the prediction algorithm to extract different underlying patterns in the time series and learn the different implicit underlying trends useful for the prediction. Additionally to the structure of the network, the desired precision of the prediction has to be defined, which determines the stopping point of the training process.

### 3. Reliability benchmark study

#### 3.1. Dataset and previous studies

The focus of this study is on mid- and long-term time series predictions. The selected case study of measured degradation values of the railway turnout system shows a gradual development as it is based on an underlying accumulating damage process. For these gradually evolving time series, long-term multi-step ahead predictions are essential. Short-term predictions will not be valuable in this context due to the long cycles and a comparably small degradation progression rate which induces only small changes between two time steps. MLMVN have shown a particularly good performance on long-term time series predictions being able to detect the relevant underlying evolving long-term trends and processes. Even though MLMVN have already been tested on several benchmark studies, both in classification as well as in time series prediction, before applying the algorithm on the railway case study, it is tested on a reliability prediction benchmark dataset.

The benchmark dataset was first introduced by Xu et al. [58] and has been used to evaluate the performance of numerous algorithms ([13], [61], etc.). The benchmark study by Xu et al. [58] includes two datasets for reliability evaluations with neural networks: the first dataset consists of an

ordered series of time-to-failure (TTF) values of a turbocharger system and corresponding computed reliability values; the second dataset is made of a series of miles between failures. The first time series shows a gradual development, whereas the second dataset is determined by very short fluctuating cycles. Because the apparent strength of MLMVN is to detect long-term trends and underlying processes, which is also the focus of the selected case study, the time series of miles between failures is considered not to be suitable to demonstrate the long-term trends and relationships due to its short-term fluctuating character. Even though the considered case study also shows an oscillating behaviour, this oscillation evolves over a long period and is not comparable to that of the miles-between-failures time series.

The reliability benchmark dataset, however, is considered more suitable to demonstrate the ability of the MLMVN for multi-step ahead predictions. In this dataset, the characteristic behaviour of the time series also evolves over several time steps, so that multi-step ahead predictions are more suitable for this time series. However, the authors are aware that the selected time series does not show all the characteristics required to perform a benchmark study that is truly comparable to the selected case study. First of all, due to the short length of the reliability time series, it is not possible to show a true long-term prediction. However, a multi-step ahead prediction is performed, which has not been performed by other studies using this benchmark dataset. Secondly, the reliability time series does not show any oscillating behaviour which is more difficult to predict and which is observed in the selected case study.

The dataset used (Table 1) consists of 40 TTF data points and corresponding reliability values computed by Equation 8:

$$R(T_i) = 1 - \frac{i - 0.3}{n + 0.4} \quad (8)$$

Two approaches have been used in other research studies to make reliability predictions from this dataset. In the first approach, only the series of computed reliability values is predicted ([58], [11], [40], [13], [24], [34]). Xu et al. [58] also included the time-to-failure values as an explanatory variable but when they did so, the results became worse and this approach was not further pursued. In the second approach, Zio et al. [61], Moura et al. [38] and Rocco S [43] predicted the time series of time-to-failure data and subsequently used the predicted values to calculate the reliability values using Equation 8.

Table 1: Turbocharger time-to-failure and reliability data from [58]

Failure order (t)	Time-to-failure $T_t/1000h$	Non-parametric estimate $R(T_t)$	Failure order (t)	Time-to-failure $T_t/1000h$	Non-parametric estimate $R(T_t)$
1	1.6	0.9930	21	6.5	0.7938
2	2	0.9831	22	6.7	0.7839
3	2.6	0.9731	23	7	0.7739
4	3	0.9631	24	7.1	0.7639
5	3.5	0.9532	25	7.3	0.7540
6	3.9	0.9432	26	7.3	0.7440
7	4.5	0.9333	27	7.3	0.7341
8	4.6	0.9233	28	7.7	0.7241
9	4.8	0.9133	29	7.7	0.7141
10	5	0.9034	30	7.8	0.7042
11	5.1	0.8934	31	7.9	0.6942
12	5.3	0.8835	32	8	0.6843
13	5.4	0.8735	33	8.1	0.6743
14	5.6	0.8635	34	8.3	0.6643
15	5.8	0.8536	35	8.4	0.6544
16	6	0.8436	36	8.4	0.6444
17	6	0.8337	37	8.5	0.6345
18	6.1	0.8237	38	8.7	0.6245
19	6.3	0.8137	39	8.8	0.6145
20	6.5	0.8038	40	9	0.6046

In both approaches, lagged data from the original dataset were used for predicting the next step (or the next two steps): predictions at the previous steps were not used as input to the next prediction step. Xu et al. [58] justified this by saying that multi-step ahead predictions would lead to accumulated errors.

