Remaining Useful Life Estimation for Unknown Motors Using a Hybrid Modeling Approach

Marcel Hildebrandt[1,2], Mohamed Khalil [1,3], Christoph Bergs[1,4], Volker Tresp[1,2], Roland Wüchner[3], Kai-Uwe Bletzinger[3], Michael Heizmann[4]
[1] Siemens AG - Corporate Technology, Munich, Germany, {marcel.hildebrandt, khalil.mohamed, christoph.bergs, volker.tresp}@siemens.com
[2] Ludwig Maximilian University - Institute of Informatics, Munich, Germany
[3] Technical University Munich - Department of Civil, Geo and Environmental Engineering, Munich, Germany, {wuechner, kub}@tum.de
[4] Karlsruhe Institute of Technology - Institute of Industrial Information Technology, Karlsruhe, Germany, michael.heizmann@kit.edu

***Abstract*—Remaining useful life estimation is a research topic of high relevance in the area of structural mechanics. To predict the remaining useful lifetime of a motor, domain experts commonly employ physical simulations based on 3D-CAD models. However, this process is laborious and in many cases no 3D-CAD model is available. Also, setting up a simulation might require substantial efforts or might even be infeasible. This article focuses on the machine learning based estimation of the remaining useful life of unknown, derived motor types of an electric motor class based on simulations of known motor types, as well as data sheets and measurements. In particular, we propose the hybrid fusion method *moSAIc* that allows to transfer the knowledge inherent in physical degradation models of motors to unknown instances. Our experiments show that *moSAIc* outperforms other state-of-the-art methods by a large margin in terms of both accuracy and robustness. Furthermore, compared to purely data-driven methods such as neural networks, *moSAIc* is explainable allowing domain experts to understand the reason for the predictions.**

***Index Terms*—RUL Estimation, Hybrid Modeling, Structural Mechanic Simulation, Machine Learning for Fleet, Ensemble Methods, Structural Health Monitoring**

I. Introduction

The digitization of industrial processes is rapidly accelerating and shows its impact in every phase of the product life cycle of a component. On the one hand, in the design phase of a component theoretical simulation models are set up to guarantee its performance specifications. On the other hand, more and more operational data of the component is provided by IoT-capable systems from the shop floor of industrial productions. The consequence of this evolution is a larger and richer database for applications that are data hungry, such as many machine learning algorithms. Industrial components are exposed to lifetime degradation during their operation phase due to different and hard-predictable loading profiles. A strategy to monitor this degradation process is Condition-based Maintenance (CBM). With this strategy, real-time capable condition monitoring can be used to decrease the amount of unnecessary maintenance operations [1]. A fundamental pillar activity in CBM is *prognostics*, or estimating the remaining lifetime of components in operation. This estimate is calculated considering the processing of the recorded sensors data, and applying the proper state estimation and failure diagnosis methods, as detailed in [2]. In this work, a novel method is proposed to estimate the remaining useful life of a component in the context of CBM, based on the combination of existing models.

In the following, the system and model definitions according to [3] are introduced to clearly define the setup for which the method is applied. [3] defines a dynamic system where $u(t)$ is the input signal, $x(t)$ is the system state, $d^{\mathrm{m}}(t), d^{\mathrm{nm}}(t)$ are measurable respectively non-measurable disturbances, and $y(t)$ is the system output. This system can be modeled using different approaches. Design engineers normally use the theoretical modeling approach based on first principals that describe the physics of the system, especially the evolution of the system state $x(t)$. Models derived from first principals result in so called *White-Box Models*.

In contrast to a White-Box Model, the system can be modeled using some flexible mathematical model, such as a neural network, by solely using measurements on input, output, and disturbances. Here, the system state $x(t)$ might not have a clear physical interpretation. Such models are called *Black-Box Models*.

In the literature the combination of both model types are usually called *Grey-Box Model*. Models of such type consist of parts that are described using first principles and parts that are derived from observed data [3].

