# Introduction to an Adaptive Remaining Useful Life Prediction for forming tools

Christoph Kellermann
*Gerresheimer Bünde GmbH*
Bünde, Germany
christoph.kellermann@gerresheimer.com

Yeremia Gunawan Adhisantoso
*Leibniz Universität Hannover*
Hannover, Germany
adhisant@tnt.uni-hannover.de

Eric Neumann
*Gerresheimer Bünde GmbH*
Bünde, Germany
eric.neumann@gerresheimer.com

Marco Munderloh
*Leibniz Universität Hannover*
Hannover, Germany
munderl@tnt.uni-hannover.de

Jörn Ostermann
*Leibniz Universität Hannover*
Hannover, Germany
office@tnt.uni-hannover.de

***Abstract*—As key components in the field of industry 4.0, data of sensors is often used for checking and observing the quality of subsystems. In modern manufacturing environments this huge amount of data enables machine health monitoring tools to analyze the behavior of mechanical components over time e.g. to estimate the remaining useful life (RUL) before breakdown. In this paper a system based on an autoencoder alike structure to forecast the deterioration of components is introduced. It is capable to predict the RUL based on the historical stress and usage conditions and identify anomalies like occurring faults by predicting the future with the encoder part, projecting it backwards with the decoder part, and then comparing it with the original data. The degradation forecast is estimated with respect to direct measurable parameters and not using a virtual health index. Our approach estimates the RUL on limited and noisy data and does not require knowledge of the true RUL. With the proposed setup our model is scalable to other production line configurations and product derivatives with different given production or quality thresholds without the need of a new training. We use real process data as well as synthetic signals for the training of the neural networks to improve the performance. We evaluate and demonstrate the performance of our RUL estimation approach against established forecast methods in the field of glass forming processes. We show that our approach of time series prediction in comparison to established prediction methods like RANSAC or ARIMA which require background knowledge delivers comparable accuracy and can additionally predict abnormal behavior.**

## I. Introduction

In the field of maintenance many policies help to reduce cost and to decrease the unscheduled downtime caused by unexpected failures or similar. The typically fixed time intervals of conventional preventive maintenance policies which neglects the historical stress and other influencing characteristics are a major disadvantage [1]. Conventional approaches indicate the health of a machine by using health degradation curves that have an exponential or linear shape [2], [3]. Apart from these empirical degradation curves, physics-based models consider physical assumptions from experts and model the mechanical fading by explicit measurable parameters [3], [4]. With increasing complexity of the manufacturing process or machines it is difficult to build a valid physics-based model [5]. Furthermore, signals are affected by noise such that the machine or component degradation is only vaguely known. It is based on prior knowledge and assumptions about how a system degrades or a fault evolves.

Our adaptive RUL algorithm tackles the problem of manually or adaptively evaluated mean time to failure based on a time series forecast with neural networks. The exact unique forecast of a component's lifetime becomes possible due to continuous monitoring of the usage behavior and prediction of the remaining lifetime by a combination of the deterioration based on the stress history component wise and given quality tolerances of the final product. Typically the wear components are replaced periodically so that there is no prior knowledge about the expected real lifetime nor the actual remaining lifetime available. The ground truth is missing and historical data is not useful to directly learn the correct RUL. In this case design of experiments (DoEs) [6] or many tests as presented in the international standard IEC 60812 [7] have to be done to get sufficient training data covering the complete life cycle.

Key contributions of this work are:

- degradation forecast by an autoencoder alike structure with limited data (and unknown real RUL) and noisy sensor data
- robust RUL estimation using the encoder part
- anomaly detection using the decoder part (detection of faults or other occurrence)
- calculation of an end of life threshold independent of the individual product specification limits and the health of the forming tool
- scaling ability to similar machines and products without the need of re-training for all machines in the field having different mechanical and dynamic conditions

This paper is divided into 5 sections. In Section II we briefly explain several techniques for RUL estimation followed by a short introduction of anomaly detection. Section III contains the proposed method to forecast the remaining useful life in operational and training mode. In Section IV, we show how to handle fewer training data and compare our approach to a real-world scenario. The evaluation scenario in Section V is located in glass forming application in which partly hot glass tubes are formed to a final glass product (e.g. syringe) and deterioration happens due to high temperatures and distortion forces. Anomalies in this process are evoked by cracks in the glass and operator interactions to stabilize the process. The conclusions are drawn in Section VI.