In our research, we predicted first the pure reliability values, as in the analyses performed by Xu et al. [58]. In the second step, the approach of Zio et al. [61] was followed in that first the TTF time series was predicted and subsequently the reliability values were calculated. However, in contrast to the previous studies (which applied only one- or two-step ahead predictions), this study made a multi-step ahead prediction. This is done by using the output of a one-step ahead prediction as the input for the subsequent prediction in an iterative process. The principle is shown in Equation 9, whereby  $\hat{x}$  represents the predicted values:

$$\begin{aligned}
 \hat{x}_{t+1} &= \hat{f}(x_{r-n+1}, \dots, x_r), \\
 \hat{x}_{t+2} &= \hat{f}(x_{r-n+2}, \dots, x_r, \hat{x}_{t+1}), \\
 \hat{x}_{t+3} &= \hat{f}(x_{r-n+3}, \dots, x_r, \hat{x}_{t+1}, \hat{x}_{t+2}), \\
 &\dots
 \end{aligned}
 \tag{9}$$

The normalized root mean squared error (NRMSE) (Equation 10) was

used to compare results from MLMVN to those from other studies:

$$NRMSE = \sqrt{\frac{\sum (x_t - \hat{x}_t)^2}{\sum x_t^2}} \quad (10)$$

### 3.2. Predicting the reliability time series

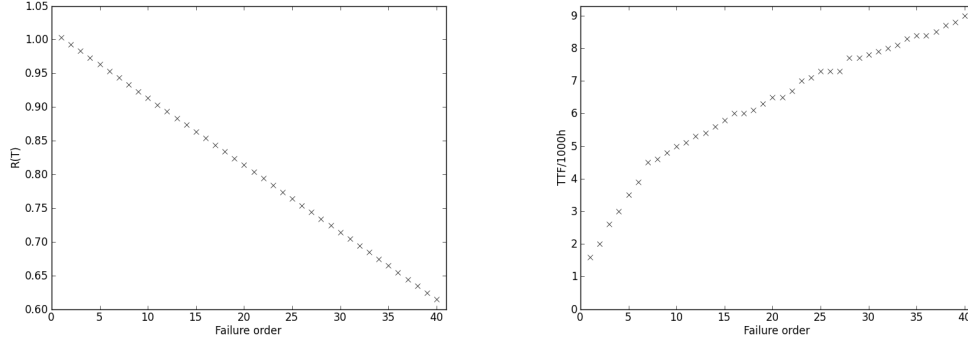
In the first approach, only the time series of the calculated reliability values was considered. The MLMVN learned the patterns in the time series using the training dataset. The trained network was then used to predict the last five values of the time series in an iterative process. The prediction was always performed one step ahead and the predicted value was subsequently used as part of the input for the prediction of the next step. The iterative prediction approach is presented in Equations 11, where  $\hat{R}$  represents the predicted values:

$$\begin{aligned} \hat{R}_{36} &= \hat{f}(R_{26}, \dots, R_{35}), \\ \hat{R}_{37} &= \hat{f}(R_{27}, \dots, R_{35}, \hat{R}_{36}), \\ &\dots \\ \hat{R}_{40} &= \hat{f}(R_{30}, \dots, R_{34}, \hat{R}_{36}, \hat{R}_{37}, \hat{R}_{38}, \hat{R}_{39}) \end{aligned} \quad (11)$$

The input data were transformed according to Equation 7. In this study, the input length was set to 10 and the network architecture was set to 10-2-100-1, but no particular effort was made to optimize it, neither the input length nor the number of neurons in the hidden layer, and it can be probably simplified.

The MLMVN algorithm was able to predict the reliability of the last five data points ( $R_{36}$ - $R_{40}$ ) precisely, with zero error. This result is not surprising because of the function linearity, since only the reliability values were considered in the time series (time-to-failure data was not used in the prediction). Indeed, as shown in Figure 2, the relationship becomes purely linear without any noise in the data. However, if the time-to-failure values are predicted first and then used to calculate the reliability values using the Equation 8, the prediction task is no longer linear (Figure 2). This approach corresponds to the second approach followed by Zio et al. [61].

Table 2 compares the results of this research with those of other studies. In case that several algorithms have been compared in the single papers, only



(a) Predicting reliability time series: purely linear relationship      (b) Predicting TTF time series: non-linear relationship

Figure 2: Different approaches to prediction

the best performing algorithm is included in the overview. It is interesting to note that while the best performing algorithm from these other studies (a particle swarm optimized support vector machine) performed relatively well (NRMSE of 0.0002 [34]), none of the algorithms applied in the cited studies detected the linear function of the input signals, which would correspond to a zero-error.