In this article, a situation is considered where no *White-Box Model* - in this case no physical simulation model for the particular motor type - is available. However, *White-Box Models* of related motor types are at hand. A straightforward approach to obtain an estimate of the remaining useful life (RUL) is to search for the most similar motor and employ the corresponding theoretical simulation model. For example, the aging behavior of a motor is highly dependent on the underlying geometry and the material properties. This information is typically stored in the catalog data. Thus, one could apply a similarity search based on features listed in the motors' catalog and either employ a model corresponding to the most similar motor or derive a weighting scheme.

In this work, *moSAIc* - a novel method to compute the RUL of a motor in the case no suitable *White-Box Model* is available is proposed. It goes beyond existing weighting schemes and *Grey-Box Models* in the sense that it rather constitutes a *Hybrid Model* that combines existing *White-Box Models* using a data-driven fusion method represented by the fusion operator $\circledcirc$ [4]. The corresponding block diagram of a hybrid method like *moSAIc* is shown in Figure 1. In this case, $u(t)$ consists of the boundary conditions of the model. These are physically parallel misalignment, angular misalignment and torque. $x_i(t)$ is represented through a finite element (FE) model of an existing motor predicting its specific RUL, and d^{m} is considered to be the modeling error due to simplifications of the FE models.

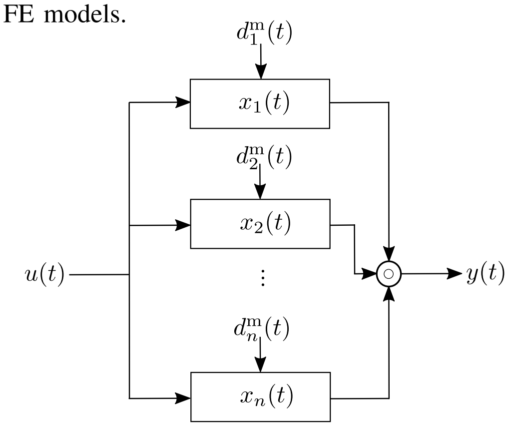

Fig. 1. Block diagram of a hybrid model

This article focuses on the RUL estimation of electric asynchronous motors using a hybrid modeling approach consisting of combined RUL estimation models using a novel ensemble approach. Section II provides an overview about relevant state-of-the-art methods. Section III introduces the underlying structural mechanics model that is used to estimate the lifetime of motors and to generate data for experiments. The different operating conditions are also described, along with the different motor variants used to build up the fleet. Section IV describes the pillars of the ensemble method used to build *moSAIc*. In Section V the experimental conditions and the evaluation procedure are outlined and the results of the experimental study are described. In particular, the performance of *moSAIc* is compared to feed-forward neural-networks, direct similarity searches, and weighted similarity search methods. Finally, Section VI contains the conclusions.

II. Related Methods

A. RUL Estimation

RUL estimation is an important task in condition based maintenance. Many approaches have been investigated in the literature to address, e.g., the question of estimating the lifetime of components in service. [5] summarizes these frameworks and give a comprehensive review of the methods.

The traditional and most-common approach for assessing RUL is based on physics-based simulation models. Hereby, the simulation models are used to evaluate the stress distribution on the motor under certain operating conditions. From the resulting stress distribution, critical regions are identified and the corresponding stress values are input to SN-curves with the goal of computing the corresponding number of cycles to failure.

In order to introduce this approach into an online monitoring framework, the sensor signals are analyzed to identify the operating conditions and anomalies, which would then be fed into the simulation model as boundary conditions. An alternative which avoids this signal processing step is to estimate RUL using statistical models. These RUL models analyze directly the raw sensor signals, which are modelled to follow a gamma-distributed random variable which correlates directly to the lifetime of the motor. [6] gives some details about these approaches. The challenge underlying stochastic models is the identification of the parameters that fit the probability distribution to the specific motor's degradation behavior under consideration. On the other hand, setting up a physical simulation is time consuming and requires a lot of expertise. Moreover, these models require detailed information about the motor type under consideration (e.g. a CAD-model) which may not be available.

