Domain Adaptation *via* Simulation Parameter and Data Perturbation for Predictive Maintenance*

Kiavash Fathi[1,2,**], Fabio Corradini[3,**], Marcin Sadurski[1,***], Marco Silvestri[3,***],
Marko Ristin[1], Afrooz Laghaei[1],
Davide Valtorta[4], Tobias Kleinert[2], Hans Wernher van de Venn[1]

[1] *Institute of Mechatronic Systems, Zurich University of Applied Sciences, 8400 Winterthur, Switzerland*
fath@zhaw.ch, sadu@zhaw.ch, rist@zhaw.ch, lagf@zhaw.ch, vhns@zhaw.ch

[2] *Chair of Information and Automation Systems for Process and Material Technology, RWTH Aachen University, 52064 Aachen, Germany*
kiavash.fathi@rwth-aachen.de, kleinert@plt.rwth-aachen.de

[3] *Department of Innovative Technologies, University of Applied Sciences of Southern Switzerland, SUPSI, Via la Santa 1, 6962, Lugano-Viganello, Switzerland*
fabio.corradini@supsi.ch, marco.silvestri@supsi.ch

[4] *SAECON Sagl - Safety and Engineering Consulting, via Grancia 1, 6926 Montagnola, Switzerland*
valtorta@saecon.net,

Abstract

Conventional data-driven predictive maintenance (PdM) solutions learn from samples of run-to-failures (R2F) to estimate the remaining useful life of an asset. In practice, such samples are scarce or completely missing. Simulation models can be oftentimes used to generate R2F samples as a replacement. However, due to the complexity of the assets, creating realistic simulation models is tedious, or even impossible. Thus generated R2F data cannot be used to create reliable PdM models as they are highly sensitive to noises in the sensors or small deviations in system working condition. To address this, we present a new concept of simulation data generation based on supervised domain adaptation for a regression problem where the remaining useful life (RUL) or the health index (HI) of the system is predicted. Apart from input and output domain shift, given the changes in the dominant failing component and its degradation process, the function mapping sensor readings to RUL and/or HI is also prone to changes and thus is a random process itself. Therefore, we aim to generate R2F training data from different working conditions and possible failure types using parameter randomization in the simulation model. By sampling from various configurations within simulation model's parameter space, we ensure that the trained data-driven PdM model's performance is not impacted by the initial conditions and/or the changes in the degradation of the system's condition indicators. Our results indicate that the model is robust to signal reading manipulation and showcases a more spread-out feature importance across a wider range of sensor readings for making predictions. We also demonstrate its applicability on the real-world factory physical system whilst our models were mainly trained using generated data.

1. Introduction

Accurate prediction of a production asset's health state enables effective implementation of a predictive maintenance (PdM) solution. Such a solution can help reduce both the cost and occurrence of unscheduled maintenance of the targeted production asset (Cui, Du, & Hawkes, 2012; Rahat et al., 2022). With the advancements in sensor technologies, data acquisition and analysis, numerous PdM solutions predict the remaining useful life (RUL) and/or the health index (HI) by either date-driven models, model-based models or hybrid of the two.

* This research was supported by Innosuisse - Swiss Innovation Agency, Innosuisse Grant no: **58020.1 IP-ENG**, entitled **MaintAIn - AI supported hybrid Predictive Maintenance**.
** These authors contributed equally.
*** These authors contributed equally.

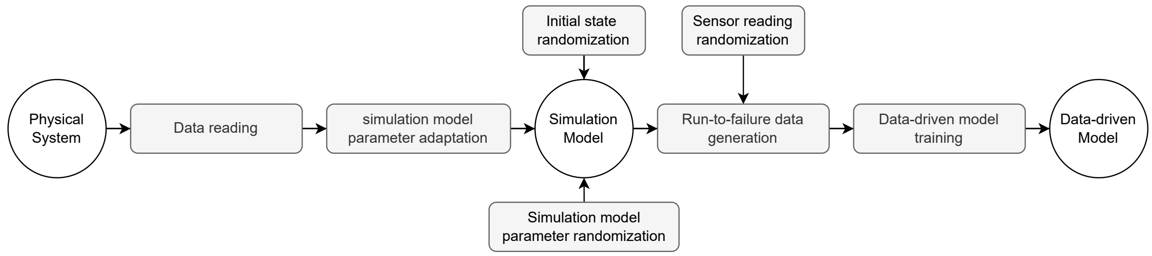

Figure 1. Overview of the proposed method

The *data-driven models* use historical data to train the model of RUL, so their quality depends on the data quality. It is thus of utmost importance that the training data meets the minimum data quality requirements (Liu, Wang, Ma, Yang, & Yang, 2012). However, it is rarely the case that the available data from a production asset not only has enough samples of failure, but also covers all possible failure types of the system (Fathi, van de Venn, & Honegger, 2021).

This in turn raises the need for the data generation via simulation models, which underlines the importance of currently missing related work on simulation-to-real transfer and domain adaptation (DA) techniques for RUL and/or HI estimation in PdM.

The *model-based approaches* leverage mathematical and physical models to estimate the RUL. This usually requires parameter tuning, *e.g.,* using Markov process model or Winner process, for converging to the behavior of the physical system (Hanachi, Liu, Banerjee, Chen, & Koul, 2014; Si, Wang, Hu, Zhou, & Pecht, 2012; Thelen et al., 2022). The parameter tuning is highly sensitive to the parameter initialization, which is normally based on the empirical knowledge from the physical system (Lei et al., 2016). Even when the system parameters are estimated correctly, any changes in the production setting and/or the production asset itself requires a re-calibration.

Both the data-driven and the model-based approaches suffer from inadequacy in practical solutions: the relevant data is either missing, or the models are not robust enough, respectively.

Hybrid PdM models combine the two approaches by learning from both the historical data and the data synthetised from the simulation models. The hybrid models have proven to be effective in terms of reliability and efficiency, and address some of the issues of pure data-driven or model-based solutions, such as reduction in data acquisition time and increased model robustness (Chang, Fang, & Zhang, 2017; D. Chen et al., 2022; Lin, Yu, Wang, Che, & Ni, 2022; Didona & Romano, 2014). In practice, the sampling from simulation models can not come up for the lack of historical data, so the conventional hybrid PdM models are also heavily dependant on annotated data. They are normally applied to systems for which extensive labelled datasets are available, *e.g.,* RUL prediction for lithium-ion batteries and aircraft engine (Fei, 2022; Saxena, 2023). As large datasets are not available for production assets in a manufacturing factory, the conventional hybrid models are inapplicable in the manufacturing setting.

