

# Simulation-driven domain adaptation for few-shot gearbox fault assessment

Luke D. van Eyk\*, Stephan Schmidt, P. Stephan Heyns

Centre for Asset Integrity Management, Department of Mechanical and Aeronautical Engineering, University of Pretoria, Pretoria, South Africa

Received 23 June 2026; Revised 12 August 2026; Accepted 4 September 2026; Published online 23 September 2026

\*Corresponding author: [luke.vaneyk@up.ac.za](mailto:luke.vaneyk@up.ac.za)

**Abstract:** This paper evaluates a simulation-driven domain adaptation (SDDA) workflow for few-shot gearbox fault assessment. The study addresses a key limitation of data-driven gearbox fault assessment methods, namely their reliance on large quantities of labelled fault data, which are rarely available in industrial applications. This work investigates whether physics-based simulated source data, a CNN-based data-driven classifier, and MK-MMD-based feature alignment can be effectively combined to assess gearbox faults under severely label-scarce target-domain conditions. The primary contribution of this work is the first systematic verification of a simulation-driven domain adaptation workflow for gearbox fault assessment under industrially relevant few-shot conditions and controlled simulation-to-real discrepancy factors. A comprehensive numerical investigation is conducted to evaluate the effect of the domain discrepancy factors, including model fidelity, sensor location, sensor noise, and physics-based modelling parameter mismatch. The results show that SDDA outperforms the source-only, target-only, and weight fine-tuning baselines across all investigated discrepancy conditions, with the relative performance gain degrading only under extreme sensor noise. The workflow is further evaluated through two experimental case studies using the Prognostics and Health Management datasets, PHM23 and PHM09. These case studies represent relatively well-aligned and substantially mismatched simulation-to-real transfer scenarios, respectively. In both cases, SDDA achieved higher fault detection performance than the considered baselines even with only a few labelled target-domain fault samples. These findings indicate that simulation-based approaches, when coupled with explicit domain alignment, can support data-efficient gearbox fault assessment in industrial environments where real fault data are scarce.

**Keywords:** Transfer learning, Domain adaptation, Gearbox fault assessment, Simulation-to-real, Few-shot, Hybrid methods, Data-driven methods, Physics-driven methods

## 1 Introduction

Data-driven models have received considerable attention in the field of gearbox fault assessment<sup>1</sup>. These models have shown strong performance, but their success is often contingent upon the availability of large volumes of historical fault data. Unfortunately, such data is typically scarce in real-world industrial settings, significantly limiting model efficacy [1]. Transfer learning has emerged as a promising solution to mitigate

this issue by transferring knowledge from a label-rich source domain to a label-scarce target domain.

Transfer learning approaches for rotating machinery fault assessment can broadly be categorised into two types: transfer in the identical machine (TIM) and transfer across different machines (TDM). In both settings, the source domain is assumed to be densely labelled while the target domain has limited or no labelled data [1].

In the case of TIM, the source and target domain data originate from the same machine, but represent different

<sup>1</sup> Fault assessment in this paper's context refers to tasks such as fault detection (binary classification) and fault diagnosis (multi-class classification).

operating conditions, typically involving variations in speed and/or load. Han et al. [2,3], Zhang et al. [4], and more recently, Qiu et al. [5] successfully illustrated such transfer tasks for the PHM09 gearbox dataset. Li et al. [6] demonstrated successful transfer for both the CWRU and PHM09 datasets, Lin et al. [7] for the Tsinghua and QPZZ gearbox datasets and Ha & Fink [8], She et al. [9,10], Chen et al. [11], and Yang et al. [12] for self-obtained gearbox datasets. Although notable and relevant, these studies do not explore the more complex TDM setting where source and target machines are structurally distinct.