## II. Related Work

In recent years the growth of available data has been exponential thanks to electromechanical systems with their huge number of sensors. The increased computational power allows improving the monitoring of systems [5], [8]. Unfortunately, a huge part of the data is unlabeled thus an unsupervised approach is required to exploit the information within. The amount of the data leads to a more precise forecast in the field of preventive maintenance and fading components. Much research and experiments have already been done in the field of degradation behaviors especially in rotary machinery and induction motors to classify operation conditions and to estimate the health of the machine [9], [10], [11]. In the field of aerospace systems like the main gearbox of helicopters Yu et al. [10] did an analysis and evaluation with respect to the heath conditions. In this section similar applications and solutions in the field of RUL estimation are compared and in the second part a screening of several approaches with the usage of neural networks is done.

### A. Estimation of Remaining Useful Life

The knowledge of the deterioration in a real production environment is the key to establish a highly reliable production. Mechatronic components operate under certain stresses and loads. Each has unique properties; hence the Remaining Useful Life (RUL) is unique for each component or subsystem before a faulty behavior occurs and followed by increasing scrap or machine downtime. Jardine et al. [1] surveyed different approaches (from statistically to AI) to estimate RULs. All RUL estimation approaches need additional knowledge about failure mechanisms to give a prognosis. Sometimes the judgments are stated by domain experts in other cases experimental data is used. Unlike unlabeled data, labeled data such as data with health state information is expensive to produce. Yet labeled data is very important to create a useful model. In order to find an applicable way to estimate the health degradation trend and not relying on domain experts' know-how an unsupervised health index was introduced by Malhotra et al. [2]. The RUL is estimated by the similarity to health index curves which are based on the reconstruction error of long short-term memory (LSTM) autoencoders. LSTM models are often used for anomaly detection by learning a model to reconstruct time series [2]. Recently, the usage of other neural networks like fully-connected neural networks (FNN) or convolutional neural network (CNN) for forecasting time series have been adopted in various areas such as estimation of electric load in power network [12].

Often advanced signal processing is used to extract useful features for diagnostics. For example the analysis of the frequency spectrum is commonly used to determine the health state of rotary machinery because their fundamental components are corresponding to the individual frequencies and can often provide explicit information on the component's condition [13], [11].

Effective fault diagnosis and current health state monitoring becomes challenging with increasing ambient noise. If the noise dominates the signal dynamic it effects heavily the RUL estimations or classifications and a higher number of training datasets is required. This problem arises especially in iterative learning [5] and self-learning processes [9]. Lu Chen at al. [9] proposed a stacked denoising autoencoder-based health state identification for various working conditions by encoding the signal and capturing statistical dependencies between the sensors.

In [9] the data acquisition and evaluation strategy for health indication of rolling bearings in rotary machinery is described. The data and the wear behavior were collected at a bearing test rig. Each test case was assigned to one corresponding operating condition. Through this very controlled approach, each dataset had an implicit label which can be used for learning the class. In practice an important influence on the performance and reliability of the fading components is the mechanical setup in the machine.

In [10], Yu et al. considered multiple faults that happen simultaneously and introduced a model-based prognosis for hybrid systems. Hybrid systems take into account that multiple faults can be incipient at different times and superpose each other. Yu et al. presented a concept of a dynamic fault isolation which separates the different faults by waiting a defined period to exhibit their symptoms on residuals of a binary coherence vector. Similar to [9], the system behavior has to be known in every condition.

### B. Autoencoder

One of the deep learning methods called autoencoder (AE) has proven effective in many fields such as image recognition [14] and audio [15] (e.g. speech) recognition. Features are learned in an unsupervised manner by minimizing the reconstruction error. The characteristic of AEs (consisting of encoder and decoder) is that the target data is concurrent with the input data. The encoder transforms the input data into a low-dimensional space which is the output of the encoder network. The goal of the decoder network is to reconstruct the inputs of the encoder using the low-dimensional output of the encoder. Both coders are trained simultaneously [13]. Using this method it is achieved that all the essential information of the input is extracted and contained in the low-dimensional space. The internal network design of the encoder and decoder depends on the needs.