Both the stochastic and the physical models fall under the category of white-box models. In contrast, data-driven approaches, also called black-box models, attempt to estimate RUL directly based on observed sensor signals. In e.g. [7], the authors propose an approach based on deep convolutional neural networks. In [8] an ensemble learning-based approach is used to estimate the RUL by fusing several multi-layer neural networks, by dynamically allocating weights to each. On the level of grey-box models, many research has been conducted trying to utilize sensor-data-based model updating techniques such as Kalman filters to calibrate the physical model and the current state of the running machinery, e.g. [9]. Additionally, regression models of health indicators which correlate directly to RUL can be applied based on the obtained sensor data which is acquired from the bearings [10]. Also, different hybrid approaches are used in the literature to estimate the RUL of a machinery. [11] introduce a hybrid framework combining data-driven and model-based methods for remaining useful life (RUL) prediction using similarity-based prediction methods. A physical degradation model in the form of an analytical system equation is combined with two data-driven techniques, one for estimating the measurement model and one for predicting future measurements. The method is applied to a battery RUL estimation use case where it outperforms comparable single model-based or data-driven approaches. Hereby a weighted average function combines historical measurement datasets to predict future measurements. The Euclidean or Mahalanobis distance is used to determine the weight of each historical dataset in a linear model. A hybrid soft computing model comprising the Fuzzy Min-Max (FMM) neural network and the Classification and Regression Tree (CART) for motor fault detection and diagnosis is introduced in [12]. They use the current signal of the induction motor to detect different types of faults, e.g., stator winding faults or broken rotor bars, to compute the RUL of the induction motor.

B. Ensemble Methods

When training supervised machine learning methods, one searches through the hypothesis space to find a suitable mapping that related the input to the output. In most settings, different competing approaches are available to perform the task under consideration. The most common routine is to select the method that shows the best performance on a validation set while discarding all other methods. However, restricting oneself to a single method ignores the fact that in many cases there is no single technique that strictly outperforms all others. It is rather the case that the employed methods achieve different levels of performance in different input regions. This implies that different approaches may complement each other. Moreover, the restriction to one single method is wasteful in the sense that it does not make use of the resources employed for constructing the algorithms that are ignored in the deployment period. To overcome these drawbacks, one can combine multiple data-driven algorithms to form one joint committee of methods. This procedure is commonly referred to as ensemble methods. Related to the domain under consideration, [8] and [13] propose ensemble methods for RUL estimation.

When employing multiple methods simultaneously one needs to aggregate different predictions. [13] compares different aggregation methods such as simple voting mechanism and various weighting scheme formulations in the context of RUL estimation. In particular, they find that an optimization-based weighting achieves the best performance. [14] proposes a weighting schemes that either consider the variance of the estimators or whether a given estimator has seen data that is similar to the input under consideration. Gating is another popular weighting scheme. The underlying idea is that a trainable meta-algorithm produces linear weight that allow to average over the available methods. Often, a neural network is used for the gating mechanism [15]. The Gating Network outputs a vector that acts as a chooser which methods to consult.

III. Structural mechanics model

Before proceeding, first the mathematical notation which is used throughout this work will be defined: Scalars are given by either lower or upper case letters ($x \in \mathbb{R}$ or $X \in \mathbb{R}$), vectors by bold lower case letters ($\mathbf{x} \in \mathbb{R}^n$), matrices by bold upper case letter ($\mathbf{X} \in \mathbb{R}^{n \times m}$), and sets by capital greek letters (e.g., Δ).

A. FEM Model

The finite elements method (FEM) has gained a wide consent over the past decades in describing structural mechanics phenomena. This is mainly due to its robustness and satisfactory performance. In this work, FEM is utilized to simulate the motor's response to given operating conditions and estimate its RUL. Typically, the raw solutions of a FE solver are the displacements of the system, while stresses can then be retrieved from the post-processing of the displacement. For each material, the cyclic load bearing capacity, in terms of load cycles (N), with respect to a stress value (S) is described by an SN-curve. SN-curves are typically obtained from experiments conducted on material samples by manufacturers or are defined through empirical mathematical models, such as Basquin or Wöhler models. Since stress is a second-order tensor, while the SN-curve requires a scalar value on the stress scale, the stress tensor is reduced to a single component, either by selecting the most influential component to the damage process, e.g., shear stress in a given plane, or by calculating an equivalent stress S_{eq} from the full tensor, such as the maximum principal stress [16].