In this paper, we focus on hybrid modelling for this specific scenarios where gathering extensive labelled data from the physical system is non-trivial (or even impossible), and where precise and robust simulation models are unavailable. To compensate for that, we propose to extract as much value as we can from both sources by applying the *domain adaptation* between the simulation and the real-world scenarios (see Fig. 1).

In the same vein, we further improve the robustness by considering the data distribution shifts which are a common consequence of diverse manufacturing requirements in an Industry 4.0 setting due to flexible and adaptable production (Fathi, Sadurski, Kleinert, & van de Venn, 2023). In order to cover as much of the parameter space of the system as possible, we alter accordingly the modelling of the initial condition and the degradation of the system.

The main contribution of this paper is thus four-fold. We:

1. Propose a hybrid PdM solution which relies mainly on the data from a simulation sub-module and few samples from the target domain for training its data-driven sub-module (supervised DA),
2. Propose a new concept of simulation data generation aiming for domain adaptation called Parameter and Data Perturbation (PDP), for covering as much of the parameter and degradation space of the physical system as possible,
3. Inspect the impact of the changes in the generated data from the simulation sub-module and the physical system on the performance of the data-driven sub-module,
4. Demonstrate how the additional simulation data used for model training results in a more spread-out feature importance across a wider range of sensor readings from the system while making predictions.

The rest of the paper is outlined as follows. First, some related work addressing synthetic data generation from simulation models for dealing with scarce labelled data and domain adaptation are presented. Afterwards, the details of the simulation model are provided. Thereafter, the results of model training using the simulation model data with PDP are presented. Lastly, discussion and the future work of this work are presented and some conclusions are drawn.

2. Related Work

2.1. Lack of annotated data from the target system

One method used for reducing the time spent gathering data from the target application using hybrid modelling is *boostrapping* (Didona & Romano, 2014). The main idea of *boostrapping* is to rely on a simulation model of the target use case and to generate initial synthetic training set for the data-driven model training. Thereafter, the data-driven model tries to incorporate knowledge from the target system as soon as a data point is available. In (Didona & Romano, 2014), the authors propose to remove the synthetic data points in the vicinity of samples from the target system to prevent obfuscating information from the real samples. However, for the purpose of RUL prediction in PdM, the annotated samples from the physical system are scarce and costly to gather. Hence, we propose to instead to keep these valuable samples and to combine them with data from different working conditions of the system generated from the simulation model

2.2. Adaptation to different working condition

Another important issue impacting the performance of data-driven PdM models is the varying working condition of the production assets in industry. These constant changes can make models trained with a specific working condition (a.k.a. source distribution) obsolete as changes occur in the system (Ragab, Chen, Wu, Kwoh, & Li, 2020). They cause a data distribution shift between the data employed to train the PdM model and the data acquired during the model's deployment in the production line. This discrepancy between the source and target distribution raises the need for techniques such as domain adaptation. In fact, domain adaptation aims to train a model on multiple source domains which are annotated so that the model can generalized to new and unseen target domains (Farahani, Voghoei, Rasheed, & Arabnia, 2021).

To the best of our knowledge, no other PdM work adopted domain adaptation methods for robust RUL and HI estimation in light of lacking annotated historical data from the physical system using simulation-to-real transfer techniques. Moreover, the current literature (Farahani et al., 2021; Yu, Fu, Ma, Lin, & Li, 2021; Rahat et al., 2022; Yang, Lei, Jia, & Xing, 2019; Gao, Liu, Huang, & Xiang, 2021; Wang, Taal, & Fink, 2021) reduce PdM to be a binary or a multi-class variable. We treat it as estimating the RUL or HI as a continuous value for better adaptability to different scenarios.

In addition, given the degradation process of different critical components of a production asset, the labelling function mapping the input space to the output space, can not only be different in the target domain, but also change in the target domain given the dominant degradation process of any arbitrary critical component (Cortes & Mohri, 2011; Nejjar, Geissmann, Zhao, Taal, & Fink, 2024)

These two, the estimation of RUL and/or HI as a continuous variable under the assumption of scarcity of annotated data from the target domain, and the possibility of data distribution shifts in the target domain are the primary motivation behind the proposed PDP method outlined in this work (Fig. 2).

2.3. Simulation-to-real transfer

Numerous application, especially safety-critical system, suffer from lack of labelled data as gathering such datasets is costly or endangers the human operator (Kaufmann et al., 2020; Tiboni, Arndt, & Kyrki, 2023). Therefore, simulation models are used to recreate different scenarios which are also labelled for model training. Nonetheless, modelling errors and the complexity of physical systems prevent the zero-shot deployment of data-driven models trained with such simulation models. One way for increasing the robustness of the trained models is randomize the dynamic parameters of the system which is a.k.a. *domain randomization* (Peng, Andrychowicz, Zaremba, & Abbeel, 2018). Doing so increases the robustness of the trained model at the the cost of its optimality (Tiboni et al., 2023). In this paper, we use the same method to cover as much as the parameter and degradation dynamics space as possible. Different starting conditions, model parameter and degradation processes are the main sources of domain randomization in this paper (Fig. 2).

3. Simulation model

Data-driven monitoring systems require continuous data collection that must extend over a period of time before they can provide effective results (Bonomi et al., 2021). Such data collections, referred to as run-to-failure (R2F) data, are normally expected to start from a healthy production asset state and end with the asset failing or malfunctioning. In the present scenario, there is the additional consideration that R2F tests are, by their nature, long-lasting while accelerated destructive testing is not always possible. Additionally, these tests are expensive with uncertain success. We propose to use a simulated model of the system to obtain realistic data (Ferrario et al., 2019) on its response to the system's most common types of wear and tear.

The model creation phase is critical because it must satisfy several conflicting requirements:

- The model must be complex enough to represent deteriorating operating conditions realistically.