Table 2: Comparison of predicted reliability with different machine learning algorithms

	MLMVN	PSO+SVM <sup>a</sup> [34]	GA-SVR <sup>b</sup> [13]	SVMG <sup>c</sup> [40]	NF <sup>d</sup> [11]	RBF <sup>e</sup> [58]	ER <sup>f</sup> [24]
NRMSE	0	0.0002	0.0004	0.00312	0.00369	0.00391	0.00187

<sup>a</sup>PSO+SVM: Particle swarm optimized support vector machine

<sup>b</sup>GA-SVR: Support vector regression with genetic algorithms

<sup>c</sup>SVMG: Support vector machines with genetic algorithms

<sup>d</sup>NF: Neural fuzzy network

<sup>e</sup>RBF: Radial basis function

<sup>f</sup>ER: Evidential reasoning with nonlinear optimization

### 3.3. Predicting the time-to-failure time series

After using the MLMVN to predict the reliability data series, we tested its ability to predict time-to-failure time series. The MLMVN was trained

using the training dataset and then applied to predict the last five values of the time series in an iterative process, always predicting one-step-ahead values and using those as input for the next step prediction. The iterative prediction approach used is shown in Equations 12, where  $\widehat{TTF}$  represents the predicted values:

$$\begin{aligned}
 \widehat{TTF}_{36} &= \hat{f}(TTF_{16}, \dots, TTF_{35}), \\
 \widehat{TTF}_{37} &= \hat{f}(TTF_{17}, \dots, TTF_{35}, \widehat{TTF}_{36}), \\
 &\dots \\
 \widehat{TTF}_{40} &= \hat{f}(TTF_{20}, \dots, TTF_{35}, \widehat{TTF}_{36}, \widehat{TTF}_{37}, \widehat{TTF}_{38}, \widehat{TTF}_{39})
 \end{aligned} \tag{12}$$

The principles used to design the network structure were similar to those described in the previous Section: the size of the first layer was set to reduce the dimensionality and the size of the second layer was set to expand into a high dimensional space, following the recommendation of Aizenberg in [4]. Specifically, the network architecture was set to 20-2-12000-1. For this time series, several input lengths have been evaluated and the best performing input length was selected. The input data were transformed according to Equation 7.

The prediction results of the TTF values are displayed in Table 3. The results show that four out of five values are predicted precisely by the MLMVN algorithm ( $\widehat{TTF}_{36}, \widehat{TTF}_{38}, \widehat{TTF}_{39}, \widehat{TTF}_{40}$ ).

Table 3: Predicted TTF values with MLMVN

No.	Time-to-failure (actual) $T_t/1000h$	Time-to-failure (predicted) $T_t/1000h$
36	8.4	8.4
37	8.5	8.6
38	8.7	8.7
39	8.8	8.8
40	9.0	9.0

The NRMSE for the consecutive predictions of the last five time-to-failure values in the dataset ( $\widehat{TTF}_{36}-\widehat{TTF}_{40}$ ) is 0.0051. As shown in Table 4, the prediction precision of the MLMVN algorithm exceeds the performance of the singular spectrum analysis (SSA) [43] and also the performance of the infinite



impulse response locally recurrent neural network (IIR-LRNN) algorithm, both for the one-step and two-step ahead predictions [61]. It should also be noted that the SSA algorithm only predicts four values and the IIR-LRNN algorithm four (one step ahead) respectively three (two step ahead) values as compared to the MLMVN algorithm which predicts five values in a multi-step ahead approach. The MLMVN algorithm also outperforms the support vector regression (SVR) algorithm for one-step ahead predictions [38] (which again only predicts four values one step ahead).

Table 4: Comparison of predicted time-to-failure with different machine learning algorithms

	MLMVN (five values) <sup>d</sup>	SSA <sup>a</sup> (four values) <sup>e</sup> [43]	SVR <sup>b</sup> (four values) <sup>e</sup> [38]	SVR (three values) <sup>f</sup> [38]	IIR-LRNN <sup>c</sup> (four values) <sup>e</sup> [61]	IIR-LRNN (three values) <sup>f</sup> [61]
NRMSE	0.0051	0.0053	0.0055	0.0024	0.0149	0.0199

<sup>a</sup>SSA: Singular spectrum analysis

<sup>b</sup>SVR: Support vector regression

<sup>c</sup>IIR-LRNN: Infinite impulse response locally recurrent neural networks

<sup>d</sup>Five steps ahead

<sup>e</sup>One step ahead

<sup>f</sup>Two steps ahead

If we compare the performance of MLMVN on the five predicted TTF values to the performance of the two-step ahead prediction of the SVR algorithm for the three predicted TTF values, the comparison makes the MLMVN appear to have a worse prediction precision. However, comparing the performance of MLMVN for the same values as predicted by the SVR algorithm, namely the last three values of the time series ( $\widehat{TTF}_{38}$ - $\widehat{TTF}_{40}$ ), shows that MLMVN were able to predict these values exactly. This would result in a NRMSE value of 0, compared to a NRMSE value of 0.0024 for the SVR algorithm in [38].

### 3.4. Conclusions on the benchmark studies

The results of both tests on the benchmark dataset (reliability and time-to-failure time series) show that MLMVN algorithms provide good results for time-series-based predictions of reliability compared to other machine learning algorithms. On the reliability benchmark data set, MLMVN were able to deduce the purely linear relationship from the lagged time series data, which previously applied algorithms were capable of. MLMVN were able to

predict subsequent values with zero error. On the time-to-failure benchmark data set, the performance of MLMVN exceeds those of other algorithms and shows better performance for a longer prediction period in a multi-step ahead prediction.