In this work, electric motors are investigated with regard to their lifetime. The aim is to generate a fleet of similar motors, and investigate how the change in operating conditions, material and geometry influence their RULs. For this purpose, Figure 2 shows the adopted geometry for a pseudo-motor. It consists of selected principal sub-components of a motor, such as housing, rotor, stator, shaft, and bearings. Other sub-components, such as the windings and slip-rings, are intentionally left out to reduce the complexity of the model. Additionally, the modeling selected features are over-simplified for the same reason; the housing fins, for instance, have been completely eliminated, and a simple hollow cylinder is used to model the housing.

The above mentioned simplification results in a modeling error which cannot be neglected. Unfortunately, setting up a comprehensive FE model for a fleet of motors can be quite cumbersome. This is due to the fact that the amount of geometric details, which require effort in geometry preparation and meshing, would eventually require long pre-processing and solution times. On the other hand, the model's simplicity helps synthesizing the required training data while utilizing negligible computational resources.

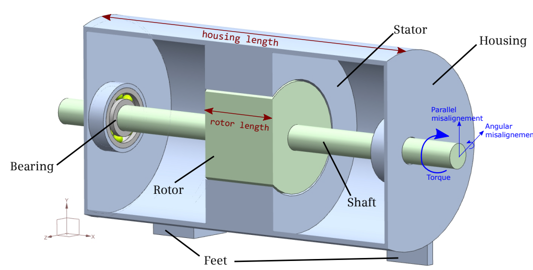

Fig. 2. Section cut through the motor model

Figure 2 shows a cross-section of the motor structure which is utilized in this work including the parameters that differ for motor variants of the fleet. In this work, 18 motors were created by the combination of the three variable parameters mentioned above. Table I summarizes the combinations where $\mathrm{E} \in \mathbb{R}$ is the elastic modulus and $\nu \in \mathbb{R}$ is the Poisson ratio.

Another aspect that influences the lifetime of the motor is the operating condition. Aside from nominal design loads, during operation, motors are subjected to operation anomalies, among which the most famous is the misalignment of the shaft's driving-end. Misalignment in motors is classified into parallel and angular misalignments, referred to in this work as $\delta, \varphi \in \mathbb{R}$, respectively. Moreover, a significant operating condition imposed on the motor is the torque ($T \in \mathbb{R}$), which varies depending on the load induced by the driven component. These operating conditions are imposed on the simulation model as boundary conditions. Since motors of a similar family are considered in the fleet, a nominal power, $P = 250\ W$, is taken to be equal for all motors. To estimate the lifetime for motors with unknown simulation models based

TABLE I
VARIANTS OF MOTOR FLEET AND THEIR CORRESPONDING PROPERTIES

Model Num.	Housing Material	Housing Length in mm	Rotor Length in mm
1			30.0
2		165.0	50.0
3	Aluminum		70.0
4	Al 5086		30.0
5		195.0	50.0
6	E = 72 GPa		70.0
7	ν = 0.33		30.0
8		225.0	50.0
9			70.0
10			30.0
11		165.0	50.0
12	Cast Iron		70.0
13	UNI 5007 Grade 25		30.0
14		195.0	50.0
15	E = 90 GPa		70.0
16	ν = 0.3		30.0
17		225.0	50.0
18			70.0

on motors with known simulation models, the response of the latter motors with respect to the aforementioned operating conditions must be learned (see Section IV). The ranges of operating conditions are denoted in Equation 1 by $\mathcal{R}$.

$$\begin{aligned} \Omega &\in \mathcal{R}\left(T = \frac{25}{13}, r = \frac{25}{3}\right) =: R_\Omega, \\ \Delta &\in \mathcal{R}\left(\delta = 0.20, \delta = 0.30\right) =: R_\Delta, \\ \Phi &\in \mathcal{R}\left(\varphi = 0.25, \varphi = 0.35\right) =: R_\Phi. \end{aligned} \tag{1}$$

