IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

IEEE RELIABILITY SOCIETY SECTION

Received September 10, 2021, accepted September 27, 2021, date of publication September 29, 2021,
date of current version October 13, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3116813

A Gaussian Process Based Method for Data-Efficient Remaining Useful Life Estimation

MAXIMILIAN BENKER[1], ARTEM BLIZNYUK[1,2], AND MICHAEL F. ZAEH[1]
[1]Institute for Machine Tools and Industrial Management, Technical University of Munich (TUM), 85748 Garching, Germany
[2]Institute for Man-Machine Interaction, RWTH Aachen University, 52074 Aachen, Germany

Corresponding author: Maximilian Benker (maximilian.benker@iwb.tum.de)

This work was supported by the European Union's Horizon 2020 Research and Innovation Programme under Grant 768575.

ABSTRACT The task of remaining useful life (RUL) estimation is a major challenge within the field of prognostics and health management (PHM). The quality of the RUL estimates determines the economical feasibility of the application of predictive maintenance strategies, that rely on accurate predictions. Hence, many effective methods for RUL estimation have been developed in the recent years. Especially deep learning methods have been among the best performing ones setting new record accuracies on bench mark data sets. However, those approaches often rely on numerous and representative run-to-failure sequences of the components under investigation. In real-world use cases, this kind of data (i.e. run-to-failure sequences and RUL labels) is hardly ever present. Therefore, this paper proposes a new, data-efficient method, which is based on Gaussian process classification to derive abstract health indicator (HI) values in a first step, and warped, monotonic Gaussian process regression for indirect RUL estimation in a second step. The proposed approach does neither rely on entire run-to-failure sequences nor on any RUL labels and was tested on the benchmark C-MAPSS turbo fan and FEMTO bearing data sets, achieving comparable results to the state-of-the art whilst using only a small fraction of the available training data. Hence, the proposed approach allows RUL estimation in use cases, in which gathering enough failure data for the application of deep learning models is infeasible.

INDEX TERMS C-MAPSS, gaussian processes, prognostics and health management, remaining useful life estimation.

NOMENCLATURE

AI	Artificial intelligence.
DI	Discrad large RUL estimates; strategy to handle large RUL estimates, described in Section IV.
DL	Deep learning.
EP	Expectation propagation.
FFT	Fast fourier transformation.
GP	Gaussian process.
GPC	Gaussian process classification.
GPR	Gaussian process regression.
HI	Health indicator.
ICPHM	International Conference on Prognostics and Health Management.
MAE	Mean absolute error.
MR	Set maximum RUL value; strategy to handle large RUL estimates, described in Section IV.
PCA	Principal component analysis.
PDF	Probability density function.
PHM	Prognostics and health management.
RMS	Root mean square.
RMSE	Root mean squared error.
RUL	Remaining useful life.

I. INTRODUCTION

With increasing digitization of production processes and the introduction of Industry 4.0 into factories all over the world, access to additional economic profit by further automating and flexibilizing the industrial processes is expected [1], [2]. Especially the topic of prognostics and health management (PHM), which forms the basis of deploying predictive maintenance strategies [3], is a prominent use case of Industry 4.0, since it offers the avoidance of unnecessary and unplanned failures of industrial assets. This leads to higher

The associate editor coordinating the review of this manuscript and approving it for publication was Zhaojun Li.

equipment efficiencies, less downtime and lower costs due to disturbances in global supply chains [4]. However, in order to take advantage of the mentioned benefits, access to the current degradation state, the future degradation state and the resulting remaining useful life (RUL) is necessary [5]. Hence, the task of RUL estimation is a crucial part in the topic of PHM and the Industry 4.0. In the literature, many authors have addressed the task of RUL estimation in many different ways, which can be categorized into physics-based approaches, statistical approaches and artificial intelligence (AI) approaches [6]. Especially the latter two have been subject of extensive investigations within the last few years. This is due to the accessibility of computing power, the availability of large, simulated benchmark data sets and ongoing algorithmic advances [7]. Many of the presented approaches in the literature achieve great results in cases where data sets containing many run-to-failure sequences for the monitored components are available [8]–[10]. In real use cases, however, the entire run-to-failure sequence is rarely available. Even in cases, where data of the monitored components has been recorded for entire life cycles, usually only few run-to-failure sequences are available. This leads to the problem of estimating the RUL based on little historic data and potentially the total absence of entire run-to-failure sequences [7].

This work addresses this issue by presenting a new approach, which is based on [11] and which combines Gaussian process classification (GPC) and Gaussian process regression (GPR) to estimate the RUL. It will be shown, that this new approach achieves good results compared to the state of the art on standard benchmark data sets needing only a fraction of the historic data provided. The approach will be evaluated based on the C-MAPSS turbo fan engine data set [12] and will be transferred to the FEMTO data set [13]. Summarizing, the following contributions will be made:

1) A new, data-efficient approach, based on a data-efficient health indicator (HI) for the task of RUL estimation will be presented.
2) The presented approach will be applied to the C-MAPSS data set with only two training instances in order to simulate training data scarcity, which is often present in real use cases.
3) The resulting RUL estimation accuracy will be compared to state of the art results, which rely on all available training data sequences.
4) The approach will be applied to the FEMTO data set in order to demonstrate its transferability.

The rest of the article is structured as follows: In Section II, an overview of the state of the art for RUL estimation is given. In Section III, the theoretical background is presented. The proposed, data-efficient approach for RUL estimation is presented in Section IV. The approach was applied to the two benchmark data sets C-MAPSS and FEMTO, which is described in detail in Section V. The results are presented in Section VI. In Section VII the work is summarized and an outlook on future work is given.

II. RELATED WORK

The literature on RUL estimation is vastly diverse and has been rapidly growing within the last few years. Lei *et al.* conducted an extensive literature research and proposed to categorize the approaches into the following four categories: physics-based, statistical-based, AI-based and hybrid approaches [6]. The latter three are referred to as *data-driven* approaches in this work.

Often, the true physical wear process is too complex to be modeled directly, which is the reason for physics-based approaches being less common in the literature [6]. On the contrary, data-driven approaches, which exploit correlations between sensor signals and wear or failure, do not model physical principals directly and, therefore, are more applicable in many cases. The data-driven approaches for RUL estimation can be further divided into direct and indirect approaches [14]. In direct approaches a given sequence of sensor signals is directly mapped to an RUL value, whereas indirect approaches estimate the current health, expressed by an HI, and extrapolate the HI value up until exceeding a preset threshold, at which the end-of-life time is defined [15]. By subtracting the time of the last measurement from the defined end-of-life time, the RUL estimate is formed.

Due to advances in methods of AI, direct approaches have been flourishing in the literature, recently. Especially the application of deep learning (DL) models with architectures of all kinds achieved impressive RUL estimation performances for various applications and often outperformed the state of the art on established benchmark data sets [8]. Chinomona *et al.*, for example, applied long short-term memory neural networks to the problem of battery RUL estimation [16], Sun *et al.* applied auto-encoder neural networks to predict the RUL of cutting tools [17] and Yang *et al.* applied convolutional neural networks to the task of bearing RUL prediction [18]. Recent approaches tackle the issue of uncertainty quantification and utilization in DL applications to RUL prediction by applying Bayesian neural networks [9]. Although all the mentioned works demonstrated high accuracies in RUL prediction, they all have in common, that they need large, representative training data sets, which are often not available in real industrial applications [7]. This issue was addressed by Lv *et al.*, who proposed a so called sequence adaption adversarial network, that yields good results on small data sets [19]. Another example for an approach, which tries to achieve good RUL estimates with little data was presented by Zhang *et al.*, who introduced a transfer learning approach [20]. Their proposed neural network was trained on a source task with large amount of data and fine tuned on a target task, in which only little data was available. Although their source task and target task data sets were both from the C-MAPSS data set, they demonstrated, that transfer learning can enhance RUL prediction performance. However, both discussed approaches still rely on RUL labels and, ultimately on run-to-failure sequences, that are very rarely present in real use cases.

