Forsch Ingenieurwes (2023) 87:479–488
https://doi.org/10.1007/s10010-023-00653-y

ORIGINALARBEITEN/ORIGINALS

Extension of the system boundary of the Digital Twin onto the sensors of the Physical Twin through the introduction of redundant soft sensors

Michel Fett[1] · **Eleanor Turner**[1] · **Richard Breimann**[1] · **Eckhard Kirchner**[1]

Received: 1 November 2022 / Accepted: 27 January 2023 / Published online: 20 March 2023

Abstract

The benefit of Digital Twins depends to a large extent on the quality of the sensor data provided. In many cases, sensor failures are only detected late in operation which can lead to serious consequences. For this reason, one approach to reduce the resulting safety issues is to use redundant sensor systems that monitor the same measureand. However, due to the additional sensors required, this is associated with additional financial and design effort.

In this publication an alternative strategy is presented, which provides a redundant sensor system with the help of soft sensors. Soft sensors use already installed physical sensors to anticipate a new measured variable via algorithms. They are often used to avoid placing sensors in inaccessible locations, but are used here to perform redundant computation of already existing metrics. The sensor data of physical and soft sensors are used as input variables for a Digital Twin. Here, these are compared with each other and can be critically questioned by the twin itself. This makes it possible to extend the system boundary of the Digital Twin to the sensors themselves and provided input variables can be checked for their validity. This allows sensor failures to be detected at an early stage and consequential damage to be averted.

Erweiterung der Systemgrenze des Digitalen Zwillings auf die Sensorik des Physischen Zwillings durch die Verwendung redundanter Softsensoren

Zusammenfassung

Der Nutzen von Digitalen Zwillingen hängt in hohem Maße von der Qualität der bereitgestellten Sensordaten ab. Dabei werden in vielen Fällen Sensorausfälle erst spät im Betrieb erkannt, was zu schwerwiegenden Folgen führen kann. Ein möglicher Ansatz, um die daraus resultierenden Sicherheitsrisiken zu reduzieren, ist daher die Verwendung redundanter Sensorsysteme, welche die gleiche Messgröße erfassen. Aufgrund der größeren Anzahl benötigter physischer Sensoren ist dies allerdings mit zusätzlichen finanziellen und konstruktiven Herausforderungen verbunden.

In dieser Publikation wird ein alternativer Ansatz vorgestellt, welcher Softsensoren nutzt um das redundante Sensorsystem zu erstellen. Softsensoren verwenden bereits integrierte physischen Sensoren, um über Algorithmen eine neue Messgröße zu antizipieren. Sie werden häufig eingesetzt, wenn aufgrund der Unzugänglichkeit von Messstellen keine physischen Sensoren verbaut werden können. Im Rahmen dieser Publikation werden sie jedoch verwendet, um eine redundante Berechnung bereits vorhandener Messgrößen durchzuführen. Die Sensordaten von physischen Sensoren und Softsensoren dienen als Eingangsgrößen für einen Digitalen Zwilling. Dieser vergleicht die Werte miteinander und ist so imstande diese kritisch zu hinterfragen. Damit ist es möglich, die Systemgrenze des Digitalen Zwillings auf die Sensoren selbst zu erweitern, Sensorausfälle frühzeitig zu erkennen und Folgeschäden abzuwenden.

✉ Michel Fett
michel.fett@tu-darmstadt.de

[1] Institute for Product Development and Machine Elements, TU Darmstadt, Otto-Berndt-Straße 2, 64287 Darmstadt, Germany

1 Introduction

The usability of Digital Twins is highly dependent on the quality of the sensor data provided [1, 2]. Malfunctions and failures of sensors can not only disturb the monitoring of the physical twin through the Digital Twin, but also lead to serious malfunctions of the physical twin, due to being controlled by the feedback of the Digital Twin. This problem can be mitigated by the integration of redundant sensors, however, the installation of several physical sensors leads to additional financial and constructive expenditure. For this reason, this publication examines the question of how redundancy can be built up by introducing soft sensors, which calculate the quantity of interest instead of measuring them. This is to be used to detect sensor failures with the resulting extended system boundary of the Digital Twin of a rolling bearing test bench. In the long term the results can be applied to different types of Digital Twins, such as test benches, production machines or customer products.

2 State of the art

2.1 The Digital Twin concept

Due to the high degree of novelty of the concept of Digital Twins, a multitude of partly contradictory understandings and definitions exist. In order to create a uniform understanding the following definition of a Digital Twin applies in the context of this publication:

A digital twin is a digital representation of a real product instance (physical twin). The representation uses models that are fed with real-time data, e.g. by sensors installed on the physical twin, to simulate its behavior. The simulation results are then fed back into the physical world via a bidirectional connection and made use of there [3–6].

In order to distinguish the digital twin from its environment, it is defined by a system boundary [7, 8]. The system boundary of the digital twin is based on the system boundary of the physical twin [9] and is drawn according to the intended use case [10]. The system boundary is primarily used to describe whether the respective digital twin is a single unit or an aggregated system of subsystems [8, 9, 11]. A more detailed consideration from a system theoretical view to identify an optimum of the system boundary between twin and environment does not take place. It is therefore not defined where the system boundary should be drawn and whether the sensors should be included as an information interface between the physical and Digital Twin. However, the strong dependence of Digital Twins on correct sensor data [1, 2] leads to the conclusion that, in general, the behavior of sensors is not described by Digital Twins. For this reason, malfunctions and failures of sensors can disturb the monitoring of the physical twin by the Digital Twin. The calculations and simulations of the Digital Twin are based on sensor data and can be used to regulate or control the physical twin. With incorrect input values this can lead to serious malfunctions of the latter.