B. RUL Computation

Let $\mathcal{U}_{ijk} := \{\Omega_i, \Delta_j, \Phi_k\}$, where $\Omega_i \in R_\Omega$, $\Delta_i \in R_\Delta$, $\Phi_i \in R_\Phi$, be a set of torques, parallel and angular misalignments, respectively, bounded by the spaces defined in Equation (1). These operating conditions serve as input to a white-box model system , which is the motor's FE model. Hereby, the ranges of chosen operating conditions are such that they lead to a deterioration of the lifetime of the motor. The lifetime estimation at the most critically loaded position is the output of this system, evaluated according to the Basquin equation, given by

$$S_{\text{eq}} = S_{\text{f}}\, n_{\text{L}}^{z}\,, \tag{2}$$

where S_{eq} is the equivalent scalar value of the stress tensor as discussed earlier. $n_{\text{L}} \in \mathbb{N}$ denotes the corresponding number of load cycles until the end of the useful lifetime, $S_{\text{f}} \in \mathbb{R}$ and $z \in \mathbb{R}$ are the fatigue strength and the fatigue coefficient, respectively. The largest maximum principal stress ($S_{\text{eq}} = 90$ MPa) lies at the root of the shaft's drive end, see Figure 3, and lies well below the yield strength of the shaft's material (AISI 1045), typically $S_{\text{y}} = 410MPa$. Since carbon steel is a homogeneous, isotropic material, it experiences the linear relationship between stress $\mathbf{S}$ and strain $\mathbf{E}_{\text{s}}$ governed by the material tensor $\mathbf{C}$. Herein, since $\mathcal{U}_{11,11,11}$ is the severest operating condition, it yields the maximum strain, and hence stress. Therefore, other combinations of operating conditions would yield values well below S_{y}, making high-cycle fatigue (HCF) the mechanism of lifetime deterioration, thus, the use of Basquin model is justified [17].

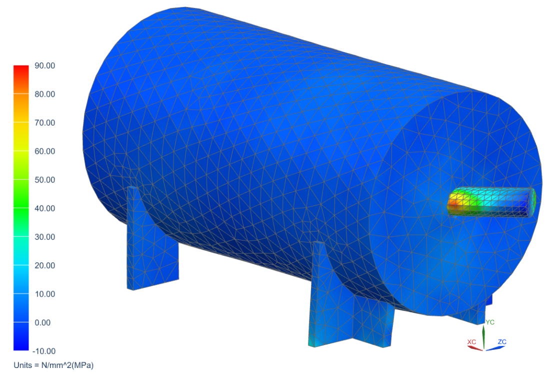

Fig. 3. Distribution of the maximum principal stresses for the operating conditions $\mathcal{U}_{ijk} = \{\frac{25}{3}, 0.3, 0.35\}$

IV. OUR METHOD

In this section *moSAIc* is introduced, a hybrid model that combines existing white-box models using a data-driven fusion method. The underlying idea is to exploit the implicit relationship between the geometry, the material properties of motors, and their aging behavior in order to transfer simulations models to motors where no specific white-box models is available. More concretely, consider the set of available simulations $\mathcal{M}$, where each $f : R_\Omega \times R_\delta \times R_\varphi \to \mathbb{R}_{\geq 0}$ for $f \in \mathcal{M}$ corresponds to a FEM model described in Section III-A. Further, suppose an indexed subset $\mathcal{B} := \{f_1^{\text{b}}, f_2^{\text{b}}, \ldots, f_{n_b}^{\text{b}}\} \subset \mathcal{M}$ from the set of all available simulations has been selected. A framework to transfer simulation models to unknown motors is proposed, in which the models in $\mathcal{B}$ span the space of attainable hybrid models. Thus, in analogy to the role of bases in linear algebra, the elements of $\mathcal{B}$ can be called basis simulations which is indicated by the superscript b. Furthermore, let $y_i := f_i^{\text{b}}(\boldsymbol{u})$ denote the predicted RUL produced by the $i-$th basis simulation depending on the arbitrary but fixed sensor input $\boldsymbol{u} \in R_\Omega \times R_\delta \times R_\varphi$. Since the basis simulations are not designed to produce accurate results for the motor under consideration, it can be expected that most predictions are imprecise. Nevertheless, they may still contain useful information concerning the RUL of the unknown motor. The exact amount of information is determined by the underlying physical properties of the motors and models at hand. Among these, the geometry as well as the material properties play a particularly important role. This information is typically stored in the data sheet of a motor. The aim is to assign weights to the RULs produced by the basis models $\boldsymbol{y} := (y_1, y_2, \ldots, y_{n_b})$ that allow us to form an aggregate prediction. The approach shares similarity with the ensemble methods described in Section II in the sense that multiple simulation models are employed with the aim to combine them by forming an aggregate prediction. Analogously to a committee of expert, each of the basis simulations outputs a prediction of the RUL.