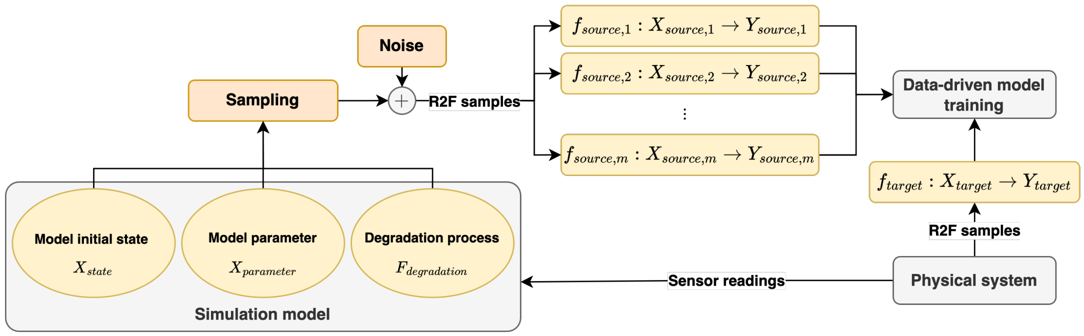

Figure 2. DA via PDP

- The model must be simple enough to be calibrated in a short time and from accessible data.
- The model must take into account the system whose condition is being monitored, the measurement instruments used, and how the data are processed. To some approximation, it must also take into account the context in which the system operates (*e.g.*, the effect of other components).

To meet all these needs, we developed a system with the following characteristics:

- Our models follow the multi-physical system (*Simscape*) as a concentrated variable model. This allows parameters to be easily configured.
- The model is wrapped in a Python script that allows generating the random parameters, handles post-processing, and eventually repeats or recovers the simulation scenarios in case of failure.

3.1. Overview of the simulation model

Our specific use case is the condition and life monitoring of a series of pneumatic cylinders. Based on its characteristics, we separate the model into 5 macro blocks (Fig. 3):

1. **Control system block:** modeled as timed output signals.
2. **Air feeding system and sensor block:** modeled while taking into account the fluid dynamic aspect and the characteristic times of the sensors (*i.e.*, thermometer, pressure switch and flow switch). The generated data are saved as time series on temporary files.
3. **Valve blocks:** modeled as adjustable restrictions controlled with a Boolean signal taking into account the fluid-dynamic aspect and implementation delays (Fig. 5).
4. **Pipe blocks:** modeled from the fluid dynamics and heat transfer point of view.
5. **Cylinder block:** the fluid-dynamic, mechanical, and thermal parts of the cylinders are modeled. The latter takes into account the velocity damping systems included in the final section of the cylinder and the speed controller valves outside the cylinder. This block also contains the modeling of possible failure types: air leakage is modeled as an adjustable restriction between the two chambers or between the chambers and the outside, and the state of the seal as a parameter that changes the friction force of the plunger. These parameters can be set with configurable constants prior to simulation. (Fig. 4).

After simulation, we read the simulated time series and compare them against the readings of the measuring instruments with the goal of obtaining the same results as the real system. During the reading, the acquisition frequency of the real system, the interaction with the cylinder's limit sensors, and any post-processing are taken into account.

In order to optimize the computing load and the amount of data transmitted, we later do not to use the time series directly. Instead, we represent the operation of the system with only a few particular values. For example, for each pneumatic cylinder, the actuation delay, the time of arrival, the airflow at departure, the airflow at arrival, the maximum airflow, the amount of air absorbed during the movement, the average pressure, and the minimum pressure are collected.

After the model is created, a calibration is performed using the available data, and the values obtained are compared with the actual values to check for a match.

To model wear damage, after an analysis of component failure modes (Nakutis & Kaškonas, 2008; Belforte, Raparelli, & Mazza, 1992; J. Chen, Zio, Li, Zeng, & Bu, 2018), effective parameters are identified to represent the state of the system. In the analyzed use case, the possible leakage is characterized as three adjustable local restrictions (between the first chamber and the environment, between the second chamber and the environment, or between the chambers) while the seal state is an adjustable friction coefficient.

Simulations are performed from an ideal operating state corresponding to the HI of 100% (the state of the part at the time of system calibration, assumed to be healthy) to a failure state, corresponding to the HI of 0%. The law by which the condition is calculated depends on the use that is made

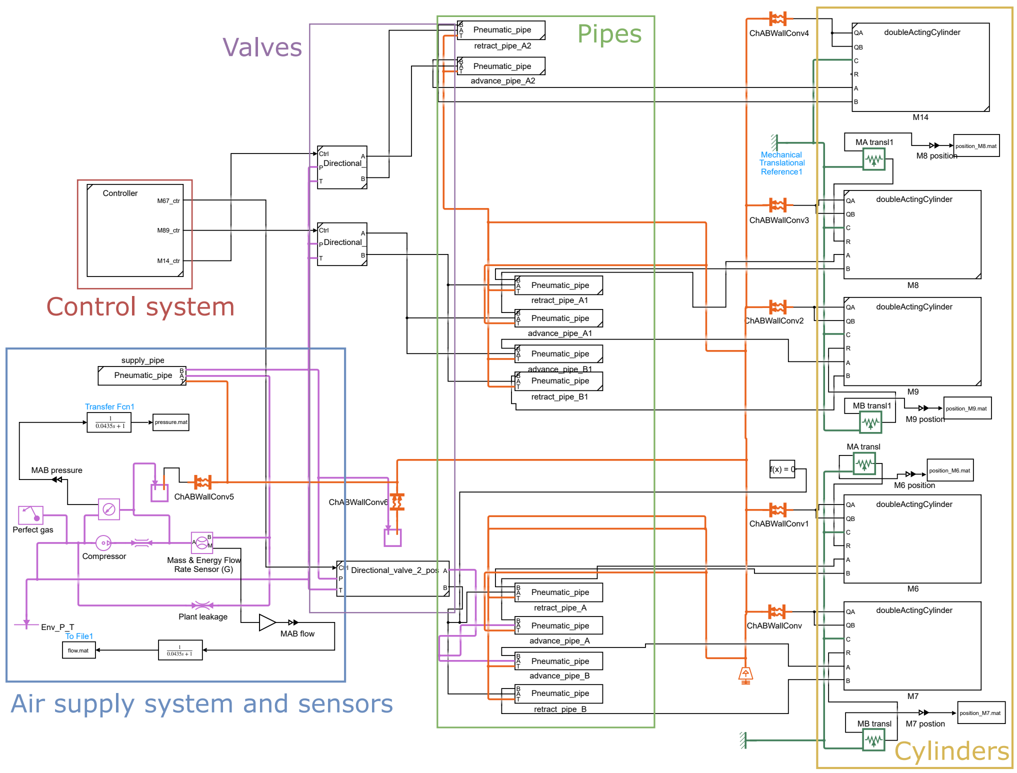

Figure 3. View of the physical model with macro blocks and the structure of major ones highlighted.