Indirect approaches to RUL estimation include the approach proposed by Nguyen and Medjaher, who optimized the construction of an abstract HI with the help of a genetic algorithm and extrapolated this HI with different machine learning regression models [21]. Their experiments showed good results on the C-MAPSS and FEMTO data sets. However, the authors did not provide details about their model implementations and the proposed approach relied on a large number of training observations. Another recent indirect approach was presented by Wen *et al.*, who also constructed a composite HI with a genetic algorithm and extrapolated it with a regression model, that followed a power law [22]. The advantage of their regression model is that, due to the power law, it can represent accelerated, linear and decelerated trends. All of those models have a monotonic behavior, as has degradation and wear. Their experiments with the C-MAPSS data set demonstrated the usefulness of the constructed HI and the resulting high accuracy in RUL estimation. However, the presented results relied on all available training units of the C-MAPSS data set. Another notable approach was presented by Li *et al.*, who aimed at estimating the RUL for machine tool ball screws on a feed drive test bench [23]. After identifying relevant features from time and frequency domain data, they constructed an HI via linear regression and extrapolated this HI with a GPR model, in order to estimate the RUL. The covariance function they applied for the GPR model was the *radial basis function* kernel (see Appendix A-A), which is implicitly assuming stationary data. A wear sensitive HI, however, is non-stationary, i.e. the expected value of the HI changes over time with degradation. Nevertheless, they outlined an approach based on only nine training observations and showed the applicability on a real world data set. Benker *et al.* built on the general idea of Li *et al.* and presented a modified concept, which first adapted a GPC model for HI construction bounding the HI values on the interval [0], [1]. Second, the HI values were extrapolated with a GPR model including a non-stationary covariance function [11]. Due to the non-stationary nature of the GPR model, decreasing HI values could be better captured and extrapolated. In contrast to using a non-stationary covariance function, Liu and Chen proposed to construct a GPR model with a linear mean function and a stationary covariance function for extrapolating a HI for the use case of battery capacity prediction [24]. Due to the linear mean function, their model can capture non-stationary data, such as a decrease in capacity. Finally, Aye and Heyns investigated GPR models, which combined different mean functions and stationary, as well as non-stationary, covariance functions [25]. On their data set from a bearing degradation test bench, the best results were performed with stationary covariance functions in combination with a linear mean function. All above mentioned GPR models did not consider monotonicity or boundedness of the HI values, however.

This work aims at addressing the shortcomings in the state of the art by proposing a novel approach, which can be trained with only a small amount of healthy and degraded observations, making run-to-failure sequences needless, and which applies a warped, monotonic GPR model to extrapolate sensible HI values within the interval [0], [1] in order to accurately estimate RUL.

III. THEORETICAL BACKGROUND

A. GAUSSIAN PROCESSES

A Gaussian process (GP), also referred to as *prior over functions* [26], is defined as a collection of random variables, any finite subset of which have a joint Gaussian distribution, where the random variables are function values $f(\mathbf{x})$ at input locations $\mathbf{x} = \left[x_1 \; x_2 \; \cdots \; x_d\right]^\top$ [27]. Any GP is completely specified by a mean function

$$m(\mathbf{x}) = \mathbb{E}\left[f(\mathbf{x})\right], \tag{1}$$

and a covariance function

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \mathbb{E}\left[\left(f(\mathbf{x}) - m(\mathbf{x})\right)\left(f(\mathbf{x}') - m(\mathbf{x}')\right)\right]. \tag{2}$$

A function following a GP can formally be denoted as $f(\mathbf{x}) \sim \mathcal{GP}\left(m(\mathbf{x}), k\left(\mathbf{x}, \mathbf{x}'\right)\right)$. Usually, the mean function $m(\mathbf{x})$ is set to zero. This is adapted in this paper as well and subsequently $f(\mathbf{x}) \sim \mathcal{GP}\left(\mathbf{0}, k\left(\mathbf{x}, \mathbf{x}'\right)\right)$. In order to sample from the random function $f(\mathbf{x})$, it can be discretized and evaluated at multiple input locations $X = \{\mathbf{x}\}_{i=1}^{N}$. This leads to a random vector $\mathbf{f}$, following a multivariate normal distribution with the mean vector being zero and the covariance matrix being $K\left(X, X'\right)$, i.e.

$$\mathbf{f} \sim \mathcal{N}\left(\mathbf{0}, K\left(X, X'\right)\right). \tag{3}$$

Exemplary sampling from different GP priors, yielding random vectors $\mathbf{f}$ with mean zero and different covariance functions is shown in Fig. 1. It can be seen, that different covariance functions impose different structures on the modeled function $f(\mathbf{x})$ even before data is observed, which allows the incorporation of prior knowledge (see Appendix A-A for details).

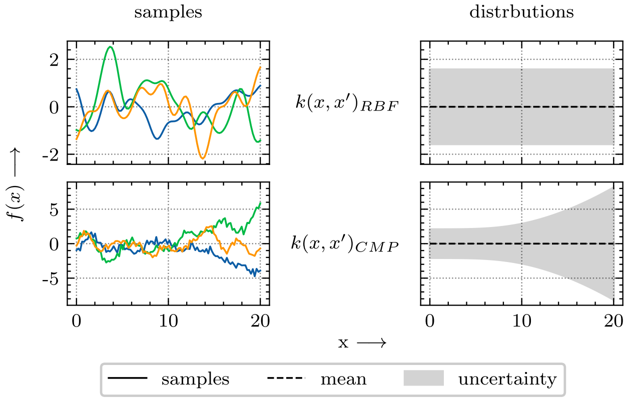


FIGURE 1. Prior distributions over functions $f(\mathbf{x}) \sim \mathcal{GP}\left(0, k\left(\mathbf{x}, \mathbf{x}'\right)\right)$ for a stationary radial basis covariance function $k\left(\mathbf{x}, \mathbf{x}'\right)_{RBF}$ and a non-stationary, composite covariance function $k\left(\mathbf{x}, \mathbf{x}'\right)_{CMP}$ (see Appendix A-A for details); it can be seen, that the covariance functions impose a certain structure on the function $f(\mathbf{x})$ before (i.e. prior to) observing any data. The grey shaded area represents 1.96 standard deviations from the mean.

B. GAUSSIAN PROCESS REGRESSION

Assuming, that a historic data set $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$ consisting of input variables $\mathbf{x}_i$ and noisy target variables $y_i = f(\mathbf{x}_i) + \epsilon_i$, where $\epsilon_i \sim i.i.d.\ \mathcal{N}\left(0, \sigma_n^2\right)$, is available, a joint distribution of observed target values $\mathbf{y} = \{y\}_{i=1}^{N}$ and unobserved target values $\mathbf{f}^\star$ at new input locations $X^\star$ can be denoted as

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^\star \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K\left(X, X^\star\right) \\ K\left(X^\star, X\right) & K\left(X^\star, X^\star\right) \end{bmatrix}\right), \quad (4)$$

with I being the identity matrix.

The posterior (i.e. predictive) distribution for new targets $\mathbf{f}^\star$ based on a new input locations $X^\star$, historic inputs X and historic targets $\mathbf{y}$ is then given by

$$\begin{aligned} p\left(\mathbf{f}^\star | X, \mathbf{y}, X^\star\right) &\sim \mathcal{N}\left(\bar{\mathbf{f}}^\star, \operatorname{cov}\left(\mathbf{f}^\star\right)\right), \text{ with} \\ \bar{\mathbf{f}}^\star &= K\left(X^\star, X\right)\left[K(X,X) + \sigma_n^2 I\right]^{-1} \mathbf{y}, \\ \operatorname{cov}\left(\mathbf{f}^\star\right) &= K\left(X^\star, X^\star\right) - K\left(X^\star, X\right) \\ &\quad \times \left[K(X,X) + \sigma_n^2 I\right]^{-1} K\left(X, X^\star\right). \end{aligned} \quad (5)$$

A detailed derivation of this solution can be found in [27]. An exemplary predictive distribution for different covariance functions is shown in Fig. 2. It can be seen, that the different covariance functions yield different results for the predictive distribution $p\left(\mathbf{f}^\star | X, \mathbf{y}, X^\star\right)$. Hence, a careful design of a suitable covariance function for the problem at hand can be an decisive step in modeling with Gaussian processes.