2.2 Soft sensors

In contrast to physical sensors, soft sensors describe an algorithm which calculates the quantity of interest in a product instead of measuring it with a physical entity [12]. The algorithms can be based on physical, empirical or data-driven models. The latter include machine learning approaches [13]. These algorithms use the measurement data of various physical sensors integrated in the product as input variables to calculate the quantity of interest. The thus calculated variable of the soft sensor neither needs a local proximity or the same measured variable as the installed physical sensors. However, it must be ensured that a correlation is given and the soft sensor $S_{\text{softsensor}}$ can be represented in the form of a function of the physical sensors $S_{i,\text{physical}}$ shown in Eq. 1 [14].

$$S_{\text{softsensor}} = f\left(S_{1,\text{physical}}, \dots, S_{n,\text{physical}}\right) \tag{1}$$

In this way, data can be simulated for locations where placing physical sensors is difficult due to design restrictions or operating conditions. In the literature, soft sensors are also called virtual sensors [14–17], extended Kalman filters [18] or, in the context of control engineering, state observers, estimators or predictors [19]. Some authors describe the soft sensor as a Digital Twin of a physical sensor [13, 14, 19]. Others use Digital Twins as comprehensive soft sensors of complex systems [20, 21]. In order to do justice to the previously mentioned definition of the term Digital Twin, the terms Digital Twin and soft sensor are not used synonymously in a general way. However, assuming that a physical counterpart exists in the product, a soft sensor will be regarded as a Digital Twin of the corresponding physical sensor in the scope of this publication. Soft sensors can, just like physical sensors, be used as input variables for a Digital Twin [22–24].

2.3 Redundancy through soft sensors

One possible approach to mitigate the effects of sensor failures is the redundant implementation of physical sensors. Redundancy can be differentiated into hot and cold redundancy. In the case of hot redundancy, two or more systems work in parallel at any given time, but can also fulfil the task on their own. In cold redundancy only one system is operated at a time. The replacement system is kept on standby and activated only in case of failure of the first system [25]. Since a redundant sensor system records the same measured variables and compares the measured values, it is classified as hot redundancy. Discrepancies between the measured values allow sensor failures or misbehavior to be identified and appropriate measures to be initiated. A redundant design of physical sensors is associated with a number of disadvantages. One drawback is increased cost, due to the usage of more sensors. Furthermore, the integration of sensors is accompanied by an increased design effort in order to meet requirements for functional fulfilment or construction space.

One alternative strategy is a data driven approach to validate sensor data and detect sensor failures like the implementation of redundant soft sensors. Approaches in recent literature utilize the idea that soft sensors can not only be used to determine unknown measured variables, but also to perform a calculation of variables already measured from existing physical sensors. There are multiple ways to calculate the required soft sensor values. The Input can be either the last few values of the respective physical sensor [17] or the values all sensors except the respective physical sensor [16]. Another alternative is to identify reliable and unreliable sets of sensors in advance. Soft sensors are then created for the unreliable ones from the data of the reliable ones [17].

Using these methods, a redundant sensor system is built without the need to integrate additional physical sensors. This is referred to in the literature as analytical [2, 17] or virtual redundancy [13, 14, 16, 19]. In the context of redundancy alignment, the measured values of the physical sensors and the predicted values of the soft sensors can be compared and deviations can be determined. The deviations are utilized to verify the measured data from the physical sensors [2]. This way the condition of the physical sensor can be monitored and failures can be detected [16, 19, 26]. Soft sensors can also be used as backups to replace the physical sensors in case of a deviation [13, 17]. In this way, process stability can be improved [14].

Redundant soft sensors cannot be used on their own, but must be processed and interpreted by a suitable data dissemination system. One possible approach is processing by a Digital Twin, for which some initial applications can be found in the literature.

Staudter et al. [2] consider soft sensors in a holistic investigation of data-induced conflicts in Digital Twins. In this context, soft sensors are used to build analytical redundancy and are used within a holistic framework to identify conflicts. However, an isolated consideration of the practical generation and use is not the focus of this publication.

He et al. [26] use a multi-block PLS approach to model the system with the aim of monitoring the process through a digital twin. In addition to the (soft) sensors, the actuators and possible process faults are modelled in order to identify the cause of a process fault when it is detected. However, the identification of the process error itself does not take place through the soft sensors.

Darvishi et al. [17, 27] in contrast, use soft sensors specifically to identify sensor faults. For this purpose, soft sensors are used to calculate sensor values from unreliable physical sensors and the occurring deviations are compared. In a first approach, the soft sensors consist of a predictor that takes into account the temporal development and an estimator that uses the remaining reliable sensors [27]. In a follow-up publication, an algorithm is used which takes both perspectives into account at the same time [17]. In each case, multilayer perceptron (MLP) neural networks are used to generate the soft sensors, using 70% [17] and 85% [27] of the available data to train each. The need for large amounts of data is explicitly mentioned as a limiting factor of neural networks. Darvishi et al. further state that the inclusion of correlations between the sensors can increase the performance compared to the use of all sensors simultaneously [17].

In this contribution, the concept of soft sensors is considered in isolated fashion in order to avoid implications that arise from interactions with related systems. For this purpose, the effects of the system boundary of Digital Twins are first examined and the soft sensors are delimited.