However, a problem of a basic autoencoder depending on too few data could be that it extracts insignificant features due to simply copying the input layer to the hidden layer if the output can be perfectly recovered [11]. The training as well as the generalization error are extremely poor if standard initialization with random values is used [9]. Because of this situation, different approaches try to expand the basic autoencoders.

Each extension tries to penalize the autoencoder and push it to learn more features and provide robust feature representations. For efficient learning of autoencoders in the training phase Vincent et al. [16] proposed corrupting initial inputs by a partially destroyed version of the input. The corrupted input is then mapped to the input of the autoencoder while at the output the uncorrupted input is presented. Through this denoising approach the autoencoder learns to recover corrupted input data. Alternatively, the corruption can be done with additional noise. Further well-known data augmentation techniques are transformations such as rotation, scale, translation [16] for images. Stacked denoising autoencoders [9] employ techniques of the associated neural networks. The hidden layers of the target AE are layer-wise pre-trained in an extra three layer network before global training of the target AE. Through this combination of different AEs, the final AE employs a better feature representation in each layer. Sparse autoencoders [11] are a derivation of normal AE. Their training criterion involves a sparsity penalty by penalizing activations of hidden layers to control the number of active neurons.

### *C. Anomaly detection*

Different definitions of an anomaly exist in the literature. In contrast to the classic definition of anomalies in this paper anomalies are not defined as outliers in a time series. Here we focus more on the evolution of our input data that we observe within a certain period of time, categorized as a normal or an anomalous event [17]. By using an autoencoder in sequence-to-sequence mode and presenting the AE while training only sequences of normal events (normal machine mode) the AE can detect the perfect health or not failed state of the machine in operational mode. The learned model could reconstruct the subsequences which belong to normal behavior [2]. This leads to a high reconstruction error when an anomaly or user interaction appears in operational mode because the model has not seen such data during the training [2].

## III. Proposed method for adaptive Remaining Useful Life Prediction

This concept tackles the problem of the missing ground truth of the deterioration and its threshold that corresponds to the failure of a component. We saw in Section II-B that models, which use health indexing need labeled data. In practice those data which contain the real remaining machine cycles (processing time and time for transporting of a device to the next production step) is rarely available. Rep resentative experiments e.g., DoEs on a testing rig must be executed to get the missing data. Stretching using cycles of machine components can be very risky due to a lack of reliable methods of fault estimation and mostly ends up in mechanical breakdowns and collateral damage. Thresholds for classification or RUL are restricted to the classes of the machinery and those thresholds are not always applicable to other machine designs or product derivatives because of lacking generalization in terms of covering all situations.

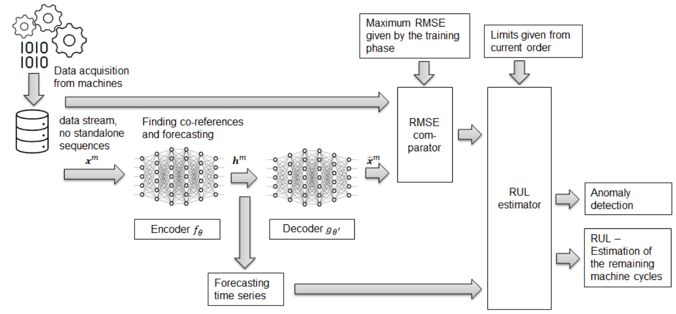



Fig. 1. Architecture of the autoencoder health estimation algorithm.

To tackle the challenges mentioned above, we propose the adaptive health state estimation approach based on an autoencoder alike structure which forecasts trends of the product specification parameters in the time domain while being able to detect anomaly by back projecting the forecast into the past using decoder part of the neural network. By estimating the trend of the product specification parameter we derive the wear condition instead of using the health index, where the threshold comes from the product specification. The End of Life (EOL) point in time is determined when the product parameter falls below its given limit value. The Root-Mean-Square-Error (RMSE) is used for the comparison between current input time series and the reconstructed time series of the decoder.