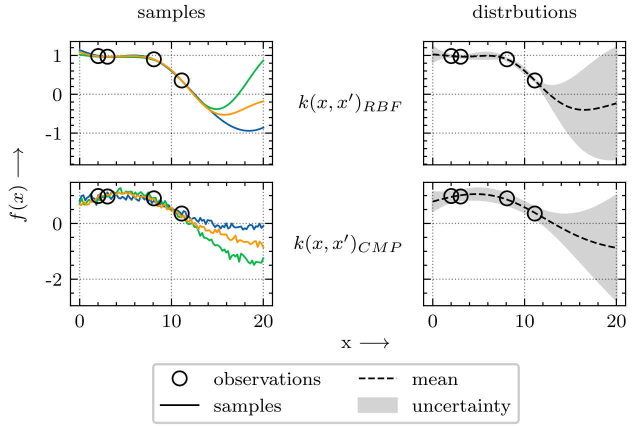


FIGURE 2. Posterior (i.e. predictive) distribution $p\left(\mathbf{y}^\star | X, \mathbf{y}, X^\star\right)$ after observing noisy data for a GP with a stationary radial basis covariance function $k\left(\mathbf{x}, \mathbf{x}'\right)_{RBF}$ and a GP with a non-stationary, composite covariance function $k\left(\mathbf{x}, \mathbf{x}'\right)_{CMP}$ (see Appendix A-A for details); the grey shaded area represents 1.96 standard deviations from the mean.

1) MONOTONICITY IN GAUSSIAN PROCESSES REGRESSION

In cases, where the function, which is about to be modeled, is known to be monotonic, imposing monotonicity is another way of introducing prior knowledge. This is a difficult task, since for GPs it implies that all values are correlated with each other [28]. The common covariance functions, such as the ones used in this work, define the correlation between function values only based on a distance between the inputs $\mathbf{x}$ (see Appendix A-A). The further the values are apart from each other the less strong they correlate up to a point, where correlation becomes negligible. One way of dealing with this difficulty is to enforce monotonicity on only a finite number of inputs. Riihimäki and Vehtari developed a method for incorporating monotonicity information in GPR by inducing virtual derivative observations. Since the derivatives of a GP are GPs themselves, it is possible to include them into the GPR model, and define them to be non-negative at the inducing input locations. Details can be found in Appendix A-C. For this work, the Matlab® implementation of this method, provided by Vanhatalo *et al.*, was used [29].

2) WARPING FOR GAUSSIAN PROCESSES REGRESSION

In this work the extrapolation of future HI values should predict values within the range [0], [1], which leads to the problem of *bounded regression*. As the GP posterior distribution $p\left(\mathbf{f}^\star | X, \mathbf{y}, X^\star\right)$ has infinite support, it is unbounded in general. One way to bound the output space of a GPR model is *warping*, which was originally proposed by Snelson *et al.* [30]. It transforms the model output by a so called *warping function*. Although Snelson *et al.* proposed to learn the warping function automatically, a Gaussian cumulative function centered at 0.5 was applied in this work, since this is known to produce good results, as well [31].

C. BINARY GAUSSIAN PROCESS CLASSIFICATION

GPs can also be used to perform classification tasks. However, the evaluation of the predictive distribution is more challenging than in the regression setting described in Section III-B. In a binary (i.e. two-class) classification task, where the predictions give class probabilities ranging from 0 to 1, the idea is to predict the latent value $f^\star$ with Eq. (5) and then map the results onto [0], [1], using a so called *squashing function* $\sigma\left(f^\star\right)$. Two examples for squashing functions are the logit and probit functions. The probability $\pi^\star$ can then be computed as

$$\begin{aligned} \pi^\star &= p\left(y^\star = 1 \,| X, \mathbf{y}, X^\star\right) \\ &= \int \sigma\left(f^\star\right) p\left(f^\star | X, \mathbf{y}, X^\star\right) df^\star. \end{aligned} \quad (6)$$

Unfortunately, the expression in Eq. (5) cannot be evaluated analytically anymore in the classification case. In order to tackle this problem, one has to resort to analytic approximations or numerical approaches. A suitable approximation was found to be expectation propagation (EP) algorithm [32], which was first proposed by Minka [33] and which is described in further detail in Appendix A-B.

IV. PROPOSED APPROACH

The proposed approach is based on two steps, which are depicted in Fig. 3. First, a *model building* step, in which a binary GPC model is trained on historic training data consisting of observations of healthy and degraded states of a system, i.e. the entire run-to-failure sequence, is not needed.

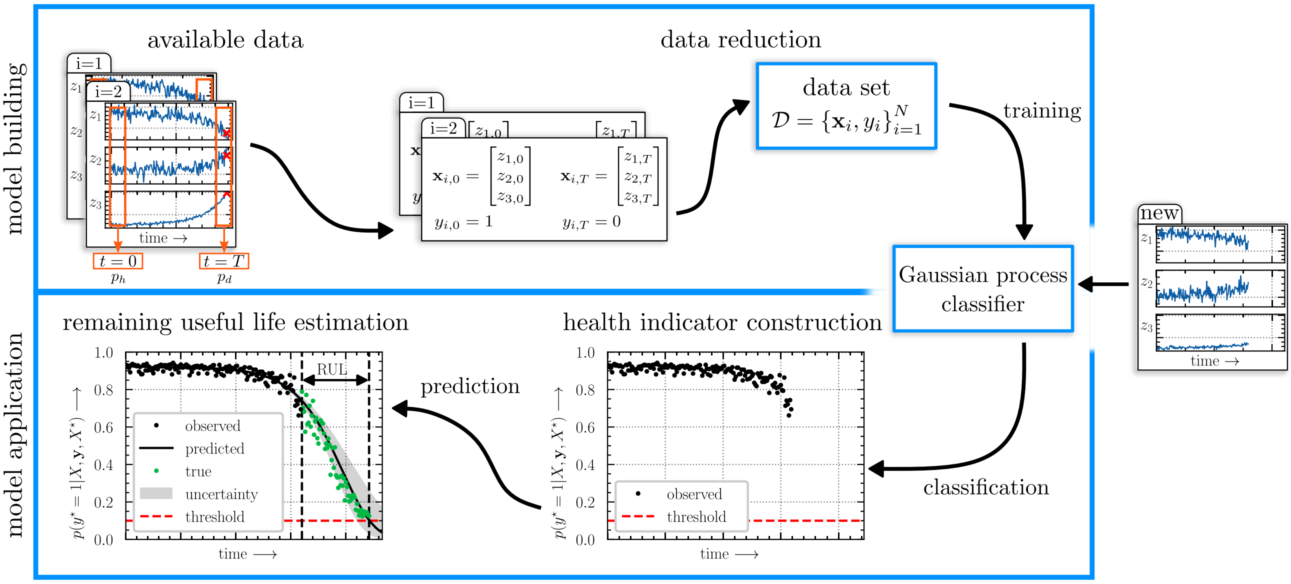


FIGURE 3. **Illustration of the proposed approach.**

Instead, only a small percentage of the first few healthy observations at the beginning of the life time (referred to as hyper-parameter p_h) and a small percentage of the last few degraded observations at the end of the life time (referred to as hyper-parameter p_d) are needed. For real use cases, this data is easier to get than entire run-to-failure sequences, since one does not necessarily need to wait until the end-of-life, but can instead artificially induce wear and make measurements. The GPC model was chosen, since it outputs a probability of class membership, i.e. a probability of a given data point belonging to a healthy or degraded state. This probability from Eq. (15) serves as an HI, which is always within the range [0], [1].

This HI is extrapolated in the second step, the *model application* step. Here, a new, unobserved sequence is handed over to the trained GPC model, which translates the sequence of sensor signals into a sequence of HI values. Based on those HI values, a warped, monotonic GPR model is applied to forecast the future progress of the HI values. The end-of-life time is defined at the point in time, where the forecasted HI values fall below a defined threshold, indicating that the system state is degraded with a critically high probability.

In some situations, where the HI values do not exhibit a trend yet, the GPR model does not forecast falling HI values and, therefore, forecasts an RUL value, which is obviously too large and not within a sensible range. This is often the case at the beginning of the life time of an investigated instance. In a real use case, where maintenance actions must be planned in advance, this is unacceptable. For this situation the two following strategies are introduced:

- Discard large RUL estimates (DI): This is a naive strategy, which simply discards the RUL estimates that are too large. It has to be noted, that this is only applicable in practice if one does not rely entirely on a predictive maintenance strategy, which requires RUL estimates, but instead can fall back on a planned maintenance schedule.
- Set a maximum RUL value (MR): This strategy limits the maximum RUL estimates to a defined value. In case the proposed approach generates estimates higher than this defined maximum RUL value, those estimates are set equal to the maximum RUL value. In contrast to the DI strategy, no RUL estimate is discarded enabling the application of an entirely predictive maintenance strategy.

In has to be noted, that both strategies imply a maximum RUL value in principal. The strategy DI discards RUL estimates that are larger than a maximum RUL value and the strategy MR limits the RUL value to the maximum RUL value. Hence, both strategies assume, that a sensible maximum RUL value can be set either based on experience or based on established life time calculations such as [34]. Both strategies, DI and MR, were evaluated within the experiments described in the next section.