In the practical side, redundant soft sensors are used to identify sensor errors. Neural networks are deliberately avoided, as they are not always suitable due to the lack of transparency, as well as the high demand for data. Instead, various alternative machine learning methods are used. To increase the performance of the soft sensors, the correla-

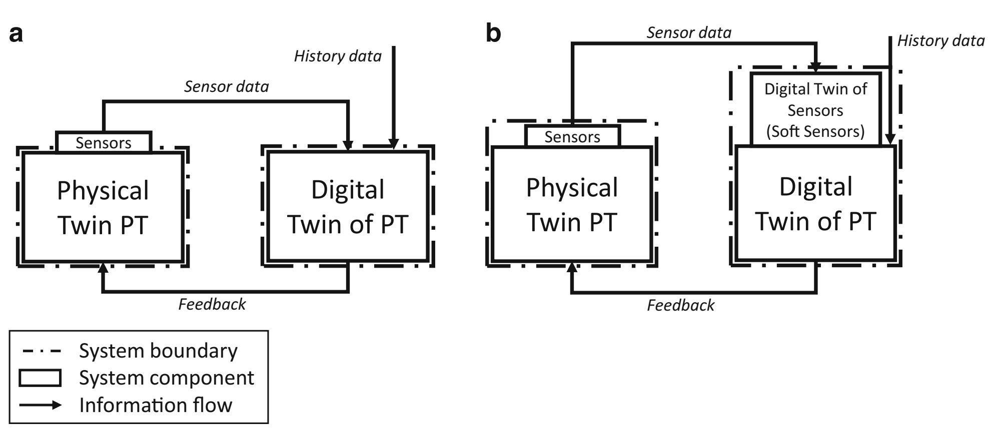

Fig. 1 Schematic presentation of the Digital Twin concept with the sensor outside (**a**) or inside (**b**) the system boundary

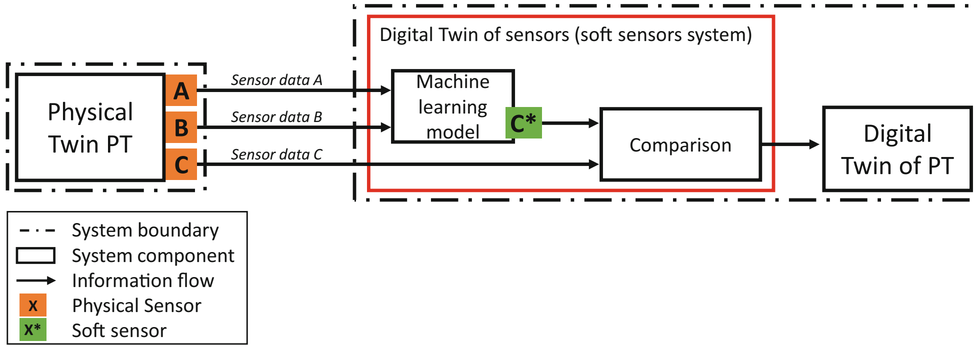

Fig. 2 Schematic presentation of the information flow from the physical twin to the Digital Twin including the processing by soft sensors

tions of the individual physical sensors are included. The entire procedure for creating the soft sensors is replicable and is presented in a methodical procedure model.

3 Extension of the system boundary of a Digital Twin

As described in Sect. 2.1, a Digital Twin is a virtual representation of a physical counterpart (physical twin). Real-time data from the physical twin is captured by sensors and transferred to the Digital Twin. The behavior of the sensors is often not represented by Digital Twins. Instead, sensor data is taken as given. This corresponds to a system bound-

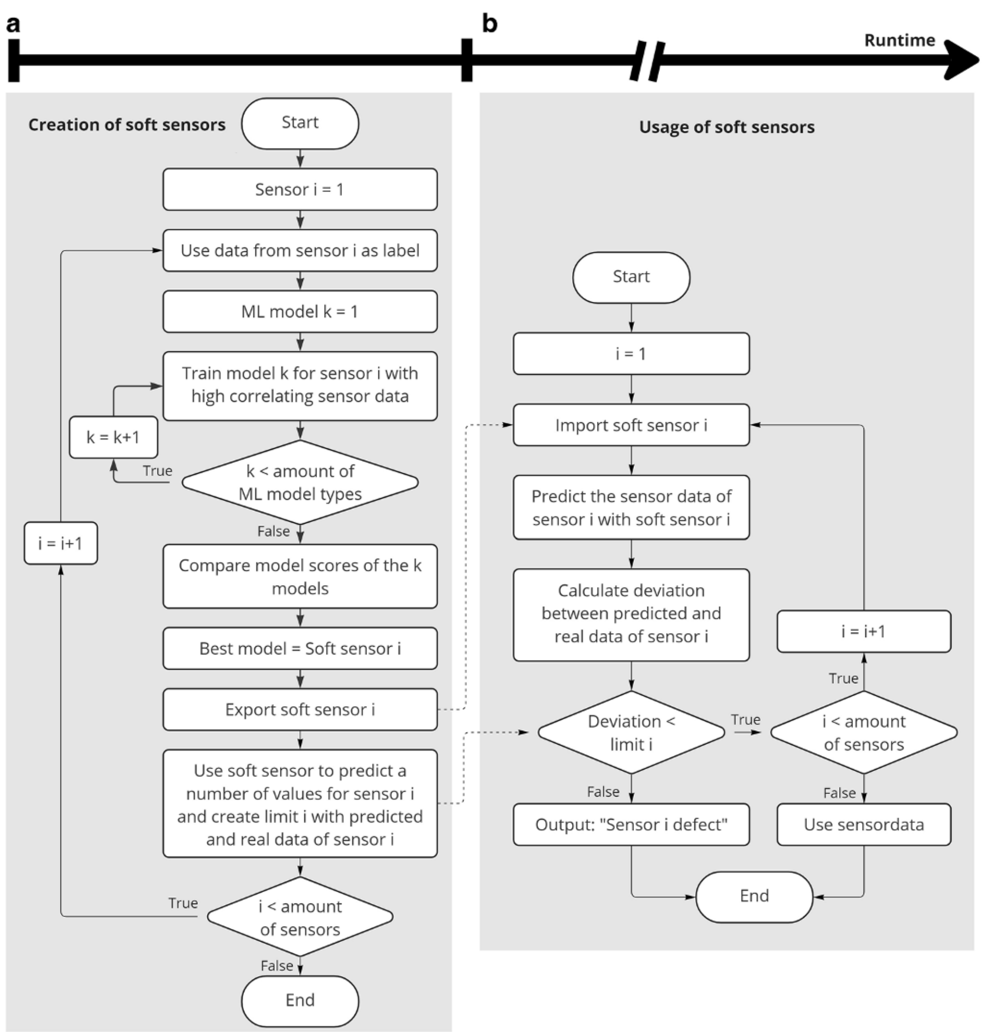

Fig. 3 Flowchart of the algorithm for the creation (**a**) and usage (**b**) the soft sensors

ary that is drawn around the physical twin, but excludes the sensors. This can be seen in Fig. 1a.