Fig. 1 provides an overview of the steps to estimate the RUL and to detect abnormal behavior or conditions. The autoencoder network alike structure consists of the encoder and decoder structure. The encoder extracts characteristics and important features/patterns of the inputs to predict the next time steps by the given previous values. It takes the raw sensor time-series data and transfers the measurement space to feature space. The encoder is used to map a multivariate input sequence $\mathbf{x}^m = (x_1^m, \ldots, x_i^m)$ from the dataset $\{\mathbf{x}^m\}_{m=1}^M$ to a fixed dimensional vector representation $\mathbf{h}^m = f_\theta(\mathbf{x}^m) = (x_{i+1}^m, \ldots, x_{i+n}^m)$ to predict a certain time horizon of size $n$. $\mathbf{h}^m$ directly represents the time series forecast. $f_\theta$ is the encoding function. The decoder is an independent network which uses the vector representation $\mathbf{h}^m$ to recreate the target sequence $\hat{\mathbf{x}}^m = g_{\theta'}(\mathbf{h}^m) = (\hat{x}_1^m, \ldots, \hat{x}_i^m)$. The reconstruction function is denoted as $g_{\theta'}$ and represents the decoder. Encoder and decoder networks are based on widely known architecture such as FNNs, LSTMs or CNNs [18]. If many input parameters are given, CNN should be used for a high-order feature representation. The suspected End of Life (EOL) is defined by the time step in which the forecast $\mathbf{h}^m$ exceeds the given threshold/limit per parameter from the product specification (see Fig. 2). The EOL is a fixed time in the future in the prediction horizon $n \cdot \tau = t_{max}$, $\tau$ denotes the machine cycle. Fig. 2 illustrates prognostic time definitions and the prediction concept as described in Fig. 1. EOL can be denoted as

$$\mathrm{EOL}(t_n) \in \{t \in R : t_{max} \geq t_n\},$$

whereas $t_n$ is the current time. The Remaining Useful Time

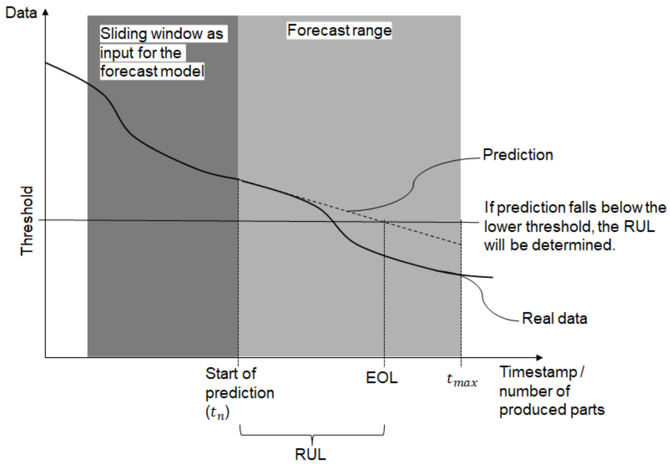



Fig. 2. Illustration of the time definitions and the prediction concept.

(RUL) is defined as

$$\mathrm{RUL}\left(t_n\right) = \mathrm{EOL}\left(t_n\right) - t_n.$$

The estimation $\hat{\mathbf{x}}^m$ of the decoder is transmitted to a comperator which calculates the $RMSE\left(\hat{\mathbf{x}}^m, \mathbf{x}^m\right)$ between $\hat{\mathbf{x}}^m$ and $\mathbf{x}^m$ and compares this result to a given RMSE threshold. This given RMSE $max\left\{RMSE_{train}\left(\hat{\mathbf{x}}^m, \mathbf{x}^m\right)\right\}$ is the maximum RMSE value detected in the training phase. The comparison of the RMSEs enables the RUL estimator to calculate the RUL based on the time forecast and the given limits of the current specification. Or if the $RMSE\left(\hat{\mathbf{x}}^m, \mathbf{x}^m\right)$ is less than $max\left\{RMSE_{train}\left(\hat{\mathbf{x}}^m, \mathbf{x}^m\right)\right\}$, the RUL estimator signalizes anomaly. In summary, the reconstruction error of the decoder indicates an anomaly. That is based on the fact that only valid data of normal machine conditions is trained so that the decoder network can perfectly recover patterns which were marked before as normal machine/component conditions.

The parameters of the encoder and decoder are learned simultaneously to archive the lowest possible reconstruction error. Before sequence to sequence training the data must be prepared according to Fig. 3 so that the data has the correct tensor size for presenting it to the network.