V. EXPERIMENTS

The proposed approach from Section IV was applied to the two data sets C-MAPSS and FEMTO. In the following subsections the data sets and implementation details of the conducted experiments are described.

A. C-MAPSS DATA SET

The C-MAPSS data set, published by [12], consists of four subsets, each subset being composed of a training and test data set of simulated run-to-failure sequences of turbo fan engines. In each sequence, 21 equally spaced time series of different sensors are recorded. The subsets are varying in the number of training sequences, test sequences, fault modes and operational conditions (see (see Table 1). The training data set consists of entire run-to-failure sequences, whereas the test

TABLE 1. Overview of the different C-MAPSS subsets.

	subset			
	FD001	FD002	FD003	FD004
training data sequences	100	260	100	249
test data sequences	100	259	100	248
operating conditions	1	6	1	6
fault modes	1	1	2	2

TABLE 2. Chosen hyper-parameters for the presented C-MAPSS and FEMTO experiments.

	C-MAPSS	FEMTO		
hyper-parameter	all subsets	C1	C2	C3
p_h	7.5 %	6.5 %	7.5 %	7 %
p_d	2 %	1.3 %	1.7 %	3 %
threshold	0.1	0.1	0.1	0.1
training instances	2	2	2	2

data set consists of interrupted run-to-failure sequences and the associated true RUL values.

Since the proposed approach builds on a GPC model, which does not need the entire run-to-failure sequence as an input, the C-MAPSS training data had to be modified, first. The absence of the entire run-to-failure sequence was simulated by reducing the training data: only a small percentage of the sequence of healthy observations at the beginning of the training run-to-failure sequence and a small percentage of the sequence of faulty observations at the end of the training run-to-failure sequence were used (see Table 2). In order to simulate data scarcity, two random engines from the training data set were selected for each of the four subsets. In case of the FD001 subset, this implied that only two out of one hundred instances (i.e. engines) were used for training. As a consequence, a data set suitable for binary classification was generated. This data set was scaled to mean zero and unit variance. Based on this data set, the GPC model was trained. After training, the unobserved sequences from the test data set served as input for the trained GPC model, which predicted the probability of class membership for each single observation within the test sequence up to the latest measurement, resulting in a sequence of HI values over time. Based on this sequence of HI values, a warped, monotonic GPR model was applied to forecast the future progress of the HI values. The end-of-life time was defined at the point in time, where the forecasted values fell below a threshold of 0.10. The RUL was calculated as the difference between the end-of-life time and the time of the latest measurement. Details concerning the implemented models are given in Appendix B. In case of applying the MR strategy for too large RUL estimates, the maximum RUL was set to 125 cycles, which is in accordance with [10] and [9]. In order to account for statistical fluctuations in the results, the experiments were conducted ten times, each time selecting two different random engines for training the GPC model.

B. FEMTO DATA SET

In contrast to the C-MAPSS data set, which provides simulated run-to-failure sequences, the FEMTO data set consists of run-to-failure sequences that were recorded on a real test bench, the so called PRONOSTIA test bench, developed at the FEMTO-ST Institute in Besano̧n, France [13]. On this test bench, rolling bearings were run to failure without inducing errors beforehand and therefore, generating realistic run-to-failure sequences for bearings. The bearings were clamped on a bearing support shaft, which was actuated by an electrical motor. On the outer ring of the bearings a radial force was induced by a hydraulic actuator. The run-to-failure experiments were performed in three different operating conditions (see Table 3). During the experiments, acceleration and temperature was measured by two accelerometers and a thermocouple. The failure threshold was set to the acceleration exceeding 20 g. The acceleration was measured every 10 s in 0.1 s snippets with a sampling rate of 25.6 kHz. Temperature measurements were conducted continuously with a sampling rate of 10 Hz. However, the temperature measurements are only available for one condition and, therefore, were neglected for all conditions in this work. For each condition two run-to-failure sequences are available. For conditions one and two, five sequences and for condition three, one sequence is available for testing. Similarly to the C-MAPSS data set, only the first few (i.e. p_h percent) and last few (i.e. p_d percent) observations of the two training run-to-failure sequences were selected for training (see table Table 2). In case of the FEMTO data set, further pre-processing of the data was necessary, since in contrast to the C-MAPSS data set, the raw data is not composed of degradation sensitive features. Three kinds of features were extracted. First, statistical features from the time and the frequency domain were calculated (see Table 7 in Appendix C). The features were transformed via PCA and the first six principal components were selected, which showed to be a sensible trade-off between dimensional reduction and information loss, explaining approximately 90 % of the variance of the original data. Second, the feature extraction approach from Sutrisno *et al.* [35] was adopted, which generates moving average values of the kurtosis of the frequency band from 5.5 kHz to 6 kHz. Third, the feature extraction method proposed by Kim *et al.* [36], which computes entropy values of specific, normalized energy spectrum bands, was implemented. Since only two training run-to-failure sequences are available for each condition, the experiments were only performed once.

TABLE 3. Overview of the different FEMTO operating conditions.

condition	load	speed
C1	4 000 N	1 800 rpm
C2	4 200 N	1 650 rpm
C3	5 000 N	1 500 rpm

C. PERFORMANCE METRICS

The resulting RUL estimates $y^{\star}_{RUL}$ of the proposed approach were judged by their deviation from the true RUL values y_{RUL} with respect to four performance metrics, which are also reported in the related work presented in Section II and, therefore, allow a comparison. For the results of the experiments with the C-MAPSS data set, the deviation of the i^{th} test prediction is denoted as $\tau_i = y^{\star}_{RUL,i} - y_{RUL,i}$. First, the root mean squared error (RMSE) for each subset in the C-MAPSS test data set was computed:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{M} (\tau_i)^2}, \tag{7}$$

with M being the number of test predictions. In addition, the mean absolute error (MAE) was computed:

$$\text{MAE} = \sum_{i=1}^{M} |\tau_i| \,. \tag{8}$$

For the C-MAPSS data set, an asymmetric score function was proposed by [12], which punishes late predictions stonger than earlier predictions:

$$s(\tau_i) = \begin{cases} s_1(\tau_i) = e^{-\frac{\tau_i}{13}} - 1, & \text{for } \tau_i < 0. \\ s_2(\tau_i) = e^{\frac{\tau_i}{10}} - 1, & \text{for } \tau_i \geq 0. \end{cases} \tag{9}$$

The rationale behind this scoring function is, that late predictions and the associated unplanned failure of an instance lead to higher costs than too early replacements. The lower, the score value is, the better is the prediction.

For the experiments with the FEMTO data set a relative error, which was originally proposed by [13], and which is calculated as

$$\text{error} = 100 \times \frac{y_{RUL} - y^{\star}_{RUL}}{y_{RUL}}, \tag{10}$$

was calculated in order to assess the performance of the proposed approach and relate it to a benchmark approach.

VI. RESULTS

In this section the results of the experiments, described in Sections V-A and V-B, are presented. The section is structured according to the contributions declared in Section I.

A. DATA-EFFICIENCY OF THE PROPOSED HEALTH INDICATOR

The first objective of this work is to present a data-efficient HI. For the evaluation of the achievement concerning this objective, three different test instances from the C-MAPSS data set, and their respective estimated HIs are shown in Fig. 4. Training was conducted with only two training instances. In addition to the HI produced by the GPC model, results produced by simple linear regression and logistic regression models are shown. It can be seen, that the resulting HIs produced by the GPC model are much steadier than the HIs produced by linear regression and logistic regression models. This is important for the second step of the proposed approach, where the estimated HI values are extrapolated in order to estimate the RUL. Actually, the HIs of the linear regression models are very noisy for engines number 16 and number 30. A sensible extrapolation of those noisy HI values is hardly possible. Another issue of using a linear regression model for estimating HIs becomes visible for engine number 52. Here, the HI estimates at the beginning of the life cycle are higher than one and the HI estimates at the end of the life cycle are lower than zero. This is an unwanted behavior, since for such HIs, defining end-of-life time thresholds is considerably harder than for HIs that are always within the range [0], [1]. The HI estimated by the logistic regression model, in contrast, is within the range [0], [1] by definition, as the logistic regression model outputs a probability of class membership. However, it can be observed, that the HIs produced by the logistic regression model are almost constant for a long proportion of the life cycle and drop abruptly when the end-of-life time is near. This can be seen especially in the case of engine number 52. This behavior is undesirable as well, since an extrapolation of a suddenly dropping HI is hard and can lead to inaccurate RUL estimates. However, the HI values estimated by the GPC model are convincing, since they are within the range [0], [1] by definition and they are neither too noisy, nor do they drop too fast. Hence, they are expected to be suitable for extrapolation.