The ability to simulate and anticipate the behavior of physical sensors through a redundant soft sensor system extends this system boundary to include the physical sensors as shown in Fig. 1b. In this way, the Digital Twin can verify sensor data and detect sensor failures and malfunctions at an early stage. This helps prevent erroneous inputs to the Digital Twin and potentially harmful behavior.

The information flow from the physical twin to the Digital Twin is of particular interest for the consideration and expansion of the system boundary through soft sensors. The soft sensor system is located in this information flow. Figure 2 schematically shows the information flow in detail.

The measured values collected by the physical sensors A, B and C are fed into the soft sensor system. In this system, the soft sensor C* is initially created from the measured values of the physical sensors A and B and later supplied by them. The values of the physical sensor C and the soft sensor C* are compared before the information is fed into the Digital Twin. For reasons of clarity, only the creation and comparison of sensor C and soft sensor C* is shown in the illustration. In the same way, this must also be done for sensors A and B.

In Fig. 2, the soft sensor system, which is the focus of this publication, is highlighted in red. It is not relevant for the scope of this publication how the information on the physical twin is collected in the form of physical sensors or how the information obtained is utilized in the Digital Twin.

To create the soft sensors, different machine learning models are investigated. Neural networks are deliberately excluded due to their lack of transparency. The algorithms for the creation and usage of the soft sensors were realized in Python. To clarify the procedure a flow chart according to DIN 66001 [28] is created, which contains both the creation and use of the soft sensors. The flow chart is shown in Fig. 3, the two algorithms are described in the following sections.

3.1 Creation of soft sensors

New soft sensors are created for each individual use case, this is shown in in Fig. 3a. For this, data is collected at the beginning of the runtime, to create a soft sensor $S_{i,\,\text{softsensor}}$ for each physical sensor $S_{i,\,\text{physical}}$. The soft sensors are based on correlations between the measured values of the individual physical sensors. In order to utilized these correlations in the best possible way, different machine learning models are investigated. In the context of this work, the models linear regression, polynominal regression, random forrest regressor and decision trees were used. For each sensor i, the k different model types are created.

All physical sensors are taken as input for the model, with the exception of the particular physical sensor i for which the respective redundancy is to be generated. The latter data are used as labels to train the model. Equation 2 shows the creation of the models using the example of linear regression with weights $\alpha_{i,j}$ and intercept variables β_i.

$$\begin{bmatrix} S_{1,\text{softsensor}} \\ \vdots \\ \vdots \\ S_{n,\text{softsensor}} \end{bmatrix} = \begin{bmatrix} 0 & \alpha_{1,2} & \dots & \alpha_{1,n} \\ \alpha_{2,1} & 0 & & \vdots \\ \vdots & & \ddots & \alpha_{n-1,n} \\ \alpha_{n,1} & \dots & \alpha_{n,n-1} & 0 \end{bmatrix} \begin{bmatrix} S_{1,\text{physical}} \\ \vdots \\ \vdots \\ S_{n,\text{physical}} \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \vdots \\ \vdots \\ \beta_n \end{bmatrix} \tag{2}$$

Then, using data collected up to this point, the model scores of the different models are calculated. For this purpose, the respective score functions of the different model types of the Scikit-learn environment are used [29–32]. The best possible model for each individual sensor is selected and used as the core of the corresponding soft sensor i. Depending on the correlations between the measured values, the individual soft sensors can only predict the measured values with a certain accuracy. For this reason, an individual failure criterion must be determined for each soft sensor. To do this, each soft sensor predicts a value for the corresponding physical sensor. These values are compared and the normalized deviation over the value range of the data is calculated. This is shown in Eq. 3.

$$\frac{\left|S_{i,\text{physical}} - S_{i,\text{softsensor}}\right|}{S_{i,\text{physical,max}} - S_{i,\text{phyisical,min}}} = \Delta S_n \tag{3}$$

The failure criterion is then determined using the average deviation over the set of measured values. After a soft sensor is created for each physical sensor, the algorithm ends. The resulting models for each sensor, as well as the failure criteria in the form of the limit, exported for the later usage.

3.2 Usage of soft sensors

The created soft sensors are applied during the rest of the runtime of the physical twin operation. For this purpose the saved soft sensors in the form of models of each physical sensor as well as the individual failure criteria are loaded in for the utilization. This is shown in Fig. 3b.