Then an aggregation mechanism decides which experts to consult.

After having obtained the estimations of the RULs $\boldsymbol{y}$, a gating-network decides how much weight to put on each expert's prediction based on the catalog data. Since the relation between the physical characteristics of a motor and its aging behavior is potentially highly non-linear, a feed-forward neural network for the gating mechanism is employed. More concretely, let $\boldsymbol{c} \in \mathbb{R}^{n_c}$ denote the catalog data of the motor under consideration. $\boldsymbol{c}$ along with the sensor data serves as input to a neural network $g : R_\Omega \times R_\delta \times R_\varphi \times \mathbb{R}^{n_c} \to \mathbb{R}^{n_b}$, where no non-linearity is applied in the output layer. In order to ease the notation let $\boldsymbol{w}(\boldsymbol{u}, \boldsymbol{c}) := (w_1(\boldsymbol{u}, \boldsymbol{c}), w_2(\boldsymbol{u}, \boldsymbol{c}), \ldots, w_{n_b}(\boldsymbol{u}, \boldsymbol{c}))$ denote the output of g depending on the catalog data. Subsequently, the entries of $\boldsymbol{w}(\boldsymbol{u}, \boldsymbol{c})$ serve as linear coefficients to weight the predictions of the basis simulations. More concretely, during the training process the aim is to fit the parameters of g such that

$$\tilde{f}(\boldsymbol{u}) \approx < \boldsymbol{w}(\boldsymbol{u}, \boldsymbol{c}), \boldsymbol{y} >, \tag{3}$$

where $\tilde{f}$ denotes the simulation model of the training motor under consideration and $< \cdot, \cdot >: \mathbb{R}^{n_b} \times \mathbb{R}^{n_b} \to \mathbb{R}$ indicates the Euclidean inner product. From a functional point of view this implies that the hybrid model based on *moSAIc* for the unknown motor is given by

$$f^{\mathrm{h}}(\cdot) = \sum_{i=1}^{n_b} w_i(\boldsymbol{u}, \boldsymbol{c}) f_i^{\mathrm{b}}(\cdot)\,. \tag{4}$$

Figure 4 illustrates the basic architecture of *moSAIc* for the case $n_b = 3$. Concerning the neural network g, various architectures are possible. In this work, only neural network with constant hidden layer sizes are considered. Furthermore, experiments with considering only the catalog data $\boldsymbol{c}$ as input of g has been executed, but are not reported in this article. While this modification limits the expressiveness of our model, it may lead to more robust results.

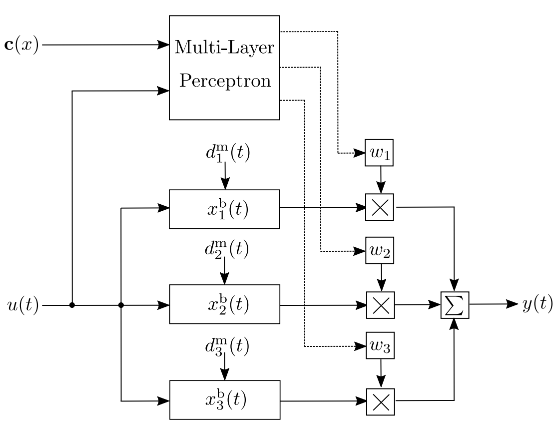

Fig. 4. The architecture of our method.