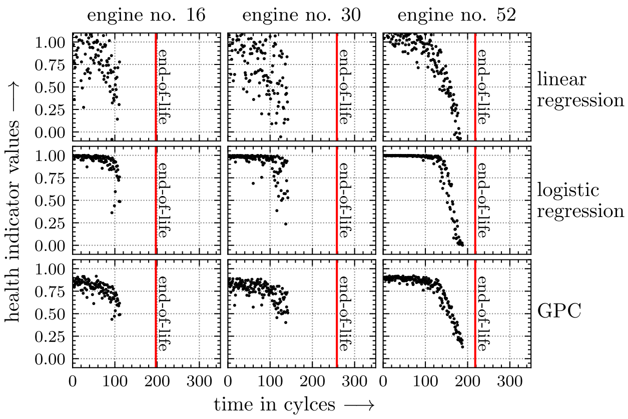


FIGURE 4. Exemplary HI sequences generated by the fitted GPC model for three engines of the C-MAPSS FD001 test subset; training was conducted with two training instances.

B. APPLICATION TO THE C-MAPSS DATA SET

The second objective of this work is to present an approach which can accurately estimate the RUL of a new, unseen instance, based on the data-efficient HI provided by the GPC model. In Fig. 5, exemplary extrapolations and the true end-of-life time values are shown, again for the three exemplary test engines 16, 30 and 52. First of all, it can be seen, that the warped, monotonic GPR model's mean prediction, shown as a black line, is in accordance with the observed HI values and their trend. Furthermore it can be seen, that the threshold set at 0.1 and the subsequently resulting end-of-life time estimate match the true end-of-life time almost exactly for the engines 16 and 30. For engine 52, the RUL estimate is not as accurate, since the observed HI values

TABLE 4. Summary table of results for the C-MAPSS data set; results show the summary statistics mean and variance of all results produced by ten experiments.

		FD001		FD002		FD003		FD004	
strategy	loss	mean	std	mean	std	mean	std	mean	std
DI	RMSE	62.89	1222.79	106.75	5917.37	73.80	2335.54	109.38	4133.06
	MAE	29.51	5.04	48.73	22.32	33.53	7.02	52.55	9.85
	score	$1.47 \cdot 10^6$	$4.20 \cdot 10^6$	$6.36 \cdot 10^9$	$1.85 \cdot 10^{10}$	$9.10 \cdot 10^7$	$1.85 \cdot 10^8$	$5.01 \cdot 10^8$	$7.36 \cdot 10^8$
MR	RMSE	32.61	219.00	68.54	1434.94	49.89	1013.54	63.93	317.72
	MAE	16.06	1.16	35.19	4.31	24.81	5.82	33.45	1.82
	score	$1.25 \cdot 10^2$	$3.18 \cdot 10^3$	$7.01 \cdot 10^4$	$1.78 \cdot 10^5$	$3.51 \cdot 10^3$	$3.71 \cdot 10^3$	$1.09 \cdot 10^4$	$2.7 \cdot 10^3$

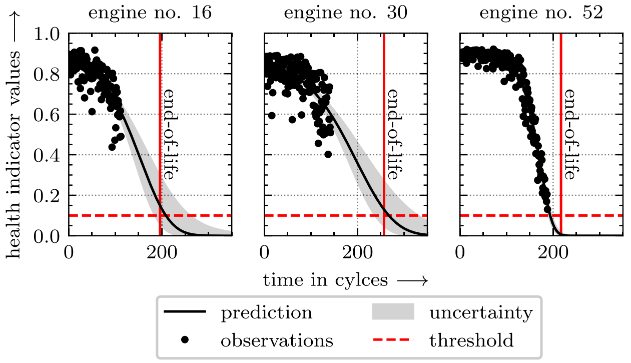


FIGURE 5. Exemplary RUL estimation for the C-MAPSS FD001 subset trained on two instances; the grey area represents 1.96 standard deviations from the mean.

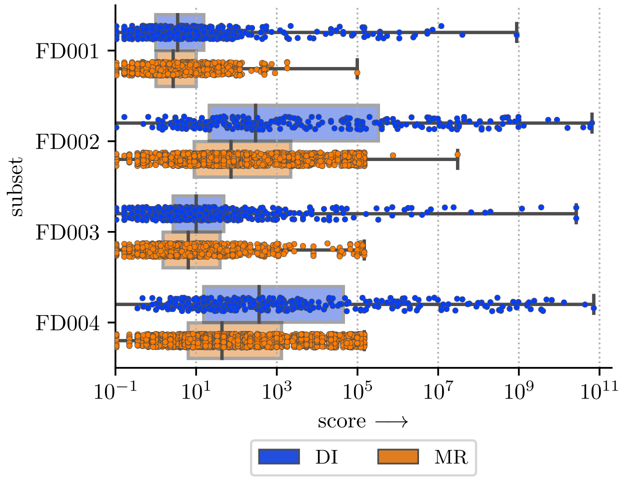


FIGURE 6. C-MAPSS results of all ten experimental runs (X-axis displayed as log scale.)

are already close to the threshold. From this example it can be seen, that the accuracy of the RUL estimate is mainly determined by the GPC model, which generates the HI values. The results for the entire C-MAPSS test data, including all subsets, are shown in Table 4. There, the results for ten runs of the experiments are reported. It becomes visible, that the mean performances differ from subset to subset, which is in line with expectations, due to the different fault modes and operating conditions. It is noticeable, that the MR strategy for handling large RUL estimates is always outperforming the DI strategy. Furthermore, the standard deviations are lower for the MR strategy. However, the standard deviations remain rather high, such that a closer look into the distribution of the performances of the single test instances is necessary. In Fig. 6, all prediction results for the performance metric score are shown for all ten runs. It can be observed, that few very poor performing predictions (i.e. high scores) are present and corrupt the reported mean and standard deviations. However, the interquartile ranges of the distributions of the scores for the FD001 and FD003 subsets, indicated by the boxplots, show both, good scores and low distance from the median score. For the FD002 and FD004 subsets, the performance is worse. Here, the distributions are not as compact around the median, which itself is at higher (i.e. worse) scores.

C. COMPARISON WITH STATE OF THE ART METHODS

The third objective of this work is the comparison to the performance reported in the state of the art. Recent work by Jiang *et al.* [37] aggregated some notable approaches from the literature and their performances. A selection of those is shown in Table 5 for convenience. Compared to the state of the art results, the achieved mean performance in terms of RMSE and score of the proposed approach is comparably good. In the case of the FD001 subset it is even outperforming some of the DL approaches. Considering that the proposed approach only uses two training instances compared to 100 training instances used by the state of the art methods, the fact that comparable performances can be achieved is remarkable. This is even more the case when considering, that no run-to-failure data was used.

D. TRANSFER TO THE FEMTO DATA SET

The fourth objective of this work was to apply the proposed approach to the FEMTO data set and comparing the achieved performance to the state of the art. This is reported in Table 6. There it can be seen, that the performance is comparable to the benchmark work from Sutrisno *et al.* [35], who were one of the winning teams of the International Conference on Prognostics and Health Management (ICPHM) 2012 FEMTO data challenge. It has to be noted, that the predictive performance on the FEMTO data set is not as remarkable as it is on the C-MAPSS data set for both, the proposed approach and the benchmark approaches from the state of the art. This is due to

TABLE 5. Selection of state of the art results for the C-MAPSS data set based on [37].