## IV. Simulation Results and Observations

In the evaluation we focus on the RUL estimation performance of our scenario described in Section I. For the evaluation we focus one feature prediction but it can be extended to multiple features. We evaluated our proposed approach for RUL estimation on two datasets - first on a synthetic dataset which represents the characteristics of the real data and contains different machines and product setups. We present the details of generating the synthetic data in Section IV-A. After presenting the essential metrics for performance comparisons we show the results for our approach of estimating the RUL performance on synthetic and real-world datasets in Section V. The real-world dataset was derived from multiple forming machines and is used to evaluate the proposed method described in Sections III. The raw signals refer to the measured signals in the time domain. In this case the time index is discrete and symbolized by the machine cycle.

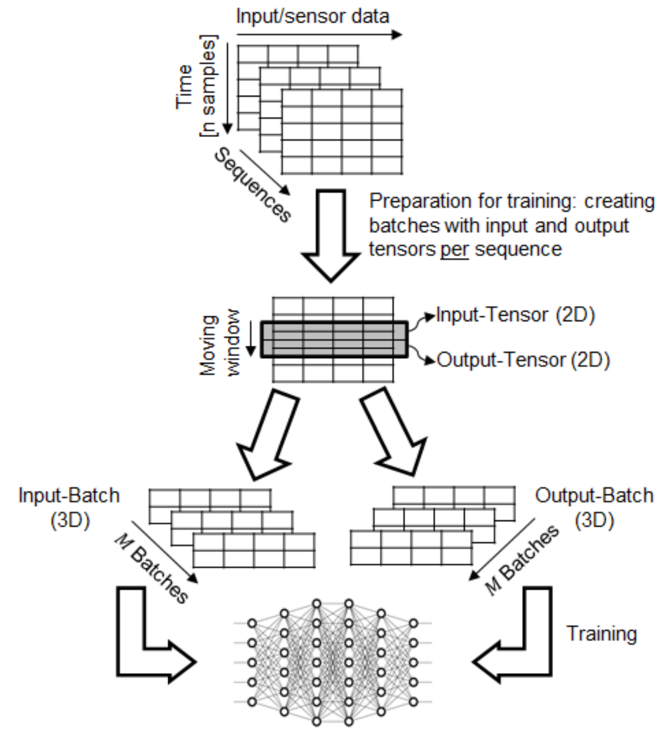
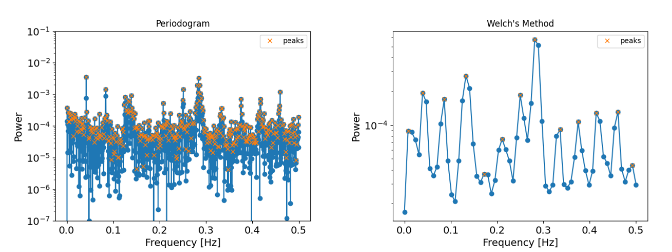



Fig. 3. Data preparation for the training.

### A. Artificial Training data generation

Due to the limitation of training data and for additional evaluation like the impact of noise, we created a signal generator to artificially synthesize process data. To do that, we first analyzed the data to find any useful insight. The most common approach for time series analysis is time series decomposition and thus dominant frequency analysis was used to extract the trend and periodic components. The dominant frequency of a signal can be computed by the square of the norm of the Fourier transformed auto-correlation function called periodogram. However, for input data which consist of multiple superimposed signals the periodogram might produce noisy results with a finite number of observations, as can be seen in Fig. 4a). The Welch's method [19] is a better way to achieve a less noisy result but at the expense of a lower spectral resolution 4b).




(a) Periodogram (b) Welch's method

Fig. 4. Spectral analysis using a) periodogram and b) Welch's method. Dominant frequencies are at 0.15 Hz and 0.27 Hz.