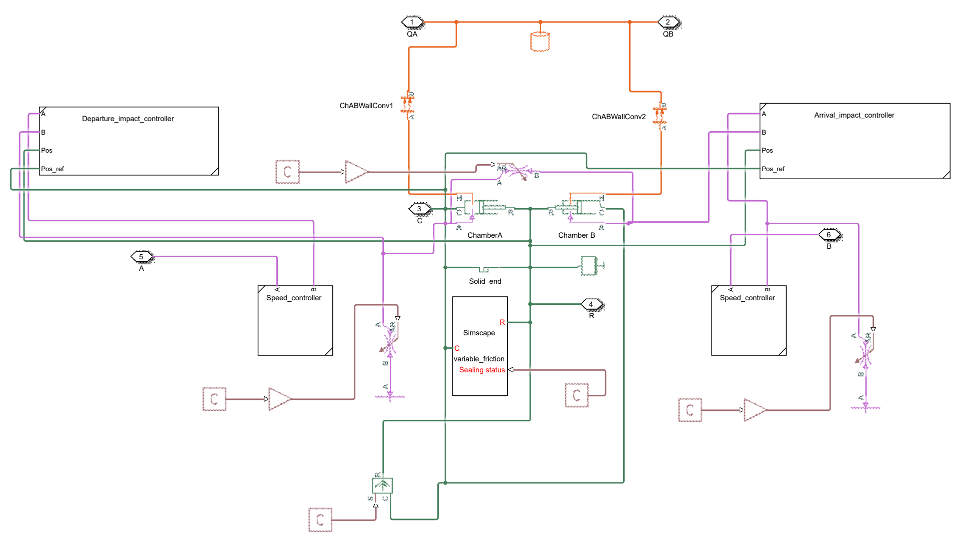

Figure 4. Detail view of one of the blocks modeling the behavior of pneumatic cylinders. It can be seen the variable restrictions that model leakage and the customized block that models seal friction.

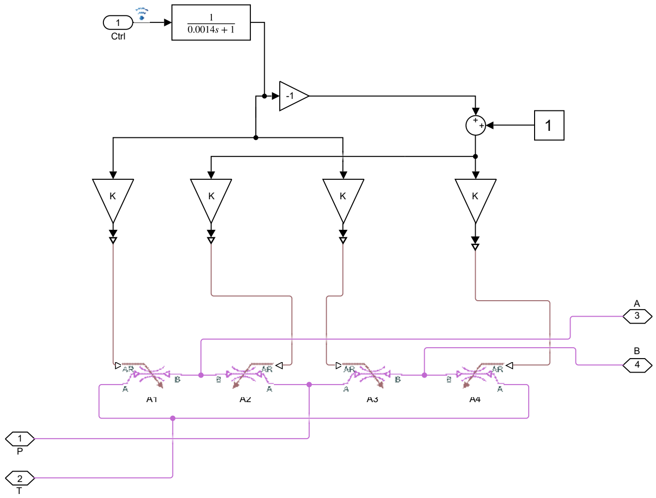

Figure 5. Detail view of one of the blocks modeling the behavior of pneumatic valves.

of the monitored machine and the requirements that it must have. For example, in the case of pneumatic cylinders, it is calculated based on the change in movement time and energy used.

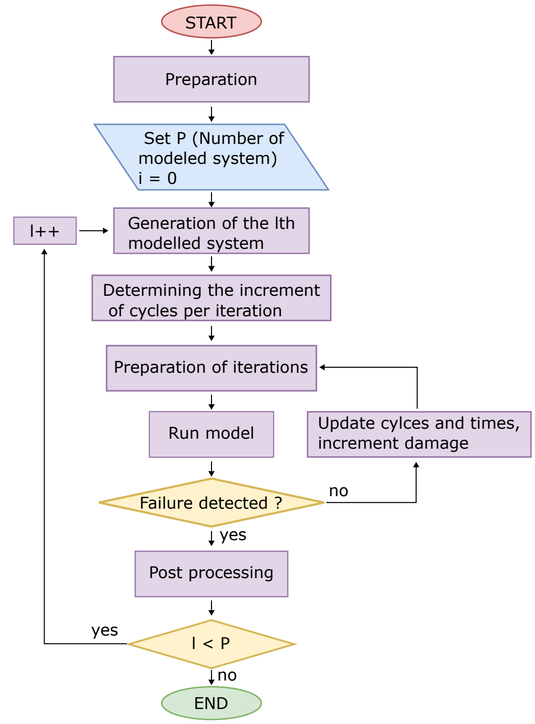

Figure 6. The flowchart of the script running the simulations

3.2. Parameter perturbation in simulation model

To make the model effective under different operating conditions, we run several simulations by varying the system boundary conditions and the damage progression law. In the logic of keeping the algorithm simple and applicable to different types of models, each failure mode was treated independently (thus each parameter modeling wear progresses at different rates but does not affect others). Of course, the effects that these parameters have on the operation of the simulated device sum up and affect the HI.

This process can be schematized in the following points (see Fig. 6):

1. **Preparation:** The model is calibrated from the conditions measured in the real system. Then it is determined what is the critical value of each of the parameters that represent the damage (the value that alone would bring the part HI to 0) through a series of simulations in which the main operating parameters are varied (see Fig. 7) and from which matrices of critical values are obtained.

2. **Generation of the modeled system:** At this stage, the system's own characteristics, those that are not expected to change over time are determined. These parameters are the damage progression laws for each component and each failure mode, and also additional static parameters such as the position of the control valves. Every damage progression is determined by a quadratic law defined as follows:

$$\begin{cases} (n < n_f): & Kd_{i,j}(n) = n^2 \cdot \frac{(2-4 \cdot nli)}{n_f^2} + n \cdot \frac{4 \cdot nli - 1}{n_f} \\ (n \geq n_f): & Kd_{i,j}(n) = \frac{2n_f(2nli-1) - x(4nli-3)}{n_f} \end{cases} \tag{1}$$

model where $Kd_{i,j}$ is the damage progression coefficient for the failure mode i of the device j, n is the number of cycles made by the device, $nli \in [0.25, 0.7]$ is the *non-linearity index*, n_f is the number of cycle that, would bring the part to failure. The law is formulated to be always increasing while being parabolic up to the n_f value, and then proceeds linearly. The damage progression coefficient multiplied by the critical damage values determines the value of the corresponding failure parameter.