## 4. Case study: predicting railway turnout system degradation

### 4.1. *Railway turnouts in network segments with high capacity utilization rates*

The most defining characteristic of a railway is that it is a track bound system. The components that enable trains to be guided from one track to another are called turnouts ("set of points" in British English and "switches" in American English). Turnouts are also safety critical components since their failure can lead to a derailment. Due to their criticality, both in terms of safety and system capacity, turnouts are subject to tightly scheduled inspection and preventive maintenance.

As with all physical infrastructure systems, turnout systems are subject to degradation. The turnout degradation process depends on several parameters, such as axle loads, train speed, condition of the train wheels, environmental conditions etc. The larger the positioning forces applied and amount of work performed, the higher the degradation. The goal of this research was to predict the turnout degradation level based on time series data of force measurements derived from a monitoring system. In this research study, it was assumed that the influence of all the parameters is reflected in the time series and that the algorithm is able to capture the systematic influences and also the periodicity of their occurrence. The study purpose was not to analyse the influencing parameters but only to predict the time series of the performed work.

### 4.2. *Data applied in the case study*

The data used in this research was collected from six force measurement devices installed along the turnout blades and the frog. The force measurement is activated when the positioning process starts and the system records the forces applied for each positioning. The measurement system also computes the work performed by the actuator system, which corresponds to the integral of the force curve, and stores this information for each of the measurement locations separately.

In this research, the work performed by all the six monitoring points was aggregated along the turnout so that the overall system performance

could be represented by the time series and so that the results would not be influenced by potential shifts within the single movement mechanisms. Since the turnout is positioned in different directions, the applied forces can vary. This research considered positioning in only one direction. The total observation period considered in this research was about 3.5 years, with approximately 16,000 measured positioning processes for one direction. The burn-in period was excluded from the analysed time series due to deviations in the setup process, adjustments of parameters etc.

#### *4.3. Preprocessing of the input data*

In the first step the data was preprocessed to handle outliers, smooth the data etc. In this pre-process, the outlier values in each of the sub-sets were removed by replacing them with the average value of the data points in the interval of 500 preceding and 500 succeeding data points. The time series data were then subdivided into intervals of 25 successive measurements and the values within these intervals were aggregated in order to reduce the data variability and increase the significance of the results. This resulted in a time series of 637 consecutive values.

Next, the time series data were normalized according to Equation 7. When using neural networks, it is important to account for possible deviations from the minimum and maximum values which are not covered by the selected dataset. These deviations can be caused by several factors such as changed environmental or operational parameters. Therefore, specified value ranges above the maximum value and below the minimum value are generally introduced to provide more flexibility to the algorithm. The range extension can, generally, be set according to the expected variability and evolution of the time series. The range extension is a standard process in data pre-processing. In case of complex valued neurons this range extension has an additional importance. Because the input and output values are, generally, converted to complex numbers (based on Equation 7), the values lie on a unit circle. Therefore, closeness between minimal and maximal values should be avoided. In this research, the data range was extended by 20% symmetrically above the maximum and below the minimum.

In the set-up and testing period, lower values than the ones in the considered dataset were observed. Because the normalization was performed on the whole data series, which included also the first part, the value range of for the analysed time series resulted in the interval  $[0.2, 0.9]$ . Due to information protection, the graphs are presented in the normalized value range.

#### 4.4. Predicting the degradation process of the railway turnout

The prediction of the turnout degradation level was performed using a time series of aggregated work values which represent the level of degradation. Lagged values from the time series were used to predict the subsequent values. Similar to the approach applied to the benchmark reliability data, multi-step ahead predictions were performed, whereby the prediction is performed iteratively one step at a time on the basis of the predictions at previous multiple steps. In this case study, a multi-step ahead prediction of 100 steps (which corresponds in average to approximately 6.5 months) was performed. The input length was set at 187 steps (which corresponds to the actual length of the first degradation cycle (Figure 3)). The input length for this case study was based on the contextual information. Generally, the input length should be sufficiently long to enable the algorithm to extract different underlying patterns and trends in the time series. The training dataset contained 350 values.

The minimum values of performed work in the time series correspond to periods with a low level of degradation and high turnout system performance, while the maximum values show points with high degradation. The goal was to predict when the maximum value would occur and the actual level of the maximum. The maximum point is assumed to be the point in time when the level of degradation has been reached at which maintenance actions are required.

Figure 3 compares the values predicted using the MLMVN algorithm to the training dataset and shows that the algorithm can long-term predict 100 consecutive values very precisely. Additionally, the point of time when the maximum value occurred was predicted precisely. This shows a good generalization ability of the algorithm. The NRMSE of the total 100 predicted points is 0.0116.