TDM methods address transfer between different but related machines. Notable works include Li et al. [6], where the source domain was set as a bearing or gearbox dataset, and the knowledge was transferred to the opposite dataset in the target domain. Li et al. [13] used the CWRU dataset as the source domain, transferring knowledge to a self-obtained bearing dataset for the target domain. Haidong et al. [14] showcased knowledge transfer from a parallel shaft gearbox test rig to a bevel gearbox test rig. Li et al. [15] transferred knowledge from the IMS bearing dataset to the CWRU bearing dataset and vice versa. More recently, in a comprehensive study, Hu et al. [16] conducted various successful transfer tasks between the Paderborn (bearing), IMS (bearing), MFPT (bearing), Macau (gearbox) and NREL (gearbox) datasets.

While these efforts rely on experimental data as the source domain, this is not always practical. Experimental source domain data may be unavailable, costly to collect, or insufficiently representative of the target application. To address these challenges, researchers have turned to simulation-driven approaches. These involve the simulation of fault data from physics-based models designed to approximate the target system. Such methods bypass many challenges associated with experimental data collection.

Recent examples from Sobie et al. [17], Liu et al. [18], and Liu et al. [19] demonstrated the efficacy of simulated source domains through improved bearing fault diagnosis performance on the IMS, CWRU and Paderborn University datasets when compared to state-of-the-art techniques that did not use simulated source domain data. Simulation-driven transfer learning approaches have also successfully been applied to gearboxes, albeit to a limited extent. Li et al. [20] developed a dynamic planetary gearbox model, which was used to generate source data. The source data contained three fault classes, which were successfully transferred to two unseen planetary gearbox datasets, with good accuracy. In four separate but related setups, Xiong et al. [21], Song et al. [22], Song et al. [23], and

Zhu et al. [24] showcased the successful transfer of knowledge between a simulated source domain and a real target domain. In all three cases, MSC ADAMS or Matlab was used to simulate a planetary gearbox based on a SpectraQuest DDS planetary gearbox setup. Jiang et al. [25] and Chen et al. [11] also contributed to this growing body of work, showcasing successful simulation-to-real transfer using various test benches and modelling approaches.

Despite these advances, a critical assumption persists across all aforementioned simulation-driven works: the availability of a large number of unlabelled fault samples from the target domain during training. This assumption often breaks down in industrial practice, where gearboxes are rarely operated to failure, and labelled or unlabelled fault samples are scarce. Few-shot learning approaches, which aim to rapidly generalise to new tasks with limited labelled information [26], are therefore required for simulation-based transfer.

To our knowledge, Matania et al. [27] present the only related work that considers few-shot target domain samples within a simulation-based setup. However, their study focuses on fault severity regression rather than fault diagnosis, and it relies on handcrafted signal features rather than end-to-end learning. This leaves a clear gap in the current literature for a comprehensive analysis of simulation-driven transfer learning methods under limited labelled fault data conditions.

This paper evaluates a simulation-driven domain adaptation (SDDA) framework for few-shot gearbox fault assessment, outlined in [section 2](#). The SDDA method generates source data from a physics-based gearbox simulation and transfers this knowledge to a real target domain using only a few labelled target samples. The algorithmic components used in this framework, namely a CNN-based feature extractor and classifier together with MK-MMD-based domain alignment, are established techniques. This work does not attempt to introduce a new neural network architecture or adaptation loss for these established techniques. The novel contribution rather lies in it being the first systematic investigation and verification of this simulation-driven transfer strategy for few-shot gearbox fault assessment under realistic simulation-to-real discrepancy conditions. The SDDA framework should therefore not be interpreted as a simple stacking of physics-based simulation, CNN classification, and MK-MMD alignment. Rather, these established components are integrated into an end-to-end evaluation framework to determine whether physics-based source data can support real target-domain fault assessment when labelled target fault samples are scarce. Unlike existing few-shot simulation-driven approaches, the evaluated

workflow requires no manual feature engineering and is assessed under industrially relevant few-shot conditions.