For each time step, the data of the physical sensors to be examined is imported. The model of each soft sensor is fed with the sensor data of the other sensors. Depending on

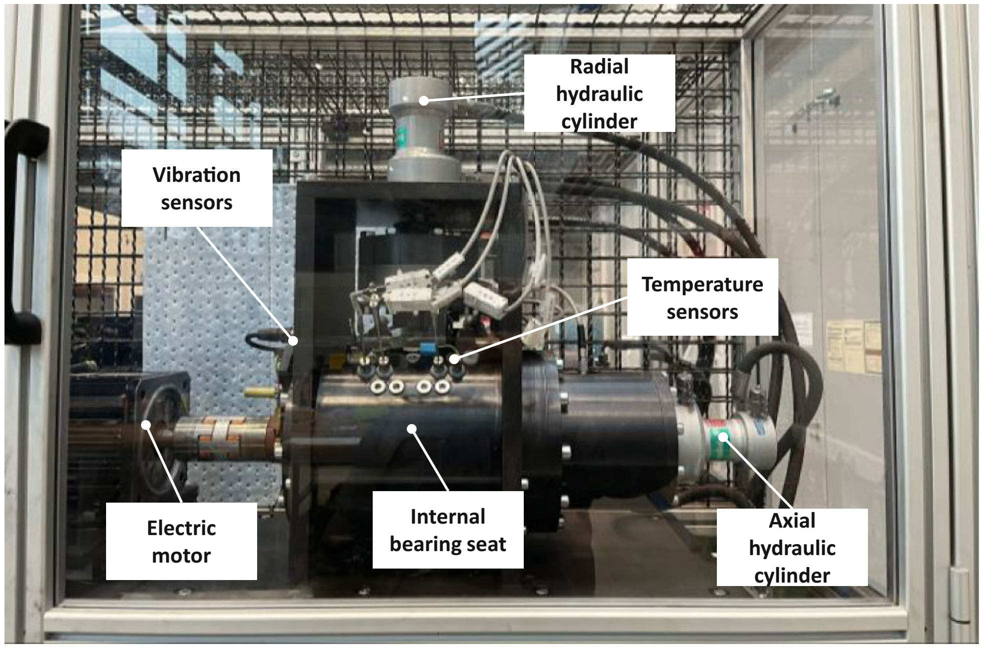

Fig. 4 Rolling bearing test bench "Athene" of the TU Darmstadt

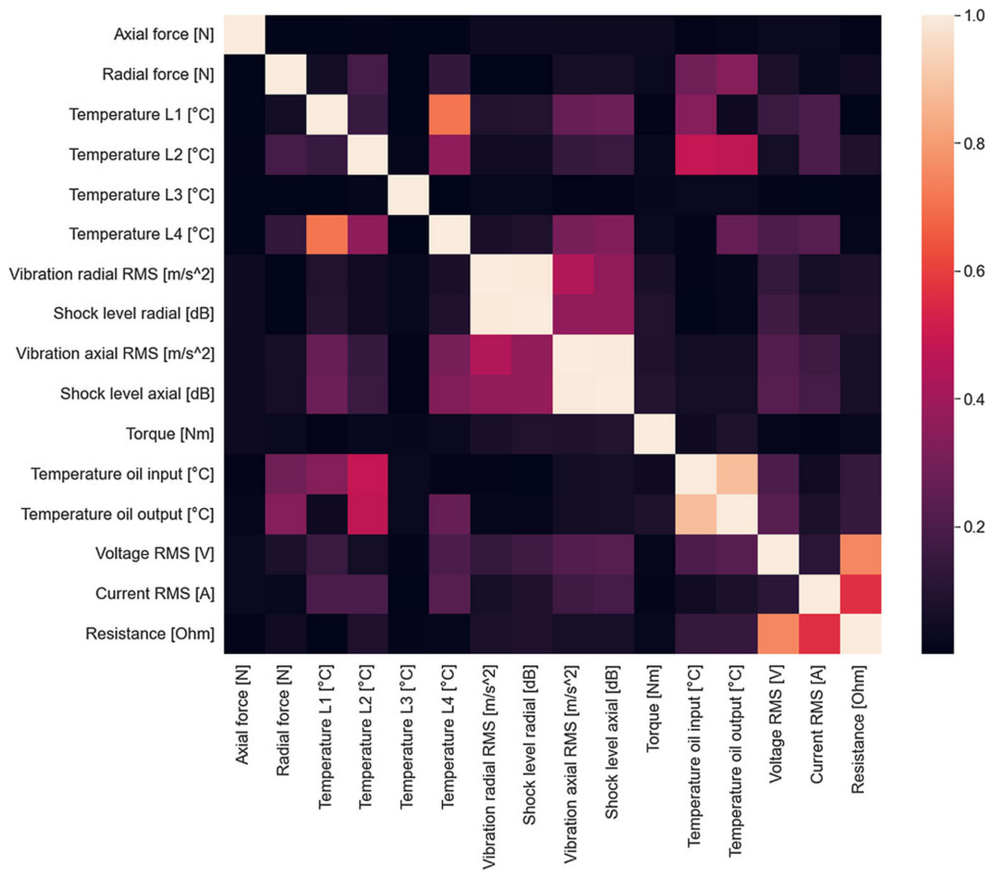

Fig. 5 Correlation matrix of the sensors installed on the test bench

the respective trained weights, these are used to predict the value of the corresponding soft sensor.

The predicted values of the soft sensor are then compared with the values of the physical sensors and the deviations are calculated. If these deviations exceed the previously defined limits, a warning is issued and appropriate measures are initiated. If the deviation of all measured variables is within their individual limits, the data of the entire time step is released for use and passed on to the actual Digital Twin of the superordinate system or product.

4 Application on a test bench

4.1 Introduction of the test bench

At the Department of Product Development and Machine Elements at TU Darmstadt, rolling bearings can be examined on the rolling bearing test bench "Athene" (see Fig. 4; [33–35]). This test bench can apply loads to rolling bearings at up to 8000 rpm in four test chambers. For this purpose, two hydraulic cylinders apply up to 40 kN axially and radially. The reactions of the bearing in the form of radial and axial vibration and impact levels are measured via sensors. Furthermore, sensors for speed, torque, axial and radial force are installed and the voltage and current is recorded. A listing of all sensors can be seen in Fig. 5 or Table 1. Since the test bench is already equipped with extensive sensor systems, the corresponding sensor data can be used to feed a Digital Twin. In contrast to consumer products, no integration of further sensor technology is necessary.