#### 4.5. Discussion of the prediction results

The prediction is imprecise in two areas of the curve. In the first third of the predicted values (values 563-592), a change in the degradation progression with a slight decrease of degradation is predicted, which was not observed in the actual dataset. A similar behaviour can be observed in the first degradation cycle (values 18-48) (Figure 3), where the degradation does not increase gradually but also has a change in the degradation progression in the first third of the cycle. This behaviour from the first cycle has been learned by the algorithm and transferred to prediction. The second imprecision can be

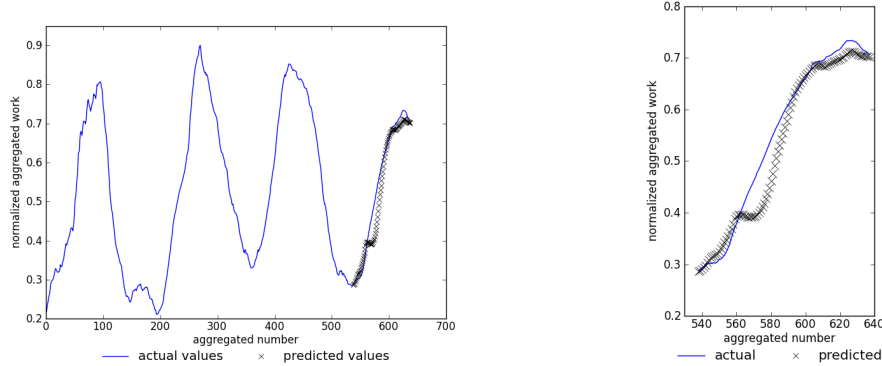


Figure 3: Predicted (crosses) and actual (solid line) levels of degradation of the turnout system

observed in the predictions around the maximum (values 605-637): the predicted work is slightly below the actual values; however, the point of time of the maximum is predicted precisely. In the corresponding region around the maximum in the testing set, the non-smoothed values of the performed work show several fluctuations and yet the prediction is capable to filter these fluctuations and predict the general trend.

The MLMVN were able to generalize the development of the time series and its cyclic behaviour seen during training, while properly adjusting the magnitude. In fact, the observed maximum values in the training dataset were all between 0.8 and 0.9 on the normalized scale, whereas the maximum in the testing dataset was 0.73.

In summary, the MLMVN showed a very good generalization ability and that it could perform a multi-step ahead prediction of 100 values in total with a very small error.

## 5. Conclusions

This research demonstrated that multilayer networks based on multi valued neurons are a powerful prediction tool for predicting reliability and degradation based on time series. In the case study considered, the MLMVN algorithms developed good results for multi-step ahead predictions and did not show accumulating errors. The results obtained show that MLMVN are

particularly suitable for mid- to long-term predictions. These results could be used by the railway operator for mid- to long-term maintenance and spare parts planning: the ability of the algorithm to predict well into the future means that the scheduled maintenance operations could be planned with a long lead time.

In the benchmark study used in this research, the MLMVN algorithm outperformed other algorithms and was able to perform multi-step ahead predictions with a very good prediction precision.

MLMVN appears not to tend to accumulate errors for multi-step ahead predictions that are based on previously predicted values. Furthermore, the algorithm is able to detect the evolution of several underlying long-term trends and processes, which is important in predicting dynamic degradation processes. In this case study, only the lagged time series values have been applied as input to the algorithm. However, MLMVN enables integrating additional influencing parameters, if this information is available, which can improve the prediction result and increase the flexibility of the approach. Therefore, MLMVN can be a powerful tool for predicting degradation paths requiring only the lagged observed measurements.

For long-term predictions, sufficiently long time series are required in order to detect the development of the underlying trends and generating processes, which may not always be available. In this case, MLMVN might not be able to detect and extract the underlying trends and generating processes, and to make precise predictions. Furthermore, MLMVN may not be able to react to a changed operating profile that has not been sufficiently long reflected in the time series, but which may have a significant influence on the degradation process. Additionally, if the trends and evolution of the influencing factors are short, they may not be able to be detected by the algorithm and be reflected in long-term predictions.

The analyses and predictions described in this paper were performed at a high level of system aggregation. If the prediction approach were to be used in industrial applications, the monitoring and prediction process would need to be switched to a finer level of aggregation prior to the point of time of the predicted maximum value of degradation, to ensure that the maintenance is performed at the optimal time and to identify the specific subsystem in which maintenance needs to be performed.

For further applications of MLMVN, especially for use in degradation prediction and prognostics, the uncertainty of the predictions needs to be quantified.

Given the strengths of MLMVN in several potential fields of application for reliability, diagnostics and prognostics, MLMVN can be applied not only to time series predictions but also to further problems in this field, such as associative memory tasks, classification tasks, signal processing or signal generation tasks. Verifying the strengths of MLMVN in these fields applied to reliability prediction problems is subject for further research.

## 6. Acknowledgement

The participation of Olga Fink to this research is partially supported by the Swiss National Science Foundation (SNF) under grant number 205121\_147175.