As depicted in Fig. 4a) and 4b), our data contains of several frequencies. Based on this information, we designed a signal

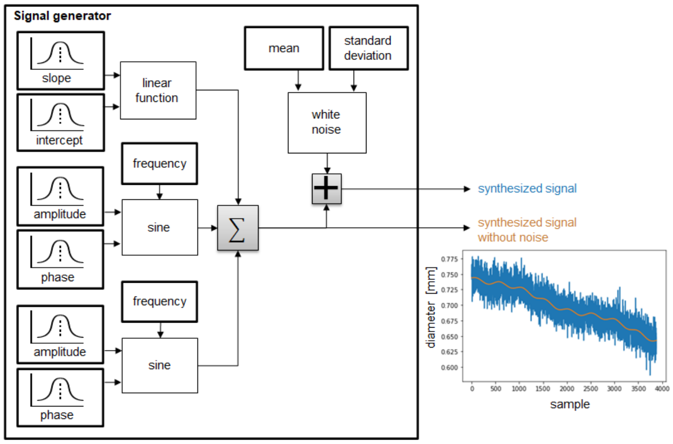



Fig. 5. Block diagram of the signal generator, all blocks with a gaussian symbol are varied for one certain synthetical dataset

generator given in Fig. 5 and generated synthetic process data for the training using our observed statistical characteristics provided in Tab. I. The trend signal is modulated by a linear function with slope and intercept drawn from Gaussian distributions. The parameters are computed from all real datasets. The two sine waves represent the two main dominant frequencies commonly occured and have the greatest amplitude in the real datasets in Fig. 4. Additive white Gaussian noise is finally added to superimposed signal.

TABLE I
PARAMETER SET USED FOR THE SIGNAL GENERATOR

| Function | Parameter | Value/Range |
|---|---|---|
| Linear function | slope | $[-3\cdot 10^{-5}\ldots 1\cdot 10^{-5}]$ |
| | intercept | $[0.1\ldots 0.65]$ |
| Sine 1 | frequency | $[0.15\pm 20Hz]$ |
| | amplitude | $[1\cdot 10^{-4}\ldots 3\cdot 10^{-3}]$ |
| | phase | $[0°\ldots 180°]$ |
| Sine 2 | frequency | $[0.27\pm 20Hz]$ |
| | amplitude | $[1\cdot 10^{-4}\ldots 5\cdot 10^{-3}]$ |
| | phase | $[0°\ldots 180°]$ |
| White noise | mean | 0 |
| | standard deviation | 0.05 |

### B. Performance metrics

The effectiveness of the proposed estimator and training with the synthetic data is verified by using several metrics to evaluate systems performance [20] which are often used for evaluation of prognostic models. The evaluations are application oriented. The standards ISO 13381-1 and IEC 60812 do not provide a definition of prognostic metrics. In the standards only a generic process is presented. The choice of the suitable model to estimate the RUL and to define the confidence interval depends on the type of given data and concrete degradation mechanisms [21], [7]. We used the following accuracy based metrics which score the similarity between real data and prediction [20] for evaluation:

$$RMSE = \sqrt{\frac{1}{L}\sum_{i=0}^{L}\Delta\left(i\right)^2},\quad MAE = \frac{1}{L}\sum_{i=0}^{L}\left|\Delta\left(i\right)\right|$$

$\Delta\left(i\right) = \mathbf{h}^m\left(i\right) - x\left(i\right)$ is the error between the estimated value $\mathbf{h}^m$ and the actual value $x\left(i\right)$.

## V. EVALUATION

We divided the 500 real process datasets into 3 sets. For training we used the training and the test datasets which cover 70% of all sequences (10% of it for testing). The remaining 30% have been used for validation only. We evaluated the forecast performance using the metrics presented in Section IV-B in two ways. Firstly, we measured the time series forecast accuracy for several networks trained with real process data (Tab. II) or synthetic data (Tab. III) for different prediction lengths. Secondly, we evaluated the influence of enriching the training dataset with the generated synthetic data. We compared the RUL estimation accuracy for the real process data for networks either trained using the enriched dataset to networks trained by real process data only. The impact of noise was evaluated using different noise settings for the signal generator during the training phase (Tab. IV).

- FNN (Encoder structure: Flatten-, 3x Linear- with leaky ReLU, Reshape-Layer; decoder is mirrored encoder.)
- CNN (Encoder structure: Reshape-, 2x Conv1D- with leaky ReLU, Reshape-, Flatten-, Linear- with ReLU, Reshape-Layer; decoder is mirrored encoder.)
- LSTM (Encoder structure: LSTM, Linear-Layer; the decoder has the same structure.)