After putting a test and validation set aside, the data that does not correspond to the basis simulations is employed during training in order to fit the parameters of g. More precisely, the training data is denoted with $\mathcal{T}$, where the set $\mathcal{T}$ contains all sensor data, catalog data, and RUL predictions triples that do not correspond to the basis simulations. During training our objective is to minimize the loss function

$$\mathcal{L} = \sum_{(\boldsymbol{u}, \boldsymbol{c}, \tilde{f}(\boldsymbol{u})) \in \mathcal{T}} \left(\tilde{f}(\boldsymbol{u}) - \underbrace{< \boldsymbol{w}(\boldsymbol{u}, \boldsymbol{c}), \boldsymbol{y} >}_{f^{\mathrm{h}}(\boldsymbol{u})} \right)^2 + \lambda ||g||_{\mathrm{F}}^2\,, \tag{5}$$

where $||g||_{\mathrm{F}}^2$ denotes the sum of the squared Frobenius norms of the weight matrices in g. Furthermore, λ is a hyper parameter that determines the strength of the L_2-regularizer. Equation (5) is minimized using a stochastic gradient method.

moSAIc can be thought of as an implicit similarity based matching of motors and simulations. However, it comes with a major advantages over a simple similarity search, i.e., comparing the catalog data of different motors and applying the simulations of the motor that is most similar (or compute a weighted average) to the motor at hand: It is a priori not clear how the catalog data influences the aging behavior, e.g., in cases where the catalog data may contain uninformative data. Moreover, the link between the catalog data and the aging behaviour may be highly non-linear and depend on the current sensor input. By training a neural network to match motors and simulation models, this mapping from data is learned instead of having to specify it in advance.

V. Experiments

A. Implementation and Evaluation

moSAIc is trained and evaluated on the data described in Section III. The RUL for each motor is measured in months, by solving Equation 2 , as follows:

$$\mathrm{RUL} = \frac{n_L}{\kappa} = \frac{1}{\kappa} \sqrt[z]{\frac{S_{eq}}{S_f}}, \tag{6}$$

where κ is a conversion factor from load cycles to months depending on the rotating speed of the motor, calculated from the torque Ω and power P, and considering the motor's average number of operating hours. Further, *moSAIc* as fusion operator $\circledcirc$ is compared to other fusion operators mentioned in Section IV: two similarity search based methods (best match and weighted) and a direct estimation of the RUL based on neural networks. The similarity based methods consist of comparing the catalog data of all basis motors and compute the Euclidean distance between the normalized feature vectors (see Table I). Then either the simulation model that is most similar to the unknown motor or employ a weighted average where the coefficients are derived from the inverse of the distances is applied. Also, experiments with different weighting schemes (e.g., based on cosine or Jaccard distances) has been done but it was found that they did not improve the performance. For all neural networks that are considered in this work a feedforward neural network with constant hidden layer size and *relu*-activation function is employed. For experimental

TABLE II
RESULTS OF ALL EVALUATED METHODS: THE NUMBERS IN BRACKETS CORRESPOND TO THE STANDARD ERRORS IN THE MONTE CARLO CROSS-VALIDATION SETTING.

Metric	MSE	MAE
Neural Network	124.25 (31.67)	5.52 (0.74)
Similarity Search	510.97 (95.13)	11.29 (1.09)
Weighted Similarity Search	398.28 (54.26)	9.98 (0.79)
moSAIc	**93.70** (19.83)	**4.48** (0.41)

consistency, all methods within the same framework as *moSAIc* in Python and TensorFlow were re-implemented.

To guarantee a fair evaluation in a realistic setting, we employed the following Monte Carlo cross-validation approach: First, randomly four basis motors and five training motors were sampled whose data is employed during training to fit the parameters of the neural network g. The remaining data is not shown to the algorithm during training, but serves as validation set in order to approximate the performance. This procedure is iterated ten times. The resulting errors on the validation set for each iteration are then averaged to obtain an estimation of the overall error. These results along with the corresponding standard errors are reported in the next section.

Having tried optimizers such as stochastic gradient descent, Adam, and AdaGrad, it has been found that the Adam optimizer resulted in the best convergence in this case. The number of hidden layers in the neural network was tuned from the range $\{1, 2, 5, 10\}$ and the number of neurons for each layer was chosen from the range $\{10, 20, 50\}$. The regularization parameter was chosen among $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$ and the learning rate was set to 10^{-5}.