The participation of Enrico Zio to this research is partially supported by the China NSFC under grant number 71231001.

The authors would like to thank the BLS AG for providing the data for this research project.

## 7. References

- [1] B. Abbasi, S. Hosseinifard, D. Coit, A neural network applied to estimate burr {XII} distribution parameters, Reliability Engineering & System Safety 95 (2010) 647 – 654. URL: <http://www.sciencedirect.com/science/article/pii/S0951832010000360>. doi:<http://dx.doi.org/10.1016/j.ress.2010.02.001>.
- [2] I. Aizenberg, C. Moraga, Multilayer feedforward neural network based on multi-valued neurons (mlmvn) and a backpropagation learning algorithm, Soft Computing - A Fusion of Foundations, Methodologies and Applications 11 (2007) 169–183.
- [3] I. Aizenberg, D.V. Paliy, J.M. Zurada, J.T. Astola, Blur identification by multilayer neural network based on multivalued neurons, Neural Networks, IEEE Transactions on 19 (2008) 883–898.
- [4] I.N. Aizenberg, Complex-Valued Neural Networks with Multi-Valued Neurons, Springer, Berlin, 2011.
- [5] A.Z. Al-Garni, A. Jamal, Artificial neural network application of modeling failure rate for boeing 737 tires, Quality and Reliability Engineering International 27 (2011) 209–219.

- [6] A.B. Andre, E. Beltrame, J. Wainer, A combination of support vector machine and k-nearest neighbors for machine fault detection, *Applied Artificial Intelligence* 27 (2013) 36–49.
- [7] A.G. de Araujo Goes, M. Alvarenga, P.F. e Melo, Naroas: a neural network-based advanced operator support system for the assessment of systems reliability, *Reliability Engineering & System Safety* 87 (2005) 149 – 161. URL: <http://www.sciencedirect.com/science/article/pii/S095183200400016X>. doi:<http://dx.doi.org/10.1016/j.res.2004.01.010>.
- [8] V. Bagdonavicius, M.S. Nikulin, Estimation in degradation models with explanatory variables, *Lifetime Data Analysis* 7 (2001) 85–103.
- [9] M. Barakat, D. Lefebvre, M. Khalil, F. Druaux, O. Mustapha, Parameter selection algorithm with self adaptive growing neural network classifier for diagnosis issues, *International Journal of Machine Learning and Cybernetics* 4 (2013) 217–233.
- [10] W. Caesarendra, A. Widodo, T. Pham Hong, Y. Bo-Suk, J.D. Setiawan, Combined probability approach and indirect data-driven method for bearing degradation prognostics, *Reliability, IEEE Transactions on* 60 (2011) 14–20.
- [11] P. Chang, K. Lin, P. PF, Hybrid learning fuzzy neural models in forecasting engine system reliability, *Fifth Asia Pacific Industrial Engineering and Management Systems Conference* (2004) 2361–2366.
- [12] S. Chatterjee, S. Bandopadhyay, Reliability estimation using a genetic algorithm-based artificial neural network: An application to a load-haul-dump machine, *Expert Systems with Applications* 39 (2012) 10943–10951.
- [13] K.Y. Chen, Forecasting systems reliability based on support vector regression with genetic algorithms, *Reliability Engineering & System Safety* 92 (2007) 423–432.
- [14] Z. Chen, Bayesian filtering: From kalman filters to particle filters, and beyond, *Statistics* 182 (2003) 1–69.



- [15] X. Dang, T. Jiang, Reliability prediction based on degradation measure distribution and wavelet neural network, in: Prognostics and System Health Management (PHM), 2012 IEEE Conference on, IEEE, 2012, pp. 1–5.
- [16] P. Deuskiewicz, S. Radkowski, On-line condition monitoring of a power transmission unit of a rail vehicle, *Mechanical Systems and Signal Processing* 17 (2003) 1321–1334.
- [17] J. Du, S. Wang, H. Zhang, Layered clustering multi-fault diagnosis for hydraulic piston pump, *Mechanical Systems and Signal Processing* (2012).
- [18] O.F. Eker, F. Camci, A. Guclu, H. Yilboga, M. Sevkli, S. Baskan, A simple state-based prognostic model for railway turnout systems, *Industrial Electronics, IEEE Transactions on* 58 (2011) 1718–1726.
- [19] S. Fararooy, J. Allan, On-line condition monitoring of railway equipment using neural networks, in: *Advanced Condition Monitoring Systems for Railways, IEE Colloquium on*, pp. 2/1–2/9.
- [20] Z.u.R. Gondal, J. Lee, Reliability assessment using feed-forward neural network-based approximate meta-models, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* (2012).
- [21] N.J. Gordon, D.J. Salmond, A.F. Smith, Novel approach to nonlinear/non-gaussian bayesian state estimation, in: *IEE Proceedings F (Radar and Signal Processing)*, volume 140, IET, 1993, pp. 107–113.
- [22] S.S. Haykin, *Neural networks and learning machines*, 3rd ed., Pearson Education, Upper Saddle River, 2009.
- [23] C. Hu, B.D. Youn, P. Wang, J.T. Yoon, Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life, *Reliability Engineering & System Safety* 103 (2012) 120 – 135. URL: <http://www.sciencedirect.com/science/article/pii/S0951832012000427>. doi:<http://dx.doi.org/10.1016/j.ress.2012.03.008>.
- [24] C.H. Hu, X.S. Si, J.B. Yang, System reliability prediction model based on evidential reasoning algorithm with nonlinear optimization, *Expert Systems with Applications* 37 (2010) 2550–2562.