	FD001		FD002		FD003		FD004	
methods	RMSE	score	RMSE	score	RMSE	score	RMSE	score
SVM [38]	40.72	$7.70 \cdot 10^3$	52.99	$3.16 \cdot 10^5$	4.63	$2.25 \cdot 10^4$	59.96	$1.41 \cdot 10^5$
D-LSTM [39]	16.14	$3.38 \cdot 10^2$	24.49	$0.45 \cdot 10^4$	16.18	$0.28 \cdot 10^3$	23.31	$1.25 \cdot 10^4$
DCNN [10]	12.61	$2.74 \cdot 10^2$	22.36	$1.04 \cdot 10^4$	12.64	$0.28 \cdot 10^3$	23.31	$1.25 \cdot 10^4$
TCNA [37]	10.45	$2.29 \cdot 10^2$	20.15	$0.58 \cdot 10^4$	9.60	$0.24 \cdot 10^3$	22.09	$0.51 \cdot 10^4$
proposed approach	32.61	$1.25 \cdot 10^2$	68.54	$7.01 \cdot 10^4$	49.89	$3.51 \cdot 10^3$	63.93	$1.09 \cdot 10^4$

TABLE 6. Summary table of results for the FEMTO data set.

condition	bearing	error in %	benchmark error in % [35]
1	3	83.16	37
	4	-44.21	80
	5	91.99	9
	6	91.48	-5
	7	73.77	-2
2	3	97.21	64
	4	10.46	10
	5	97.54	-440
	6	0.87	49
	7	-98.73	-317
3	3	86.26	90
mean absolute error in %		70.52	100.27
mean error in %		44.53	-38.64

the complexity of the raw data of the FEMTO data set and the resulting difficulty to extract failure sensitive features. Nevertheless, the proposed approach achieves comparable results with only a fraction of the data the benchmark approach consumes.

VII. DISCUSSION AND OUTLOOK

In this work, a novel Gaussian process based approach for data-efficient RUL estimation was presented. The approach is estimating the RUL indirectly by first constructing a GPC model to translate a machinery's sensor signal into an HI value. In a second step, this HI value is extrapolated with the help of a warped, monotonic GPR model. The time the extrapolated HI values surpass a preset threshold is defined to be the estimated end-of-life time. Based on this, the estimated RUL can be calculated. The approach was applied to the well known benchmark data set C-MAPSS. In addition, the poposed approach was applied to the FEMTO data set. The experiments yielded the following insights:

1) In use cases where the operating conditions are not relevant for correlating a signal to wear, the approach yields good results compared with the state of the art, as it was shown for the C-MAPSS FD001 subset.
2) In use cases with different fault modes, the approach also yields comparatively good results, as well, as it was shown for the C-MAPSS FD003 subset.
3) When operating conditions are not controlled but implicitly recorded in the sensor signals, the approach fails to extract those operating conditions automatically, leading to bad RUL estimates, as it was shown with the C-MAPSS FD002 and FD004 subsets. Hence, further investigations for cases, in which operating conditions are known and can be controlled, are needed. One approach to account for operating conditions for the C-MAPSS data set was presented by Wang *et al.* [40].
4) The approach can be also applied to a real world data set, which was exemplarily shown with the FEMTO data set. Here, the major constraint of the approach became visible: the proposed approach relies on signals, which are sensitive to wear and failure. This is the case in the C-MAPSS data set, which explains the good performance on that data set. For the FEMTO data set, however, this is not the case. The extraction of reliable, failure sensitive features has not been achieved satisfactorily with the adapted pre-processing techniques from the state of the art, which ultimately leads to worse performances on this data set.

Summarizing, the proposed approach adds value to the research in RUL estimation as it enables accurate predictions in use cases, where wear and failure sensitive features are known, operating conditions can be controlled but historic run-to-failure data is absent except for single observations of healthy and faulty states.

Future improvements of the proposed approach should deal with the automatic extraction of wear and failure sensitive features (as exemplarily shown by Michau *et al.* [41]), the incorporation of controlled operating conditions into the GPC and GPR model, optimizing the hyper-parameters of the approach and transferring the approach to other real world data sets.

APPENDIX A
ADDITIONAL DETAILS ON GAUSSIAN PROCESSES

A. COVARIANCE FUNCTIONS

In this appendix, the different covariance functions used in this work are formally denoted. All formal definitions are based on [27]. First, the *radial basis function* (RBF) covariance function was used in order to model smooth, stationary functions. Formally it is defined as

$$k\left(\mathbf{x}, \mathbf{x}'\right)_{RBF} = \sigma^2 \exp\left[-\frac{\left(\mathbf{x} - \mathbf{x}'\right)^2}{2\ell^2}\right], \tag{11}$$

with the lengthscale parameter ℓ and the magnitude parameter σ^2. Second, a non-stationary covariance function designed for remaining useful life estimation and presented in [11] was used for the Figs. 1 and 2. It is defined as

$$k\left(\mathbf{x},\mathbf{x}'\right)_{CMP}=k\left(\mathbf{x},\mathbf{x}'\right)_{RBF}+k\left(\mathbf{x},\mathbf{x}'\right)_{Poly}+k\left(\mathbf{x},\mathbf{x}'\right)_{WN}, \quad (12)$$

with the polynomial covariance function being

$$k\left(\mathbf{x},\mathbf{x}'\right)_{Poly}=\sigma^2\,k\left(\mathbf{x},\mathbf{x}'\right)_{DP}k\left(\mathbf{x},\mathbf{x}'\right)_{DP}, \quad (13)$$

where $k\left(\mathbf{x},\mathbf{x}'\right)_{DP}=\mathbf{x}\cdot\mathbf{x}'$, and the *white noise* (WN) covariance function being

$$k\left(\mathbf{x},\mathbf{x}'\right)_{WN}=\sigma^2. \quad (14)$$

B. EXPECTATION PROPAGATION SOLUTION FOR GAUSSIAN PROCESS CLASSIFICATION

According to Rasmussen and Williams [27] and for the case of GPC, EP can be used to approximate the predictive distribution $p\left(y^\star=1\,|X,\mathbf{y},X^\star\right)$ with a variational distribution $q(y^\star=1|X,\mathbf{y},X)$, which can be computed with

$$q(y^\star=1|X,\mathbf{y},X^\star)=\int\Phi(f^\star)q\left(f^\star|X,\mathbf{y},X^\star\right)df^\star, \quad (15)$$

where $q\left(f^\star|X,\mathbf{y},X^\star\right)$ is Gaussian with mean

$$\mathbb{E}_q\left[f^\star|X,\mathbf{y},X^\star\right]=k\left(X,X^\star\right)^\top\left(k\left(X,X\right)+\tilde{\Sigma}\right)^{-1}\tilde{\boldsymbol{\mu}}, \quad (16)$$

and variance

$$\mathbb{V}_q\left[f^\star|X,\mathbf{y},X^\star\right]=k\left(X^\star,X^\star\right)-k\left(X,X^\star\right)^\top \times\left(k\left(X,X\right)+\tilde{\Sigma}\right)^{-1}k\left(X,X^\star\right). \quad (17)$$

The predictive probability for a new input $q(y^\star=1|X,\mathbf{y},X)$ is given by:

$$\Phi\left(\frac{k\left(X,X^\star\right)^\top\left(k\left(X,X\right)+\tilde{\Sigma}\right)^{-1}\tilde{\boldsymbol{\mu}}}{\sqrt{1+k(X^\star,X^\star)-k(X,X^\star)^\top\left(k\left(X,X\right)+\tilde{\Sigma}\right)^{-1}k(X,X^\star)}}\right). \quad (18)$$

The parameters $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$ are the mean vector and covariance matrix of the *local likelihood approximations* of the approximate posterior $q\left(f^\star|X,\mathbf{y},X^\star\right)$. They are sequentially obtained in multiple EP steps until convergence of the marginal likelihood can be observed. Φ is the probit function.

C. MONOTONICITY FOR GAUSSIAN PROCESS REGRESSION

Riihimäki and Vehtari presented a method to locally ensure monotonicity around observations [42]. The idea is to add so called *virtual observations* $(\mathbf{x}_v|\mathbf{y}_v)$ that carry monotonicity information and jointly train the GP on the normal and virtual observations. The positions $\mathbf{x}_v$ can be either chosen to uniformly lie on a grid, or adaptively inserted at places with high probability of having the wrong kind of gradient. If the distance between two virtual observations is comparable to the length-scale of the covariance function, the region between them will also be affected by their monotonicity information, due to its covariance structure. As the derivative of a GP is also a GP [43], it is possible to add derivative observations into the process. The linearity of expected value and variance operations allows analytic expressions describing the derivative of an expected value, the covariance between a derivative and a distribution, and the covariance between two derivatives. Unfortunately, this insights make it possible to include only specific values of *slope* and not an overall increasing or decreasing trend of the latent function. For this reason, a probit likelihood is used to link virtual monotonicity observations $y_v\in\{-1,1\}$, at points $\mathbf{x}_v$ with their derivatives, resulting in

$$p\left(y_v\mid\frac{\partial f_v}{\partial x_v^{(d)}}\right)=\Phi\left(\frac{\partial f_v}{\partial x_v^{(d)}}\frac{1}{\nu}\right)=\int_{-\infty}^{\frac{\partial f_v}{\partial x_v^{(d)}}\frac{1}{\nu}}\mathcal{N}(t\mid0,1)\,dt, \quad (19)$$