As reference we use established methods which do not need to be trained:

- RANSAC: Linear regression model [22]
- ARIMA: An autoregressive integrated moving average model for time series forecasts [23]

Tab. II and Tab. III give the metrics for selected forecast lengths. The best forecast results of the three networks are highlighted. In both training scenarios the neural networks (FNN, CNN, LSTM) shows increasing forecast errors with increasing forecast length. For this time series with less complex characteristics but with high noise the FNN and CNN show a good performance compared to the RANSAC benchmark method. By enriching the dataset with synthetic data the RUL estimation performance of the FNN and the LSTM could be improved by 21.88% and 6.55%, respectively. We suppose that the generalization performance of the FNN is improved by the synthetic data as it represents a wider parameter spread. The CNN structure was able to extract sufficient features out of the real datasets. The results of Tab. IV demonstrate that the training with the correct configuration (in this case a noise level of 0.05) of the synthetic data related to the real process data leads to a good performance. Finally, we normalized the best RUL accuracy (RANSAC method). In the evaluation depicted in Tab. IV, the neural networks that trained with the synthetic data were used. Again, the two FNN and CNN networks reach a similar performance to the RANSAC reference method. From the evaluation we see that although the neural networks have many degrees of freedom, simple function processes such as in this selected application deliver good results. In contrast, prior knowledge like frequency characteristics of the time series is required for the established ARIMA and linear regression methods.

TABLE II
PERFORMANCE COMPARISON OF THE PREDICTION ACCURACY TRAINED WITH REAL PROCESS DATA, EVALUATED AGAINST REAL PROCESS DATA.

| Forecast | 500 machine cycles | | | | |
|---|---|---|---|---|---|
| **Method** | FNN | CNN | LSTM | ARIMA | RANSAC |
| **RMSE** | 0.01184 | **0.00716** | 0.02538 | 0.49781 | **0.00557** |
| **MAE** | 0.01096 | **0.00646** | 0.02511 | 0.19215 | **0.00507** |
| **Forecast** | 1000 machine cycles | | | | |
| **Method** | FNN | CNN | LSTM | ARIMA | RANSAC |
| **RMSE** | 0.01588 | **0.01087** | 0.02679 | 0.23162 | **0.00841** |
| **MAE** | 0.01447 | **0.00960** | 0.02634 | 0.10467 | **0.00755** |

TABLE III
PERFORMANCE COMPARISON OF THE FORECAST ACCURACY TRAINED WITH SYNTHETIC DATA, EVALUATED AGAINST REAL PROCESS DATA.

| Forecast | 500 machine cycles | | | | |
|---|---|---|---|---|---|
| **Method** | FNN | CNN | LSTM | ARIMA | RANSAC |
| **RMSE** | **0.00615** | 0.00701 | 0.03833 | 0.49781 | **0.00557** |
| **MAE** | **0.00560** | 0.00623 | 0.03636 | 0.19215 | **0.00507** |
| **Forecast** | 1000 machine cycles | | | | |
| **Method** | FNN | CNN | LSTM | ARIMA | RANSAC |
| **RMSE** | **0.00794** | 0.00859 | 0.04862 | 0.23162 | **0.00841** |
| **MAE** | **0.00706** | 0.00759 | 0.04638 | 0.10467 | **0.00755** |

In the case of more complicated applications, the neural networks will be superior. In addition, our proposed concept can be used to predict abnormal behavior.

TABLE IV
COMPARISON OF THE TIME SERIES FORECAST ACCURACY WITH DIFFERENT NOISE LEVEL OF THE SYNTHETIC DATA ASSESSED AGAINST REAL DATA. FORECAST WAS 1000 MACHINE CYCLES.

| Noise $\sigma$ | RMSE | | | MAE | | |
|---|---|---|---|---|---|---|
| | FNN | CNN | LSTM | FNN | CNN | LSTM |
| **0.05** | 0.008 | 0.009 | 0.047 | 0.007 | 0.008 | 0.046 |
| **0.015** | 0.017 | 0.453 | 0.024 | 0.015 | 0.146 | 0.021 |

## VI. CONCLUSION

The health condition of wear parts often affects the performance of the machine which leads to breakdowns and heavy economic losses. We saw that a complex domain knowledge is required and not all mechanical condition of the machine and user can be captured. Compared to traditional diagnostic algorithms, the main advantage of our proposes method is that the algorithm takes direct data from the specification and the model is independent to machine setup and product specification. Through the evaluation of the different datasets it is shown that the proposed RUL estimation and the training strategy with the synthetic data is working. Based on this proposed method further work will include more integration of relevant data sources to the algorithm and more experimental tests to detect the complex temporal dependencies between the different sensors which indicates the degradation trend to extract more useful knowledge through this information fusion and get more precise decisions.