3. **Determining the increment of cycles per iteration:** For each model defined in the previous step, a series of simulations must be run in which the number of cycles performed by the machine increases progressively so as to increase the various damage parameters. The simulations must proceed until the HI of at least one of the parts drops to 0. To keep the computation time acceptable, it was decided that only one cycle is actually modeled in each simulation while the cycle counter and time are recalculated using these formulas:

$$n_k = n_{k-1} + min\left(n_{f,i,j}\right)/N \tag{2}$$

$$t_k = t_{k-1} + min\left(n_{f,i,j}\right) \cdot t_n \tag{3}$$

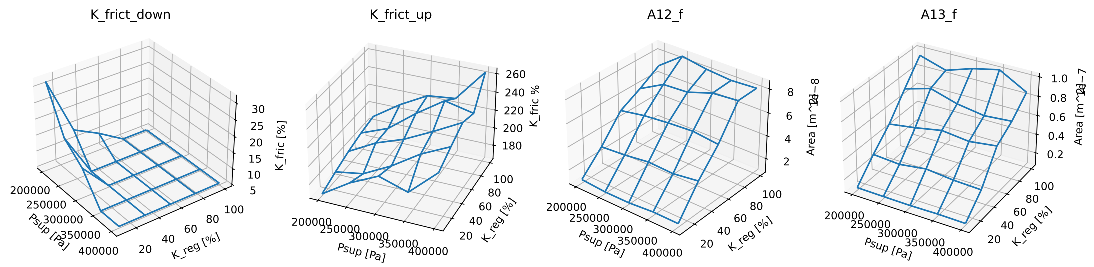

Figure 7. The figure shows the critical matrices for a pneumatic cylinder. The matrices represent the critical value of various damage parameters (friction reduction, friction increase, leakage between chambers, leakage between the second chamber and the environment) as a function of supply pressure and speed control valve adjustment. It can be seen that the critical friction is greatly influenced by pressure while the critical value of leakage is more influenced by valve adjustment.

Where, n_k in the number of cycles made by the device in the iteration k, $n_{f,i,j}$ is the cycles to fail for the failure mode i of the device j, N is the desired number of iterations, t_k is the virtual time in which the simulation k takes place, and t_n is the number of cycles performed in the unit of time.

This reduces the computational time by simulating only a number of cycles equal to about N, which are equally spaced. In practice, the number of simulations may vary in relation to N, as the overlapping effects of increasing different damage parameters may dampen or accentuate their impact on the part condition.

4. **Preparation of iteration:** Before each simulation, some parameters that may vary during the lifetime of the part, such as environmental parameters, are randomly calculated. Specifically, in the presented use case, the pressure of the air supply system, the ambient pressure, the ambient temperature, the sampling start time, the pressure drop of the supply system, and the friction coefficient of the sealing of each cylinder are varied.
5. **Run to failure:** For each iteration first the damage parameters are calculated using the equation:

$$P_{i,j,k} = Kd_{i,j,k}\left(n_k\right) \cdot P_{crit,i,j} \tag{4}$$

where $P_{i,j,k}$ is the damage parameter for the failure mode i of the device j and the iteration k, $Kd_{i,j,k}\left(n_k\right)$ is the damage coefficient for the failure mode i of the device j and the iteration k (see Eq. 1 and 2), $P_{crit,i,j}$ is the critical parameter value for the failure mode i of the device j computed from the critical matrix crested in the preparation phase.

After the computation of the damage parameters, the physical model is run, and finally, the HI of each part is calculated. In case one of the parts has a HI equal to or less than zero the iterative process is terminated otherwise the next one is run.
6. **Post-processing:** At the end of the iterations, the collected data is saved and labeled with the RUL and the HI (depending on the use case) related to the corresponding iteration.

4. Data analysis and prediction model performance

In the conducted studies we assume that labelled data from the target domain (physical system) is available, which categorizes the proposed method as a supervised DA solution (Motiian, Piccirilli, Adjeroh, & Doretto, 2017). In fact, as soon as the generated R2F data from the simulation and the labelled asset sensor readings are available, the acquired data can be used to train a prediction model. In addition, gradient boosted trees (T. Chen & Guestrin, 2016) have been used to estimate the health status of the physical system, defined as HI or RUL, given the sensor readings from it. For evaluating the performance boost from PDP, two separate regression models (XGBR) with the same complexity will be trained using the following datasets:

- Limited annotated data (10% of the available data) from the physical system (the model trained with this dataset is referred to as **XGBR1**)
- Limited annotated data (10% of the available data) from the physical system and the R2F from the simulation model by employing PDP (the model trained with this dataset is referred to as **XGBR2**)

4.1. Use cases and experiments

In this industrial project, two different systems, from SMC Schweiz AG[1] and TCI engineering[2], are tested as use cases to inspect the scalability of the proposed method. The former, is a pneumatic pick and place demonstrator which can be used to mimic different failure types given various working conditions (*e.g.*, by changes in the main pressure of the compressed air). For this use case, the deployed model is used to predict the RUL of the system as degradation in the physical system does not cause any significant financial loss. In fact, this demonstrator is used to create real R2F data for testing the accuracy of the RUL predictions.

The latter; however, is used in a production line owned by

[1]https://www.smc.eu/de-ch
[2]https://www.tci-sa.ch/en/

a third-party company and thus no failures can be artificially built during production. For this use case only the HI of the system is predicted as no failure samples are available from the physical system. As shown later, given the changes in the working conditions, the customer's needs and minor inspections, there are numerous fluctuations in the HI values, resulting them not to be monotonic.

In what follows, the results of model performance comparison for predicting the RUL and HI is provided in detail. However, the similar results of feature importance distribution for RUL and HI predictions, subsection 4.3, and the robustness to the noise, subsection 4.5, are not included in order to conserve space. Nonetheless, this omission does not diminish the completeness of this paper in any manner.