Table 1 Model score and deviations of selected sensors on the test bench

Sensor	Best ML model type	Model quality score	Average deviation [%]
Axial force	Lr	0.00006964	24.22
Radial force	Rfr	0.6024	7.502
Temperature L1	Rfr	0.7832	44.84
Temperature L2	Rfr	0.6910	44.54
Temperature L3	Dt	0.9999	46.96
Temperature L4	Rfr	0.8054	26.87
Vibration radial RMS	Rfr	0.9964	0.9726
Shock level radial	Pr	0.9774	1.444
Vibration axial RMS	Rfr	0.9992	1.870
Shock level axial	Pr	0.9934	1.188
Torque	Pr	0.0110	23.89
Temperature oil input	Rfr	0.9563	13.56
Temperature oil output	Rfr	0.9279	13.39
Voltage RMS	Pr	0.9999	<0.001
Current RMS	Pr	0.9999	1.719
Resistance	Pr	0.9999	3.274

Since the necessary IT infrastructure for real-time processing of the data is currently still under development, stored data sets are used. The data sets are not loaded as a whole but row wise in discrete time steps of one second, so a real time data input is simulated and the transferability of the results is increased. This applies to both the Digital Twin and the soft sensors implemented later.

In the following, a soft sensor system is created and applied for the sensors of this test bench. As illustrated in Fig. 2, the digital twin of the test bench itself is not the focus of this work and will therefore not be discussed further below.

4.2 Creation of the soft sensors

Data from a long-term test was used to create/train the soft sensors. As described in Sect. 3.2, the first hour of the experiment was used to create the models of the soft sensors. The inconsistent run-in phase was deliberately excluded. Figure 5 shows a correlation matrix of the physical sensor data during the training period. The brighter a single cell of the matrix is, the higher the correlation of the two corresponding physical sensors. It is emphasized that the correlations are differently pronounced.

The data of the training period is first split using a train test split (0.75; 0.25). The training data is used to create four different machine learning models (linear regression (lr), polynomial regression (pr), random forrest regressor (rfr) and decision trees (dt)) for each of the physical sensors and to determine the model scores. The model with the highest model score is then automatically selected and used as the basis for the corresponding soft sensors.

The test data is then used to determine the deviations between data of the physical sensors and the predicted values of the soft sensors. For each sensor 10,000 random data points are used and the average deviation is determined. The deviations are then normalized to the range of values of the data.

Table 1 shows the most suitable ML models for each sensor, their model score and the average deviation of the predictions that can be obtained with them. The normalization to the value range of the data leads to sensors with quasi-static measured values with a small value range showing high average deviations. The temperatures fluctuate with about ± 1 °C around a constant value range, so that small deviations of 0.5 °C already lead to the high deviations shown here. The model scores and average deviations of the soft sensors for axial force and torque are also striking. This may be due to defective sensors during data collection, but requires closer investigation.

For the purposes of this example, the failure criteria for each sensor are set as ten times the average deviation.

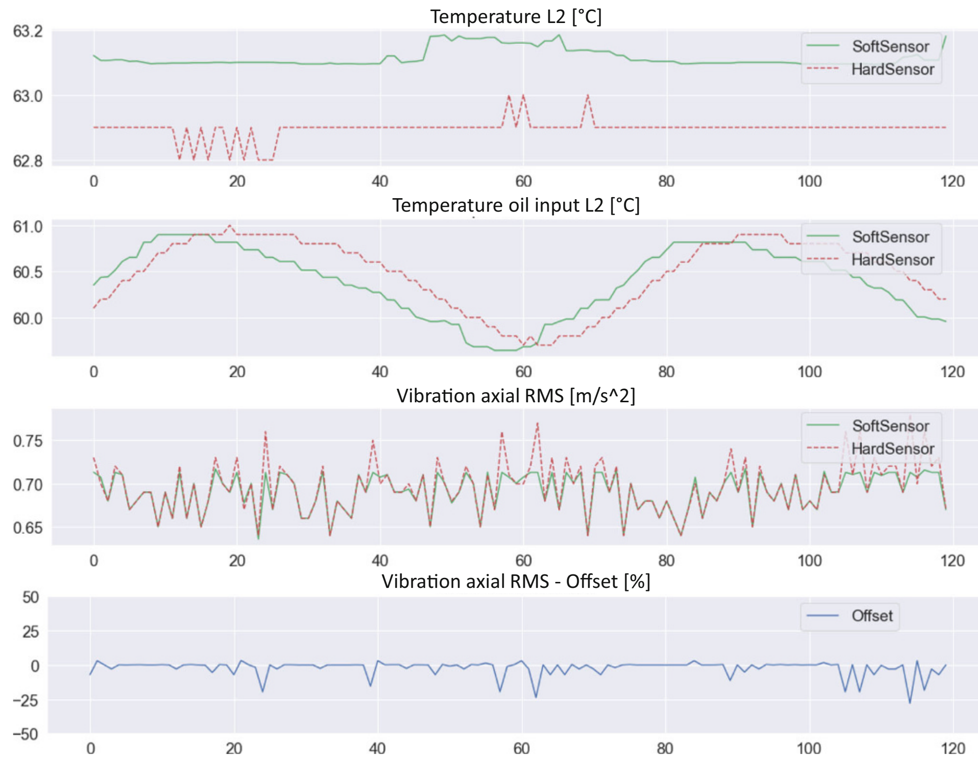

Fig. 6 Dashboard of calculated values of the soft sensors (normal operation)