- [25] Q. Hu, M. Xie, S. Ng, G. Levitin, Robust recurrent neural network modeling for software fault detection and correction prediction, *Reliability Engineering & System Safety* 92 (2007) 332 – 340. URL: <http://www.sciencedirect.com/science/article/pii/S0951832006001001>. doi:<http://dx.doi.org/10.1016/j.ress.2006.04.007>.
- [26] J.E. Hurtado, D.A. Alvarez, Neural-network-based reliability analysis: a comparative study, *Computer Methods in Applied Mechanics and Engineering* 191 (2001) 113–132.
- [27] A. Jamali, M. Ghamati, B. Ahmadi, N. Nariman-zadeh, Probability of failure for uncertain control systems using neural networks and multi-objective uniform-diversity genetic algorithms (muga), *Engineering Applications of Artificial Intelligence* 26 (2013) 714–723.
- [28] G. Jin, D.E. Matthews, Z. Zhou, A bayesian framework for on-line degradation assessment and residual life prediction of secondary batteries inspacecraft, *Reliability Engineering & System Safety* 113 (2013) 7 – 20. URL: <http://www.sciencedirect.com/science/article/pii/S0951832012002670>. doi:<http://dx.doi.org/10.1016/j.ress.2012.12.011>.
- [29] B. Kaushik, N. Kaur, A.K. Kohli, Achieving maximum reliability in fault tolerant network design for variable networks, *Applied Soft Computing* (2013).
- [30] M. Khatibinia, M.J. Fadaee, J. Salajegheh, E. Salajegheh, Seismic reliability assessment of {RC} structures including soilstructure interaction using wavelet weighted least squares support vector machine, *Reliability Engineering & System Safety* 110 (2013) 22 – 33. URL: <http://www.sciencedirect.com/science/article/pii/S095183201200186X>. doi:<http://dx.doi.org/10.1016/j.ress.2012.09.006>.
- [31] Z. Kurd, T.P. Kelly, Using fuzzy self-organising maps for safety critical systems, *Reliability Engineering & System Safety* 92 (2007) 1563 – 1583. URL: <http://www.sciencedirect.com/science/article/pii/S0951832006002122>. doi:<http://dx.doi.org/10.1016/j.ress.2006.10.005>.

- [32] N. Lehrasab, H.P.B. Dassanayake, C. Roberts, S. Fararooy, C.J. Goodman, Industrial fault diagnosis: Pneumatic train door case study, Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 216 (2002) 175–183.
- [33] B. Li, M.Y. Chow, Y. Tipsuwan, J.C. Hung, Neural-network-based motor rolling bearing fault diagnosis, IEEE Transactions on Industrial Electronics 47 (2000) 1060–1069.
- [34] I.D. Lins, M.d.C. Moura, E. Zio, E.L. Drogue, A particle swarm-optimized support vector machine for reliability prediction, Quality and Reliability Engineering International 28 (2012) 141–158.
- [35] Y. Liu, H.Z. Huang, D. Ling, Reliability prediction for evolutionary product in the conceptual design phase using neural network-based fuzzy synthetic assessment, International Journal of Systems Science 44 (2013) 545–555.
- [36] S. Lolas, O.A. Olatunbosun, Prediction of vehicle reliability performance using artificial neural networks, Expert Systems with Applications 34 (2008) 2360–2369.
- [37] G. Marichal, M. Artes, J.G. Prada, O. Casanova, Extraction of rules for faulty bearing classification by a neuro-fuzzy approach, Mechanical Systems and Signal Processing 25 (2011) 2073 – 2082. URL: <http://www.sciencedirect.com/science/article/pii/S0888327011000240>. doi:<http://dx.doi.org/10.1016/j.ymssp.2011.01.014>.
- [38] M.d.C. Moura, E. Zio, I.D. Lins, E. Drogue, Failure and reliability prediction by support vector machines regression of time series data, Reliability Engineering & System Safety 96 (2011) 1527–1534.
- [39] M. Nourelfath, N. Nahas, Quantized hopfield networks for reliability optimization, Reliability Engineering & System Safety 81 (2003) 191–196.
- [40] P.F. Pai, System reliability forecasting by support vector machines with genetic algorithms, Mathematical and Computer Modelling 43 (2006) 262–274.