4.2. Model accuracy for predicting the HI

The XGBR1 and XGBR2 prediction models are tested on the data attained from the physical system which were not previously exposed to them during model training. These model have the R^2 scores of 0.676 and 0.853 respectively, which indicates the superiority of the model trained with simulation data. Moreover, for ensuring that sample selection does not impact the accuracy comparison between XGBR1 and XGBR2, these accuracy values are calculate as the mean of accuracy given different seeds for sampling data from the physical system.

In addition, Fig. 8 demonstrates the progression of the HI of the physical system during approximately 3 months. Given the fact that, the data acquisition from the physical system did not start right after a maintenance, there is no reference asset behavior which represent a fully healthy behavior. Therefore, the calculated HI values are equally biased to start from a value which is as close to 1 as possible. In addition, as stated in subsection 4.1, numerous internal and external factors constantly impact the physical system during production, which prevents a monotonic HI sequence.

4.3. Feature importance distribution

The trained XGBRs can provide insight about the importance of different sensor readings from the system. In fact, importance values indicate how influential one feature is in determining the output of the prediction model. Considering that, data reading from industrial assets is not always perfect, it can be expected that there are scenarios where the attained data from an asset contains noise, has missing values or in an extreme case the installed sensor does not provide any data. In such situations, it is vital to determine the role of different sensor readings and also try to train models which use a wider range of sensor readings from the physical system. By doing so, erroneous predictions from the model are prevented, resulting in enhanced model reliability.

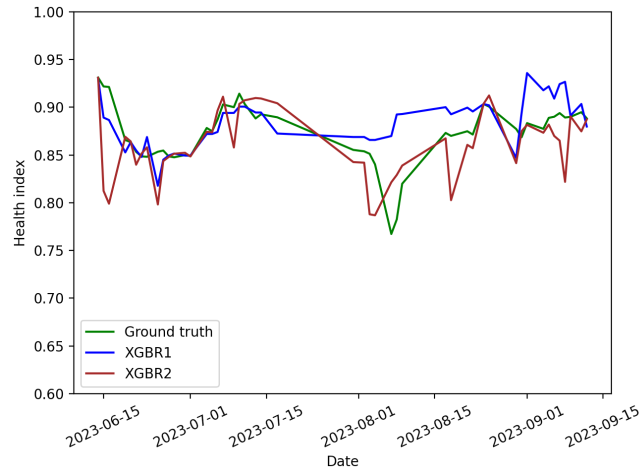

Figure 8. Asset HI along with XGBR1 and XGBR2 predictions

On a separate note, by comparing the significance of different asset readings, it is also possible to verify if the trained prediction model has converged to a PdM solution or is merely a preventive maintenance model relying on the cycle number.

As it can be seen in figures 9 and 10, given the sparsity of data from the physical model, the XGBR trained only using the data from the physical system has a highly unbalanced feature importance across different readings of the studied asset. Therefore, from a model reliability point of view, the simulation model can significantly enhance the performance of the model. Please note that for sake of anonymity and data protection for the involved industrial partner in the conducted studies, hashed sensor reading names are provided in the aforementioned figures.

4.4. Model comparison for predicting the RUL

The XGBR1 and XGBR2 trained for the SMC Schweiz AG demonstrator have fairly similar R^2 scores of 0.980 and 0.917 respectively. The higher accuracy in predicting the RUL is partially attributed to fact that the controlled laboratory setup allows us to control all the boundary conditions. Whereas in the shopfloor, the system is influenced by numerous factors that cannot be controlled or predicted, *e.g.*, a drop in pressure or regulation intervention. Additionally, it is also due to the fact that the induced failure in the system is a result of (semi-)linear opening of the valves in the demonstrator for mimicking leakage in various parts of the system. Nonetheless, as discussed in subsection 4.3, the XGBR1 model puts more emphasis on the information about the cycle count and ignores the information from the rest of the available sensor readings making it a less desirable solution given PdM requirements, especially in case that a faster failure than the ones seen before during model training occurs in the system.

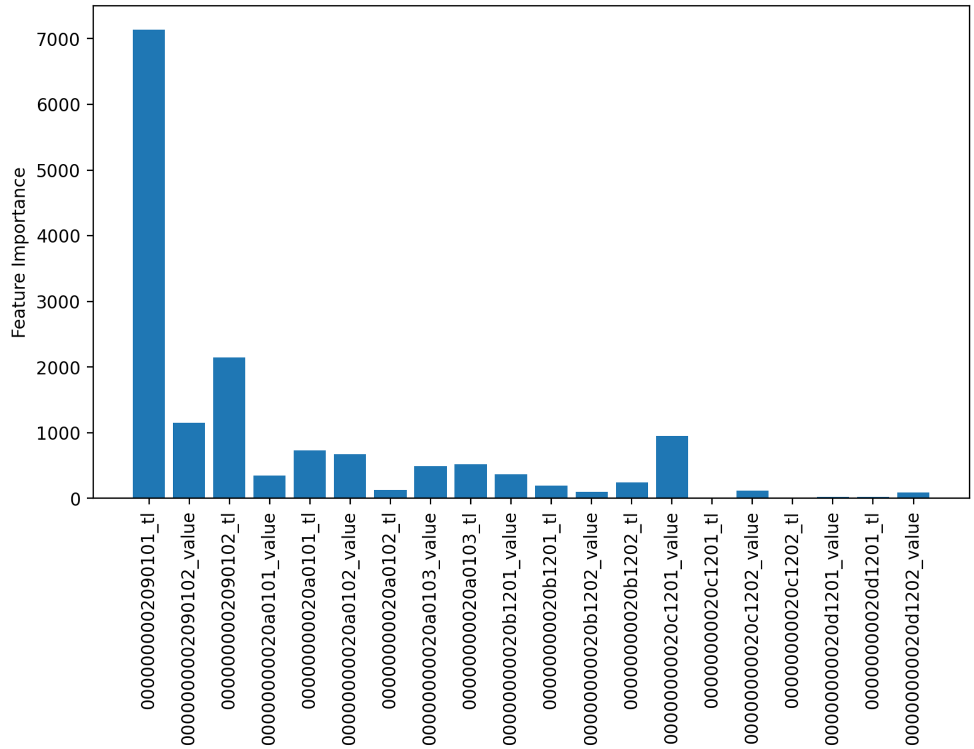

Figure 9. Feature importance among different asset sensor readings for XGBR1 model. This model relies on a limited number of sensor readings from the system