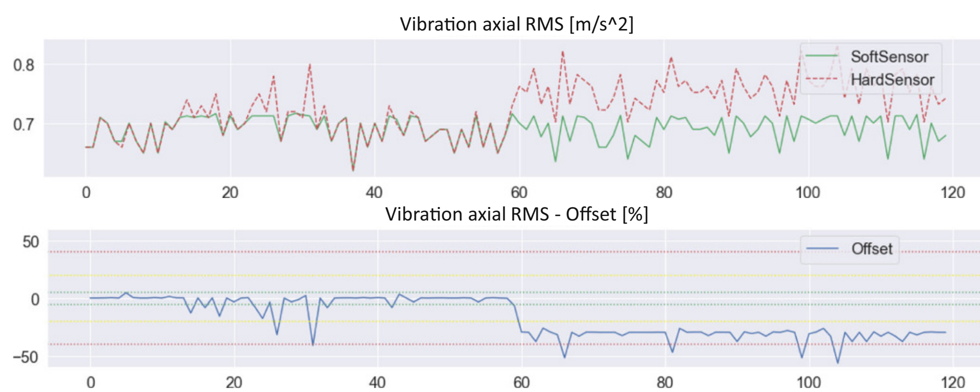

Fig. 7 Dashboard of calculated values of the soft sensors (sensor failure)

4.3 Usage of the soft sensors

For each physical sensor, the corresponding previously created model is loaded. This is fed with the corresponding physical sensor data and thus the values of the respective soft sensors are calculated based on the weights of the model.

The calculated values of the soft sensor are then compared with the real measured values of the physic sensor and the relative deviation is calculated. This is then compared with the limit values derived from the mean deviation. If this limit value is exceeded for a certain time, an error message is displayed.

Figure 6 shows an exemplary dashboard with three soft sensors in comparison with the physical. For the axial vibration the deviation is determined.

4.4 Detection of sensor failures

To evaluate the function of the soft sensors developed, a physical sensor is artificially damaged. For this purpose, an offset is applied to the physical sensor of the axial vibration from second 60 of the observed time range. This leads to the violation of the failure criterion of the maximum permissible deviation in form of the yellow line. The sensor of the axial vibration is indicated as defective. This is shown in Fig. 7.

5 Discussion

It was shown that it is possible to build a redundant sensor system using soft sensors. The sensor behavior is modeled, which extends the system boundary of the Digital Twin onto the sensors and makes it possible to detect sensor errors. The theoretical concepts were applied exemplarily at a rolling bearing test bench of the TU Darmstadt.

At the current time, the IT infrastructure required for real-time processing is still under development, which is why the exemplary implementation was only possible with existing data sets. Care was taken to process these data sets row wise in one-second increments, which increases the transferability of the results, but this still needs to be finally evaluated. Furthermore, the models are trained for individual use cases. In the future, the use of measurement data from four-quadrant experiments can be used, to create soft sensors, which can be used with various use cases at different operating points. Furthermore, the choice of failure criteria should be reconsidered. Although the derivation of the selection criteria from the mean deviations during the training leads to satisfying results, the interdependencies of the different soft sensors cannot be covered sufficiently. Even if the evaluation an the rolling bearing test bench shows that the soft sensors fulfill their purpose, the determination of the failure criteria from the average deviations must be critically questioned and extensively verified.

Acknowledgements The support of this contribution through the FVA889II project (funding number: 22467 BG), by the German Federal Ministry for Economic Affairs and Climate Action (Bundesministerium für Wirtschaft und Klimaschutz BMWK) and the AiF e. V. is gratefully acknowledged.

Supported by:

Federal Ministry
for Economic Affairs
and Climate Action

on the basis of a decision
by the German Bundestag

Funding Open Access funding enabled and organized by Projekt DEAL.

Conflict of interest M. Fett, E. Turner, R. Breimann and E. Kirchner declare that they have no competing interests.

References

1. Röhm B, Emich B, Anderl R (2021) Approach of simulation data management for the application of the digital simulation twin. Proc CIRP 100:421–426. https://doi.org/10.1016/j.procir.2021.05.098
2. Staudter GA, Öztürk T, Martin DM, Hartig J, Molitor DA, Hoppe F, Anderl R, Groche P, Pelz PF, Weigold M (2021) An approach for mastering data-induced conflicts in the digital twin context. IJPLM 13:25. https://doi.org/10.1504/IJPLM.2021.10038666
3. Stark R, Anderl R, Thoben K-D, Wartzack S (2020) WiGeP-Positionspapier: „Digitaler Zwilling". Z Wirtsch Fabrikbetr 115:47–50. https://doi.org/10.3139/104.112311
4. Schleich B, Anwer N, Mathieu L, Wartzack S (2017) Shaping the digital twin for design and production engineering. CIRP Ann 66:141–144. https://doi.org/10.1016/j.cirp.2017.04.040
5. Wilking F, Schleich B, Wartzack S (2021) Digital Twins—Definitions, classes and business scenarios for different industry sectors. Proc Des Soc 1:1293–1302. https://doi.org/10.1017/pds.2021.129
6. Czwick C, Martin G, Anderl R, Kirchner E (2020) Cyber-Physische Zwillinge. Z Wirtsch Fabrikbetr 115:90–93. https://doi.org/10.3139/104.112310
7. Winkler P, Gallego-García S, Groten M (2022) Design and simulation of a digital twin mobility concept: an electric aviation system dynamics case study with capacity constraints. Appl Sci 12:848. https://doi.org/10.3390/app12020848
8. West S, Stoll O, Meierhofer J, Züst S (2021) Digital twin providing new opportunities for value co-creation through supporting decision-making. Appl Sci 11:3750. https://doi.org/10.3390/app11093750
9. Nasirahmadi A, Hensel O (2022) Toward the next generation of digitalization in agriculture based on digital twin paradigm. Sensors. https://doi.org/10.3390/s22020498
10. Singh S, Weeber M, Birke KP (2021) Implementation of battery digital twin: approach, functionalities and benefits. Batteries 7:78. https://doi.org/10.3390/batteries7040078
11. Loaiza JH, Cloutier RJ (2022) Analyzing the implementation of a digital twin manufacturing system: using a systems thinking approach. Systems 10:22. https://doi.org/10.3390/systems10020022
12. Fortuna L, Graziani S, Rizzo A, Xibilia MG (2007) Soft sensors for monitoring and control of industrial processes. Springer, London
13. Pelz P, Dietrich I, Schänzle C, Preuß N (2018) Towards digitalization of hydraulic systems using soft sensor networks. RWTH Aachen University, Aachen
14. Peniak P, Rástočný K, Kanáliková A, Bubeníková E (2022) Simulation of virtual redundant sensor models for safety-related applications. Sensors. https://doi.org/10.3390/s22030778
15. Zang Y, Qian Y, Wang H, Xu A, Zhou X, Sheng G, Jiang X (2021) A novel optical localization method for partial discharge source using ANFIS virtual sensors and simulation fingerprint in GIL. IEEE Trans Instrum Meas 70:1–11. https://doi.org/10.1109/TIM.2021.3097856
16. Mattera CG, Quevedo J, Escobet T, Shaker HR, Jradi M (2018) A method for fault detection and diagnostics in ventilation units using virtual sensors. Sensors. https://doi.org/10.3390/s18113931
17. Darvishi H, Ciuonzo D, Eide ER, Rossi PS (2021) Sensor-fault detection, isolation and accommodation for digital twins via modular data-driven architecture. IEEE Sensors J 21:4827–4838. https://doi.org/10.1109/JSEN.2020.3029459
18. Pfeiffer B-M, Oppelt M, Leingang C (2019) Evolution of a digital twin for a steam cracker. In: 2019 24th IEEE International Confer-