B. Results

Table II displays the results of all methods under consideration. *moSAIc* outperforms the other methods in terms of having a lower mean squared error and median squared error by a significant margin. More concretely, when measured with respect to the mean squared error and median absolute error, *moSAIc* outperform all other baseline methods by at least 20%. Moreover, *moSAIc* is also more robust in the sense that the standard deviation is significantly lower than for the other methods.

VI. CONCLUSION

The hybrid fusion method *moSAIc*, which allows to transfer the knowledge inherent in physical degradation models of motors to unknown instances, was proposed. The basic idea is to model the relation between both the geometry and the material properties of a motor and its aging behaviour via neural networks. To validate the effectiveness of the method experiments based on simulated data were conducted. To sum up, the main findings of the study are:

- *moSAIc* outperforms the other baseline methods in terms of accuracy and robustness.
- *moSAIc* reduces the time effort to estimate RUL of an unknown motor in comparison to other state-of-the-art methods.

So far, only an ensemble of physical simulation models has been considered. While their predictions are aggregated via a trainable neural network, the simulation models themselves are static in the sense that their predictions do not change during training. In future work, we will explore model improvements by using trainable experts. Further, it is planned to experiment with relevance propagation [18] in order to make the results of *moSAIc* even more explainable.

REFERENCES

[1] 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. *Mechanical Systems and Signal Processing*, 104:799–834, 2018.

[2] Rafael Gouriveau, Kamal Medjaher, and Noureddine Zerhouni. *From Prognostics and Health Systems Management to Predictive Maintenance 1: Monitoring and Prognostics*. John Wiley & Sons, 2016.

[3] A. Kroll. *Computational Intelligence: Eine Einführung in Probleme, Methoden und technische Anwendungen*. Oldenbourg, München, 2013.

[4] C. Bergs, M. Heizmann, and H. Held. Hybrid modeling approaches with a view to model output prediction for industrial applications. *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, pages 258–263, 2018.

[5] C. Okoh, R.r Roy, J. Mehnen, and L. Redding. Overview of remaining useful life prediction techniques in through-life engineering services. *Procedia CIRP*, 16:158–163, 2014.

[6] X.-S. Si, C.-H. Wang, W.and Hu, and D.-H. Zhou. Remaining useful life estimation–a review on the statistical data driven approaches. *European journal of operational research*, 2011.

[7] X. Li, Q. Ding, and J.-Q. Sun. Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*, 172:1–11, 2018.

[8] Z. Li, D. Wu, C. Hu, and J. Terpenny. An ensemble learning-based prognostic approach with degradation-dependent weights for remaining useful life prediction. *Reliability Engineering & System Safety*, 2017.

[9] R. K. Singleton, E. G. Strangas, and S. Aviyente. Extended kalman filtering for remaining-useful-life estimation of bearings. *IEEE Transactions on Industrial Electronics*, 62(3):1781–1790, 2015.

[10] W. Ahmad, S. A. Khan, M. Islam, and J.-M. Kim. A reliable technique for remaining useful life estimation of rolling element bearings using dynamic regression models. *Reliability Engineering & System Safety*, 2018.

[11] L. Liao and F. Koettig. A hybrid framework combining data-driven and model-based methods for system remaining useful life prediction. *Applied Soft Computing*, 44:191–199, 2016.

[12] M. Seera, C. P. Lim, D. Ishak, and H. Singh. Offline and online fault detection and diagnosis of induction motors using a hybrid soft computing model. *Applied Soft Computing*, 13(12):4493–4507, 2013.

[13] C. Hu, B. D. Youn, and P. Wang. Ensemble of data-driven prognostic algorithms with weight optimization and k-fold cross validation. In *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 1023–1032. American Society of Mechanical Engineers, 2010.

[14] V. Tresp and M. Taniguchi. Combining estimators using non-constant weighting functions. In *Advances in neural information processing systems*, pages 419–426, 1995.

[15] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[16] E. Santecchia, A. Hamouda, F. Musharavati, E. Zalnezhad, M. Cabibbo, M. El Mehtedi, and S. Spigarelli. A review on fatigue life prediction methods for metals. *Advances in Materials Science and Engineering*, 2016, 2016.

[17] R. G. Budynas, J. K. Nisbett, et al. *Shigley's mechanical engineering design*, volume 8. McGraw-Hill New York, 2008.

[18] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.