- [41] Y. Peng, H. Wang, J. Wang, D. Liu, X. Peng, A modified echo state network based remaining useful life estimation approach, in: Prognostics and Health Management (PHM), 2012 IEEE Conference on, IEEE, 2012, pp. 1–7.
- [42] J. Reifman, Survey of artificial intelligence methods for detection and identification of component faults in nuclear power plants, *Nuclear Technology* 119 (1997) 76–97.
- [43] C.M. Rocco S, Singular spectrum analysis and forecasting of failure time series, *Reliability Engineering & System Safety* 114 (2013) 126–136.
- [44] C.M. Rocco S, E. Zio, A support vector machine integrated system for the classification of operation anomalies in nuclear components and systems, *Reliability Engineering & System Safety* 92 (2007) 593–600.
- [45] Z. Sadovsky, C.G. Soares, Artificial neural network model of the strength of thin rectangular plates with weld induced initial imperfections, *Reliability Engineering & System Safety* 96 (2011) 713 – 717. URL: <http://www.sciencedirect.com/science/article/pii/S0951832011000275>. doi:<http://dx.doi.org/10.1016/j.ress.2011.02.010>.
- [46] A.E. Smith, D.W. Coit, L. Yun-Chia, Neural network models to anticipate failures of airport ground transportation vehicle doors, *Automation Science and Engineering, IEEE Transactions on* 7 (2010) 183–188.
- [47] K.L. Son, M. Fouladirad, A. Barros, E. Levrat, B. Iung, Remaining useful life estimation based on stochastic deterioration models: A comparative study, *Reliability Engineering & System Safety* 112 (2013) 165 – 175. URL: <http://www.sciencedirect.com/science/article/pii/S0951832012002529>. doi:<http://dx.doi.org/10.1016/j.ress.2012.11.022>.
- [48] C. Srivaree-Ratana, A. Konak, A.E. Smith, Estimation of all-terminal network reliability using an artificial neural network, *Computers & Operations Research* 29 (2002) 849–868.
- [49] M. Suliman, M.A. Manzoul, Neural network realization of markov reliability and fault-tolerance models, *Microelectronics and Reliability* 31 (1991) 141–147.

- [50] R.M. Tallam, T.G. Habetler, R.G. Harley, Self-commissioning training algorithms for neural networks with applications to electric machine fault diagnostics, *IEEE Transactions on Power Electronics* 17 (2002) 1089–1095.
- [51] P. Tamilselvan, P. Wang, Failure diagnosis using deep belief learning based health state classification, *Reliability Engineering & System Safety* 115 (2013) 124–135.
- [52] P. Tamilselvan, P. Wang, Failure diagnosis using deep belief learning based health state classification, *Reliability Engineering & System Safety* 115 (2013) 124 – 135. URL: <http://www.sciencedirect.com/science/article/pii/S0951832013000574>. doi:<http://dx.doi.org/10.1016/j.ress.2013.02.022>.
- [53] W.L. Tan, N.M. Nor, M.Z. Abu Bakar, Z. Ahmad, S.A. Sata, Optimum parameters for fault detection and diagnosis system of batch reaction using multiple neural networks, *Journal of Loss Prevention in the Process Industries* 25 (2012) 138–141.
- [54] A. Tawafan, M.B. Sulaiman, Z.B. Ibrahim, Adaptive neural subtractive clustering fuzzy inference system for the detection of high impedance fault on distribution power system, *IAES International Journal of Artificial Intelligence (IJ-AI)* 1 (2012) 63–72.
- [55] Z. Tian, L. Wong, N. Safaei, A neural network approach for remaining useful life prediction utilizing both failure and suspension histories, *Mechanical Systems and Signal Processing* 24 (2010) 1542–1555.
- [56] G. Vachtsevanos, *Intelligent fault diagnosis and prognosis for engineering systems*, Wiley, Hoboken, NJ, 2006.
- [57] M.L.D. Wong, L.B. Jack, A.K. Nandi, Modified self-organising map for automated novelty detection applied to vibration signal monitoring, *Mechanical Systems and Signal Processing* 20 (2006) 593–610.
- [58] K. Xu, M. Xie, L.C. Tang, S.L. Ho, Application of neural networks in forecasting engine systems reliability, *Applied Soft Computing* 2 (2003) 255–268.

- [59] R.C.M. Yam, P.W. Tse, L. Li, P. Tu, Intelligent predictive decision support system for condition-based maintenance, *The International Journal of Advanced Manufacturing Technology* 17 (2001) 383–391.
- [60] D.M. Yang, A.F. Stronach, P. MacConnell, J. Penman, Third-order spectral techniques for the diagnosis of motor bearing condition using artificial neural networks, *Mechanical Systems and Signal Processing* 16 (2002) 391–411.
- [61] E. Zio, M. Broggi, L.R. Golea, N. Pedroni, Failure and reliability predictions by infinite impulse response locally recurrent neural networks, *Chemical engineering transactions* 26 (2012).