ence on Emerging Technologies and Factory Automation (ETFA). IEEE, pp 467–474 https://doi.org/10.1109/ETFA.2019.8869449
19. Jiang Y, Yin S, Dong J, Kaynak O (2021) A review on soft sensors for monitoring, control, and optimization of industrial processes. IEEE Sensors J 21:12868–12881. https://doi.org/10.1109/JSEN.2020.3033153
20. Mehlan FC, Nejad AR, Gao Z (2022) Digital twin based virtual sensor for online fatigue damage monitoring in offshore wind turbine Drivetrains. J Offshore Mech Arct Eng. https://doi.org/10.1115/1.4055551
21. Toso F, Torchio R, Favato A, Carlet PG, Bolognani S, Alotto P (2021) Digital twins as electric motor soft-sensors in the automotive industry. In: 2021 IEEE International Workshop on Metrology for Automotive (MetroAutomotive). IEEE, pp 13–18 https://doi.org/10.1109/MetroAutomotive50197.2021.9502885
22. Rodriguez B, Sanjurjo E, Tranchero M, Romano C, Gonzalez F (2021) Thermal parameter and state estimation for digital twins of E-powertrain components. IEEE Access 9:97384–97400. https://doi.org/10.1109/ACCESS.2021.3094312
23. Lopez PC, Udugama IA, Thomsen ST, Roslander C, Junicke H, Mauricio-Iglesias M, Gernaey KV (2020) Towards a digital twin: a hybrid data-driven and mechanistic digital shadow to forecast the evolution of lignocellulosic fermentation. Biofuels Bioprod Bioref 14:1046–1060. https://doi.org/10.1002/bbb.2108
24. Lafarge R, Hütter S, Tulke M, Halle T, Brosius A (2021) Data based model predictive control for ring rolling. Prod Eng Res Devel 15:821–831. https://doi.org/10.1007/s11740-021-01063-1
25. Kirchner E (2020) Werkzeuge und Methoden der Produktentwicklung. Springer, Berlin, Heidelberg
26. He R, Chen G, Dong C, Sun S, Shen X (2019) Data-driven digital twin technology for optimized control in process systems. ISA Trans 95:221–234. https://doi.org/10.1016/j.isatra.2019.05.011
27. Darvishi H, Ciuonzo D, Rossi PS (2021) Real-time sensor fault detection, isolation and accommodation for industrial digital twins. In: 2021 IEEE International Conference on Networking, Sensing and Control (ICNSC). IEEE, pp 1–6 https://doi.org/10.1109/ICNSC52481.2021.9702175
28. DIN 66001: DIN 66001:1983-12, Informationsverarbeitung; Sinnbilder und ihre Anwendung. Beuth Verlag GmbH, Berlin, vol.
29. sklearn Documentation: Documentation RandomForrestRegressor. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html#sklearn.ensemble.RandomForestRegressor.score. Accessed 29.12.2022
30. sklearn Documentation: Documentation LogisticRegression. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#sklearn.linear_model.LogisticRegression.score. Accessed 29.12.2022
31. sklearn Documentation: Documentation LinearRegression. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html#sklearn.linear_model.LinearRegression.score. Accessed 29.12.2022
32. sklearn Documentation: Documentation DecisionTreeRegressor. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html#sklearn.tree.DecisionTreeRegressor.score. Accessed 29.12.2022
33. Harder A, Zaiat A, Becker-Dombrowsky FM, Puchtler S, Kirchner E (2022) Investigation of the voltage-induced damage progression on the raceway surfaces of thrust ball bearings. Machines 10:832–842. https://doi.org/10.3390/machines10100832
34. Martin G, Becker FM, Kirchner E (2022) A novel method for diagnosing rolling bearing surface damage by electric impedance analysis. Tribol Int 170:107503. https://doi.org/10.1016/j.triboint.2022.107503
35. Schirra T, Martin G, Puchtler S, Kirchner E (2021) Electric impedance of rolling bearings—consideration of unloaded rolling elements. Tribol Int 158:106927. https://doi.org/10.1016/j.triboint.2021.106927