with ν being a parameter controlling the steepness of the probit function. If $\nu\rightarrow0$, probit becomes a step function, assigning all $\frac{\partial f_v}{\partial x_v^{(d)}}>0$ to 1 and $\frac{\partial f_v}{\partial x_v^{(d)}}<0$ to -1, symbolizing increasing and decreasing behaviors. Following [42], this work set $\nu=10^{-6}$, in order to tolerate small errors. Equipped with this tool, the full posterior on all latent function values becomes

$$p\left(\mathbf{f},\mathbf{f}'|X,X_v,\mathbf{y},\mathbf{y}_v\right)=\frac{1}{Z}p\left(\mathbf{y}|\mathbf{f}\right)p\left(\mathbf{y}_v|\mathbf{f}'\right)p\left(\mathbf{f},\mathbf{f}'|X,X_v\right), \quad (20)$$

where $\mathbf{f}'$ represents the derivative of $\mathbf{f}$, Z the marginal likelihood, $p\left(\mathbf{y}_v\mid\mathbf{f}'\right)$ is the introduced likelihood in 19, and $p\left(\mathbf{f},\mathbf{f}'\mid X,X_v\right)$ the GP prior

$$p\left(\mathbf{f},\mathbf{f}'\mid X,X_v\right)=\mathcal{N}\left(\mathbf{f}_{joint}\mid\mathbf{0},K_{joint}\right), \quad (21)$$

with

$$\mathbf{f}_{joint}=\begin{bmatrix}\mathbf{f}\\ \mathbf{f}'\end{bmatrix}, \qquad K_{joint}=\begin{bmatrix}K(X,X) & K(X,X_v)\\ K(X_v,X) & K(X_v,X_v)\end{bmatrix}. \quad (22)$$

This creates a similar problem as in GP classification, because $p\left(\mathbf{y}_v\mid\mathbf{f}'\right)$ is non-Gaussian and thus the expression cannot be evaluated analytically. Again, EP was used to approximate this posterior.

APPENDIX B
MODEL IMPLEMENTATION DETAILS

For the HI value estimation the `GPy.core.GP()`[1] implementation was used with an RBF covariance function, a Bernoulli likelihood, a probit link function and EP as inference method. The maximum number of optimization

[1] https://github.com/SheffieldML/GPy

TABLE 7. Overview of statistical features used for PCA for the FEMTO data experiments. In the time domain, $x[i]$ is the i^{th} sample of a signal and N the total number of samples. In the frequency domain, $X[f_i]$ is the magnitude at the i^{th} frequency and f_M the highest frequency. For the frequency domain, a FFT was used to create a one-sided positive frequency spectrum.

name	definition	description
time domain features		
RMS	$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x[i]^2}$	describes the energy content of the signal
delta RMS	$\Delta RMS = RMS_t - RMS_{t-1}$	difference between the RMS value of snapshot at time t and the previous snapshot at time $t-1$
mean	$\mu = \frac{1}{N}\sum_{i=1}^{N} x[i]$	mean of the signal values
peak value	$PV = \max_{i=1 \ldots N} \lvert x[i] \rvert$	maximum absolute value of the signal
peak to peak	$PtP = x_{max} - x_{min}$	distance between the maximum and the minimum value of a signal
variance	$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N} (x[i]-\mu)^2$	spread of the PDF of a signal
skewness	$Skew = \frac{1}{N}\sum_{i=1}^{N} (\frac{x[i]-\mu}{\sigma})^3$	symmetry of the PDF of a signal
kurtosis	$Kurt = \frac{1}{N}\sum_{i=1}^{N} (\frac{x[i]-\mu}{\sigma})^4$	sharpness of the PDF of a signal
crest factor	$CF = \frac{PV}{RMS}$	impact during rolling element and raceway contact
impulse factor	$IF = \frac{PV}{\frac{1}{N}\sum_{i=1}^{N} \lvert x[i] \rvert}$	ratio of the Peak value and the absolute mean
margin factor	$MF = \frac{PV}{(\frac{1}{N}\sum_{i=1}^{N} \sqrt{\lvert x[i] \rvert})^2}$	ratio of the Peak value and the RMS of absolute signal
frequency domain features		
dominant frequency	$f_D = \arg\max_{f=1 \ldots M} X[f_i]$	frequency that carries most of the energy in the spectrum
median frequency	$f_{med} = \frac{1}{2}(f_{\lfloor (M+1)/2 \rfloor} + f_{\lceil (M+1)/2 \rceil})$	frequency that divides the energy in the spectrum equally
spectral roll-off	$f_R,\ \text{where} \sum_{i=1}^{R} X[f_i]^2 = 0.95 \sum_{i=1}^{M} X[f_i]^2$	frequency at which 95% of the total energy in the spectrum is covered
spectral centroid	$f_C = \frac{\sum_{i=1}^{M} X[f_i] f_i}{\sum_{i=1}^{M} X[f_i]}$	center of mass of the spectrum
spectral flux	$SF = \sum_{i=1}^{M} (NM_t[i] - NM_{t-1}[i])^2$	squared difference between the normalized magnitudes of the spectra of two successive short-term windows
	$NM_t[i] = \frac{X_t[f_i]}{\sum_{j=1}^{M} X_t[f_j]}$	

steps was set to 200. For the extrapolation of the HI values, the Matlab® `gpstuff` implementation of monotonic regression, `gp_monotonic`, was used [29]. A Gaussian likelihood with log uniform prior on the variance was used, as well as an RBF covariance function with a uniform prior on the length-scale parameter and squared uniform prior on the magnitude parameter. Positive noise, which is added to the diagonal of the covariance matrix for numerical stability, was set to the value 10^{-9} and the default scaled conjugate gradient method was chosen for optimization. The warping function was defined as a Gaussian cumulative function with mean 0.5 and variance 0.1.

APPENDIX C
EXTRACTED STATISTICAL FEATURES

The extracted features used for the FEMTO experiments are described in Table 7.

AUTHOR CONTRIBUTIONS

Maximilian Benker: Conceptualization, methodology, validation, investigation, software, writing - original draft, supervision. **Artem Bliznyuk**: Methodology, validation, investigation, software, writing - review & editing. **Michael F. Zaeh**: Writing - review & editing, supervision, funding acquisition.

REFERENCES

[1] H. Lasi, P. Fettke, H. G. Kemper, T. Feld, and M. Hoffmann, ''Industry 4.0,'' *Bus. Inf. Syst. Eng.*, vol. 6, no. 4, pp. 239–242, 2014.

[2] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, ''Smart factory of industry 4.0: Key technologies, application case, and challenges,'' *IEEE Access*, vol. 6, pp. 6505–6519, 2017.

[3] R. Gouriveau, K. Medjaher, and N. Zerhouni, *From Prognostics and Health Systems Management to Predictive Maintenance 1: Monitoring and Prognostics*. Hoboken, NJ, USA: Wiley, 2016.

[4] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, ''Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications,'' *Mech. Syst. Signal Process.*, vol. 42, nos. 1–2, pp. 314–334, Jan. 2014.

[5] A. Heng, A. C. C. Tan, J. Mathew, N. Montgomery, D. Banjevic, and A. K. S. Jardine, ''Intelligent condition-based prediction of machinery reliability,'' *Mech. Syst. Signal. Process.*, vol. 23, no. 5, pp. 1600–1614, Jul. 2009.

[6] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, ''Machinery health prognostics: A systematic review from data acquisition to RUL prediction,'' *Mech. Syst. Signal Process.*, vol. 104, pp. 799–834, May 2018.

[7] O. Fink, Q. Wang, M. Svensén, P. Dersin, W.-J. Lee, and M. Ducoffe, ''Potential, challenges and future directions for deep learning in prognostics and health management applications,'' *Eng. Appl. Artif. Intell.*, vol. 92, Jun. 2020, Art. no. 103678.

[8] L. Zhang, J. Lin, B. Liu, Z. Zhang, X. Yan, and M. Wei, ''A review on deep learning applications in prognostics and health management,'' *IEEE Access*, vol. 7, pp. 162415–162438, 2019.

[9] M. Benker, L. Furtner, T. Semm, and M. F. Zaeh, ''Utilizing uncertainty information in remaining useful life estimation via Bayesian neural networks and Hamiltonian Monte Carlo,'' *J. Manuf. Syst.*, Dec. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0278612520301928

[10] X. Li, Q. Ding, and J.-Q. Sun, ''Remaining useful life estimation in prognostics using deep convolution neural networks,'' *Rel. Eng. Syst. Saf.*, vol. 172, pp. 1–11, Apr. 2018.