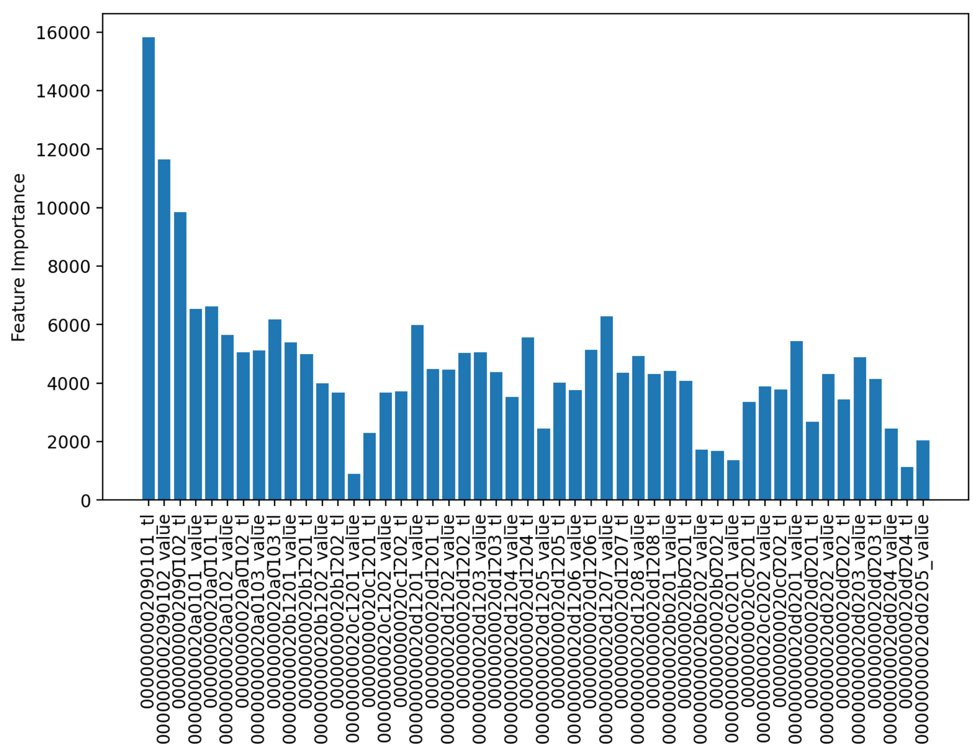

Figure 10. Feature importance among different asset sensor readings for XGBR2 model. This model has a more spread-out feature importance across different sensor readings of the system

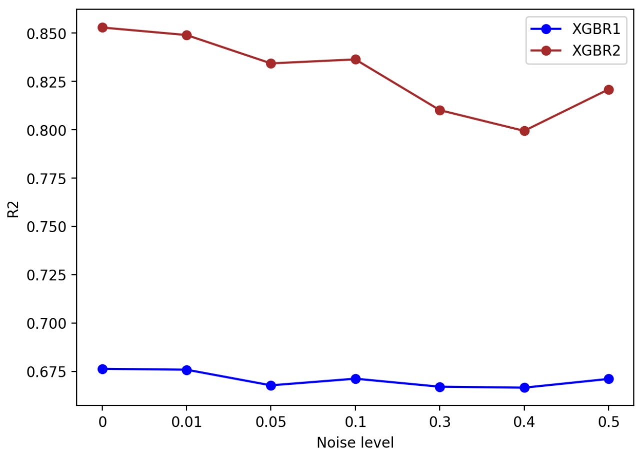

Figure 11. Impact of asset reading noise on the accuracy of the trained prediction models

4.5. Robustness to asset sensor reading noise

In this part of the conducted studies, the impact of noise on the most decisive asset readings on the accuracy of the XGBR1 and XGBR2 for the TCI engineering case are examined. For manipulating the asset sensor readings, 8 of the most influential features given the values in figures 9 and 10 are selected. Thereafter, samples of each of these features ($x^i \in X$) are distorted as follows:

$$x^i_{new} = x^i + Max\{x^j\}_{j=1}^{|X|} \times noise\,level \times d \sim N(0,1) \quad (5)$$

where $|X|$ is the total number of feature readings, $noise\ level$ is scalar value (see Fig 11) and d is a sampled value from $N(0,1)$ which represents normal distribution with mean of 0 and standard deviation of 1. Fig 11 shows the impact of noise on the R^2 score of the predictors. As it can be seen, regardless of the added noise value, the performance of the model trained with the additional simulation data is superior which suggest the robustness of the trained model compared to the prediction model trained only with scarce data from the physical system.

5. Discussion and conclusion

Gathering annotated data from a physical system for developing PdM solutions is one of the most time-consuming and expensive steps which inhibits many end users in industry for utilizing the full potential of their production assets. In the conducted studies, we aimed to introduce a novel approach for RUL and HI prediction model training which uses data generated from a simulation model and a minimal set of samples from the physical system as apposed to complete R2F datasets from the asset. We aimed to highlight the importance of simulation data generation with PDP for covering as much as of the parameter space of an asset for enhancing the performance of the prediction model despite the scarcity of asset readings. It was shown how the proposed method, in the best case scenario, increases the R^2 score of the trained model by 26% while simultaneously using a wider range of sensor readings from the physical system. Furthermore, the results of model performance deterioration in presence of asset reading noise demonstrated that, regardless of the added noise to the readings, the R^2 score of the model trained with the additional simulation model data is higher. For the future work, we aim to develop classifiers for different working conditions of an asset and then find the corresponding regions of the data generated from the simulation model for a more fine-tuned data generation. In addition, we aim to inspect the performance of the prediction model in different regions of the parameter space and generate data from the simulation model which explicitly can boost the performance of the model in the chosen region of the parameter space.

References

Belforte, G., Raparelli, T., & Mazza, L. (1992). Analysis of typical component failure situations for pneumatic cylinders under load. *Lubrication engineering*, *48*(11), 840–845.

Bonomi, N., Cardoso, F., Confalonieri, M., Daniele, F., Ferrario, A., Foletti, M., ... Pedrazzoli, P. (2021). Smart quality control powered by machine learning algorithms. *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, 764–770.

Chang, Y., Fang, H., & Zhang, Y. (2017). A new hybrid method for the prediction of the remaining useful life of a lithium-ion battery. *Applied energy*, *206*, 1564–1578.