TABLE V
RELATIVE RUL PERFORMANCE COMPARISON NORMALIZED BY RANSAC.

| Method | FNN | CNN | LSTM | ARIMA | RANSAC |
|---|---|---|---|---|---|
| **RMSE RUL** | 105% | 102% | 149% | 122% | 100% |
| **MAE RUL** | 108% | 106% | 160% | 126% | 100% |

## REFERENCES


[1] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, no. 7, pp. 1483 – 1510, 2006.

[2] P. Malhotra, V. TV, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Multi-sensor prognostics using an unsupervised health index based on lstm encoder-decoder," 2016.

[3] N. Gugulothu, V. TV, P. Malhotra, L. Vig, P. Agarwal, and G. Shroff, "Predicting remaining useful life using time series embeddings based on recurrent neural networks," 2017.

[4] C. H. Oppenheimer and K. A. Loparo, "Physically based diagnosis and prognosis of cracked rotor shafts," in *Component and Systems Diagnostics, Prognostics, and Health Management II* (P. K. Willett and T. Kirubarajan, eds.), vol. 4733, pp. 122 – 132, International Society for Optics and Photonics, SPIE, 2002.

[5] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: An overview," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 657–667, 2015.

[6] D. C. Montgomery, *Design and analysis of experiments*. John wiley & sons, 2017.

[7] IEC60812, *Failure Modes and Effects Analysis (FMEA and FMECA)*. International Electrotechnical Commission, 2018.

[8] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *Trans. on Indust. Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[9] C. Lu, Z.-Y. Wang, W.-L. Qin, and J. Ma, "Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification," *Sig. Proc.*, vol. 130, pp. 377 – 388, 2017.

[10] M. Yu, D. Wang, and M. Luo, "Model-based prognosis for hybrid systems with mode-dependent degradation behaviors," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 1, pp. 546–554, 2014.

[11] W. Sun, S. Shao, R. Zhao, R. Yan, X. Zhang, and X. Chen, "A sparse auto-encoder-based deep neural network approach for induction motor faults classification," *Measurement*, vol. 89, pp. 171 – 178, 2016.

[12] A. Gasparin, S. Lukovic, and C. Alippi, "Deep learning for time series forecasting: The electric load case," 2019.

[13] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mechanical Systems and Signal Processing*, vol. 72-73, pp. 303 – 315, 2016.

[14] C. Xing, L. Ma, and X. Yang, "Stacked denoise autoencoder based feature extraction and classification for hyperspectral images," *Journal of Sensors*, vol. 2016, 2016.

[15] Y.-A. Chung, C.-C. Wu, C.-H. Shen, H.-Y. Lee, and L.-S. Lee, "Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder," 2016.

[16] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, (New York, NY, USA), pp. 1096–1103, Association for Computing Machinery, 2008.

[17] M. Atzmueller, A. Schmidt, and D. Arnu, "Sequential Modeling and Structural Anomaly Analytics in Industrial Production Environments," in *Proc. LWA 2016 (KDML Special Track)*, (Potsdam, Germany), University of Potsdam, 2016.

[18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[19] P. Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on audio and electroacoustics*, vol. 15, no. 2, pp. 70–73, 1967.

[20] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," *2008 International Conference on Prognostics and Health Management*, pp. 1–17, 2008.

[21] ISO, "13381-1:2015," *Condition Monitoring and Diagnostics of Machines*, 2015.

[22] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *ACM*, vol. 24, pp. 381–395, June 1981.

[23] J. Yan, M. Koç, and J. Lee, "A prognostic algorithm for machine performance assessment and its application," *Production Planning & Control*, vol. 15, no. 8, pp. 796–801, 2004.