[11] M. Benker, R. Kleinwort, and M. F. Zah, ''Estimating remaining useful life of machine tool ball screws via probabilistic classification,'' in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, San Francisco, CA, USA, Jun. 2019, pp. 1–7.

[12] A. Saxena, K. Goebel, D. Simon, and N. Eklund, ''Damage propagation modeling for aircraft engine run-to-failure simulation,'' in *Proc. Int. Conf. Prognostics Health Manage.*, Denver, CO, USA, Oct. 2008, pp. 1–9.

[13] P. Nectoux, R. Gouriveau, K. Medjaher, E. Ramasso, B. Chebel-Morello, N. Zerhouni, and C. Varnier, ''PRONOSTIA: An experimental platform for bearings accelerated degradation tests,'' in *Proc. IEEE Int. Conf. Prognostics Health Manage.*, Denver, CO, USA, Feb. 2012, pp. 1–8.

[14] R. Satishkumar and V. Sugumaran, ''Estimation of remaining useful life of bearings based on support vector regression,'' *Indian J. Sci. Technol.*, vol. 9, no. 10, p. 10, Mar. 2016.

[15] A. Mosallam, K. Medjaher, and N. Zerhouni, ''Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction,'' *J. Intell. Manuf.*, vol. 27, no. 5, pp. 1037–1048, Oct. 2016.

[16] B. Chinomona, C. Chung, L.-K. Chang, W.-C. Su, and M.-C. Tsai, ''Long short-term memory approach to estimate battery remaining useful life using partial data,'' *IEEE Access*, vol. 8, pp. 165419–165431, 2020.

[17] C. Sun, M. Ma, Z. Zhao, S. Tian, R. Yan, and X. Chen, ''Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing,'' *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2416–2425, Apr. 2019.

[18] B. Yang, R. Liu, and E. Zio, ''Remaining useful life prediction based on a double-convolutional neural network architecture,'' *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9521–9530, Dec. 2019.

[19] H. Lv, J. Chen, and T. Pan, ''Sequence adaptation adversarial network for remaining useful life prediction using small data set,'' in *Proc. IEEE 18th Int. Conf. Ind. Informat. (INDIN)*, Warwick, U.K., Jul. 2020, pp. 115–118.

[20] A. Zhang, H. Wang, S. Li, Y. Cui, Z. Liu, G. Yang, and J. Hu, ''Transfer learning with deep recurrent neural networks for remaining useful life estimation,'' *Appl. Sci.*, vol. 8, no. 12, p. 2416, Nov. 2018.

[21] K. T. P. Nguyen and K. Medjaher, ''An automated health indicator construction methodology for prognostics based on multi-criteria optimization,'' *ISA Trans.*, vol. 113, pp. 81–96, Jul. 2021.

[22] P. Wen, S. Zhao, S. Chen, and Y. Li, ''A generalized remaining useful life prediction method for complex systems based on composite health indicator,'' *Rel. Eng. Syst. Saf.*, vol. 205, Jan. 2021, Art. no. 107241.

[23] P. Li, X. Jia, J. Feng, H. Davari, G. Qiao, Y. Hwang, and J. Lee, ''Prognosability study of ball screw degradation using systematic methodology,'' *Mech. Syst. Signal Process.*, vol. 109, pp. 45–57, Sep. 2018.

[24] J. Liu and Z. Chen, ''Remaining useful life prediction of lithium-ion batteries based on health indicator and Gaussian process regression model,'' *IEEE Access*, vol. 7, pp. 39474–39484, 2019.

[25] S. A. Aye and P. S. Heyns, ''An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission,'' *Mech. Syst. Signal Process.*, vol. 84, pp. 485–498, Feb. 2017.

[26] M. Radford Neal, *Bayesian Learning for Neural Networks* (Lecture Notes in Statistics). New York, NY, USA: Springer, 1996.

[27] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning Adaptive Computation and Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[28] I. Ustyuzhaninov, I. Kazlauskaite, C. H. Ek, and N. Campbell, ''Monotonic Gaussian process flows,'' in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 3057–3067.

[29] J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari, ''GPstuff: Bayesian modeling with Gaussian processes,'' *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 1175–1179, Apr. 2013.

[30] E. Snelson, Z. Ghahramani, and C. Rasmussen, ''Warped Gaussian processes,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 16, 2003, pp. 337–344.

[31] B. S. Jensen, J. B. Nielsen, and J. Larsen, ''Bounded Gaussian process regression,'' in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, Southampton, U.K., Sep. 2013, pp. 1–6.

[32] M. Kuss and C. E. Rasmussen, ''Assessing approximate inference for binary Gaussian process classification,'' *J. Mach. Learn. Res.*, vol. 6, pp. 1679–1704, Oct. 2005.

[33] P. T. Minka, ''Expectation propagation for approximate Bayesian inference,'' in *Proc. 17th Conf. Uncertainty Artif. Intell.*, San Francisco, CA, USA, 2001, pp. 362–369.

[34] *Rolling Bearings—Dynamic Load Ratings and Rating Life*, Standard ISO 281:2007, International Organization for Standardization, 2007.

[35] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, ''Estimation of remaining useful life of ball bearings using data driven methodologies,'' in *Proc. IEEE Conf. Prognostics Health Manage.*, Denver, CO, USA, Jun. 2012, pp. 1–7.

[36] S. Kim, S. Park, J.-W. Kim, J. Han, D. An, N. H. Kim, and A.-H. Choi, ''A new prognostics approach for bearing based on entropy decrease and comparison with existing methods,'' in *Proc. Annu. Conf. PHM Soc.*, 2016, vol. 8, no. 1, pp. 1–5.

[37] Y. Jiang, C. Li, Z. Yang, Y. Zhao, and X. Wang, ''Remaining useful life estimation combining two-step maximal information coefficient and temporal convolutional network with attention mechanism,'' *IEEE Access*, vol. 9, pp. 16323–16336, 2021.

[38] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, ''Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2306–2318, Oct. 2017.

[39] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, ''Long short-term memory network for remaining useful life estimation,'' in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Dallas, TX, USA, Jun. 2017, pp. 88–95.

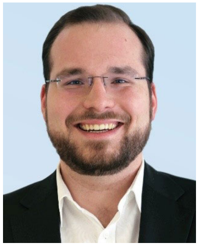
[40] Q. Wang, A. Farahat, C. Gupta, and H. Wang, ''Health indicator forecasting for improving remaining useful life estimation,'' in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Detroit, MI, USA, Jun. 2020, pp. 1–8.

[41] G. Michau, T. Palm, and O. Fink, ''Deep feature learning network for fault detection and isolation,'' in *Proc. Annu. Conf. PHM Soc.*, 2017, vol. 9, no. 1, pp. 1–11.

[42] J. Riihimäki and A. Vehtari, ''Gaussian processes with monotonicity information,'' in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 645–652.

[43] E. Solak, R. Murray-Smith, WE. Leithead, D. Leith, and CE. Rasmussen, ''Derivative observations in Gaussian process models of dynamic systems,'' in *Proc. Adv. Neural Inf. Process. Syst.* Cambridge, MA, USA: MIT Press, 2003, pp. 1033–1040.

MAXIMILIAN BENKER received the B.Sc. and M.Sc. degrees in industrial engineering from the Technical University of Berlin. He is currently pursuing the Ph.D. degree with the Institute for Machine Tools and Industrial Management (iwb), Technical University of Munich.



ARTEM BLIZNYUK received the B.Sc. degree in engineering science and the M.Sc. degree in robotics, cognition, intelligence from the Technical University of Munich. He is currently pursuing the Ph.D. degree with the Institute for Man-Machine Interaction, RWTH Aachen University.

MICHAEL F. ZAEH graduated in mechanical engineering from Technical University of Munich, where he also earned his doctorate degree in 1993 under the supervision of Professor Milberg, at the Institute for Machine Tools and Industrial Management (iwb). From 1994 to 1995, he was Chief Engineer and Department Head for Machine Tools and Production Technology under the direction of Professor Reinhart. In 1996, he switched over to the private sector, working for a manufacturer of machine tools used for gear wheel machining, where he held various management positions. In 2002, Professor Zaeh accepted the Chair of Machine Tools and Production Technology at Technical University of Munich and has held the position of Director of the iwb since then.