Chen, D., Meng, J., Huang, H., Wu, J., Liu, P., Lu, J., & Liu, T. (2022). An empirical-data hybrid driven approach for remaining useful life prediction of lithium-ion batteries considering capacity diving. *Energy*, *245*, 123222.

Chen, J., Zio, E., Li, J., Zeng, Z., & Bu, C. (2018). Accelerated Life Test for Reliability Evaluation of Pneumatic Cylinders. *IEEE Access*, *6*, 75062–75075. (Conference Name: IEEE Access)

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Cortes, C., & Mohri, M. (2011). Domain adaptation in regression. In *International conference on algorithmic learning theory* (pp. 308–323).

Cui, L., Du, S., & Hawkes, A. G. (2012). A study on a single-unit repairable system with state aggregations. *IIE Transactions*, *44*(11), 1022–1032.

Didona, D., & Romano, P. (2014). On bootstrapping machine learning performance predictors via analytical models.

arXiv preprint arXiv:1410.5102.

Farahani, A., Voghoei, S., Rasheed, K., & Arabnia, H. R. (2021). A brief review of domain adaptation. *Advances in data science and information engineering: proceedings from ICDATA 2020 and IKE 2020*, 877–894.

Fathi, K., Sadurski, M., Kleinert, T., & van de Venn, H. W. (2023). Source component shift detection classification for improved remaining useful life estimation in alarm-based predictive maintenance. In *2023 23rd international conference on control, automation and systems (iccas)* (p. 975-980). doi: 10.23919/ICCAS59377.2023.10316874

Fathi, K., van de Venn, H. W., & Honegger, M. (2021). Predictive maintenance: an autoencoder anomaly-based approach for a 3 dof delta robot. *Sensors*, *21*(21), 6979.

Fei, C. (2022). *Lithium-ion battery data set.* IEEE Dataport. Retrieved from `https://dx.doi.org/10.21227/fh1g-8k11` doi: 10.21227/fh1g-8k11

Ferrario, A., Confalonieri, M., Barni, A., Izzo, G., Landolfi, G., & Pedrazzoli, P. (2019). A Multipurpose Small-Scale Smart Factory For Educational And Research Activities. *Procedia Manufacturing*, *38*, 663–670.

Gao, Y., Liu, X., Huang, H., & Xiang, J. (2021). A hybrid of fem simulations and generative adversarial networks to classify faults in rotor-bearing systems. *ISA transactions*, *108*, 356–366.

Hanachi, H., Liu, J., Banerjee, A., Chen, Y., & Koul, A. (2014). A physics-based modeling approach for performance monitoring in gas turbine engines. *IEEE Transactions on Reliability*, *64*(1), 197–205.

Kaufmann, E., Loquercio, A., Ranftl, R., Müller, M., Koltun, V., & Scaramuzza, D. (2020). Deep drone acrobatics. *arXiv preprint arXiv:2006.05768*.

Lei, Y., Li, N., Gontarz, S., Lin, J., Radkowski, S., & Dybala, J. (2016). A model-based method for remaining useful life prediction of machinery. *IEEE Transactions on reliability*, *65*(3), 1314–1326.

Lin, R., Yu, Y., Wang, H., Che, C., & Ni, X. (2022). Remaining useful life prediction in prognostics using multi-scale sequence and long short-term memory network. *Journal of computational science*, *57*, 101508.

Liu, J., Wang, W., Ma, F., Yang, Y., & Yang, C. (2012). A data-model-fusion prognostic framework for dynamic system state forecasting. *Engineering Applications of Artificial Intelligence*, *25*(4), 814–823.

Motiian, S., Piccirilli, M., Adjeroh, D. A., & Doretto, G. (2017). Unified deep supervised domain adaptation and generalization. In *Proceedings of the ieee international conference on computer vision* (pp. 5715–5725).

Nakutis, Žilvinas., & Kaškonas, P. (2008). An approach to pneumatic cylinder on-line conditions monitoring. *Mechanics*, *72*(4), 41–47.

Nejjar, I., Geissmann, F., Zhao, M., Taal, C., & Fink, O. (2024). Domain adaptation via alignment of operation profile for remaining useful lifetime prediction. *Reliability Engineering & System Safety*, *242*, 109718.

Peng, X. B., Andrychowicz, M., Zaremba, W., & Abbeel, P. (2018). Sim-to-real transfer of robotic control with dynamics randomization. In *2018 ieee international conference on robotics and automation (icra)* (pp. 3803–3810).

Ragab, M., Chen, Z., Wu, M., Kwoh, C. K., & Li, X. (2020). Adversarial transfer learning for machine remaining useful life prediction. In *2020 ieee international conference on prognostics and health management (icphm)* (pp. 1–7).

Rahat, M., Mashhadi, P. S., Nowaczyk, S., Rognvaldsson, T., Taheri, A., & Abbasi, A. (2022). Domain adaptation in predicting turbocharger failures using vehicle's sensor measurements. In *Phm society european conference* (Vol. 7, pp. 432–439).

Saxena, A. (2023). *Nasa turbofan jet engine data set.* IEEE Dataport. Retrieved from `https://dx.doi.org/10.21227/pjh5-p424` doi: 10.21227/pjh5-p424

Si, X.-S., Wang, W., Hu, C.-H., Zhou, D.-H., & Pecht, M. G. (2012). Remaining useful life estimation based on a nonlinear diffusion degradation process. *IEEE Transactions on reliability*, *61*(1), 50–67.

Thelen, A., Li, M., Hu, C., Bekyarova, E., Kalinin, S., & Sanghadasa, M. (2022). Augmented model-based framework for battery remaining useful life prediction. *Applied Energy*, *324*, 119624.

Tiboni, G., Arndt, K., & Kyrki, V. (2023). Dropo: Sim-to-real transfer with offline domain randomization. *Robotics and Autonomous Systems*, *166*, 104432.

Wang, Q., Taal, C., & Fink, O. (2021). Integrating expert knowledge with domain adaptation for unsupervised fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1–12.

Yang, B., Lei, Y., Jia, F., & Xing, S. (2019). An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings. *Mechanical Systems and Signal Processing*, *122*, 692–706.

Yu, K., Fu, Q., Ma, H., Lin, T. R., & Li, X. (2021). Simulation data driven weakly supervised adversarial domain adaptation approach for intelligent cross-machine fault diagnosis. *Structural Health Monitoring*, *20*(4), 2182–2198.