To rigorously evaluate the SDDA approach in this setting, we conduct a comprehensive numerical investigation (section 3) that systematically examines how domain discrepancy factors such as model fidelity, sensor location, sensor noise, and machine mismatch affect transfer performance. These experiments generate data in a controlled environment to enable critical assessment of key transferability challenges in few-shot simulation-driven fault assessment. We then evaluate the method in two real-world case studies. Case study 1 (section 4) focuses on fault detection using the PHM23 dataset, where the simulated model is well-aligned with the real system. Case study 2 (section 5) evaluates the fault detection ability of the SDDA approach using the PHM09 dataset, representing a scenario with substantial domain mismatch. Based on this work's findings, conclusions and recommendations are presented in section 6.

In summary, this work makes three contributions to few-shot gearbox fault assessment:

1. The formulation and evaluation of an end-to-end simulation-driven domain adaptation workflow for gearbox fault assessment under severely limited labelled target-domain fault data.
2. A comprehensive numerical investigation that quantifies the effect of key simulation-to-target discrepancy factors, including model fidelity, sensor noise, sensor location, and machine mismatch, on few-shot simulation-driven fault assessment.
3. A comparative evaluation of simulation-driven domain adaptation against source-only, target-only, weight fine-tuning, and anomaly-detection baselines on two experimental gearbox datasets representing different levels of simulation-to-real mismatch.

Although previous works have explored simulation-driven transfer learning for gearbox fault assessment, to our knowledge, this is the first work to conduct a thorough investigation and verification of these techniques under few-shot conditions. Together, these contributions advance the field of gearbox fault assessment by addressing the underexplored intersection of simulation-driven domain adaptation and few-shot learning, with significant implications for industrial deployment.

## 2 Few-shot methodology: Simulation-driven domain adaptation

The simulation-driven domain adaptation (SDDA) methodology evaluated in this work provides a structured approach for few-shot gearbox fault assessment using simulated source data and limited labelled target data. The method comprises three core components: a physics-based gearbox model, a data-driven model (DDM), and a calibration mechanism. The physics-based gearbox model simulates gearbox vibration signals used for source domain data. A convolutional neural network (CNN) is selected as the DDM because it is capable of extracting meaningful features for fault assessment from raw time- or frequency-domain vibration signals. Finally, domain adaptation is utilised as the calibration mechanism, which enables the few-shot learning task. The individual algorithmic components are established in the transfer learning and fault diagnosis literature; their role here is to provide a reproducible baseline for systematically evaluating simulation-driven few-shot transfer in gearbox fault assessment. The SDDA methodology is shown in Fig. 1 and will be discussed in greater detail in the following sub-sections.

### 2.1 Physics-based gearbox model

For this work, a physics-based, single-stage, spur gearbox is modelled. This specific gearbox is chosen because it is common in gearbox modelling literature [28]. The approach for modelling gearbox vibrations is broken down into three distinct, but interconnected steps, namely:

- Dynamic modelling
- Stiffness modelling
- Fault modelling

These three modelling steps are presented in the three sub-sections to follow. The core focus of this work is on the calibration mechanism within the SDDA method, and not the physics-based model. A brief overview of each of the three modelling aspects is presented in this work, with the requisite parameters included in Appendix A.2. For a more complete, refined explanation of the physics-based modelling process, refer to [29], where the full modelling framework is described.

#### 2.1.1 Dynamic modelling of the physics-based gearbox model

The dynamics of a gearbox are modelled by solving the equations of motion of an  $N$  degrees of freedom (DOF) lumped parameter model, which is defined by:

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{f}(t), \quad (1)$$

where  $\mathbf{M} \in \mathbb{R}^{N \times N}$ ,  $\mathbf{C} \in \mathbb{R}^{N \times N}$  and  $\mathbf{K} \in \mathbb{R}^{N \times N}$  respectively are the mass, damping and stiffness matrices and  $\ddot{\mathbf{u}} \in \mathbb{R}^N$ ,  $\dot{\mathbf{u}} \in \mathbb{R}^N$ ,  $\mathbf{u} \in \mathbb{R}^N$  and  $\mathbf{f} \in \mathbb{R}^N$  are the acceleration, velocity, displacement and force vectors.

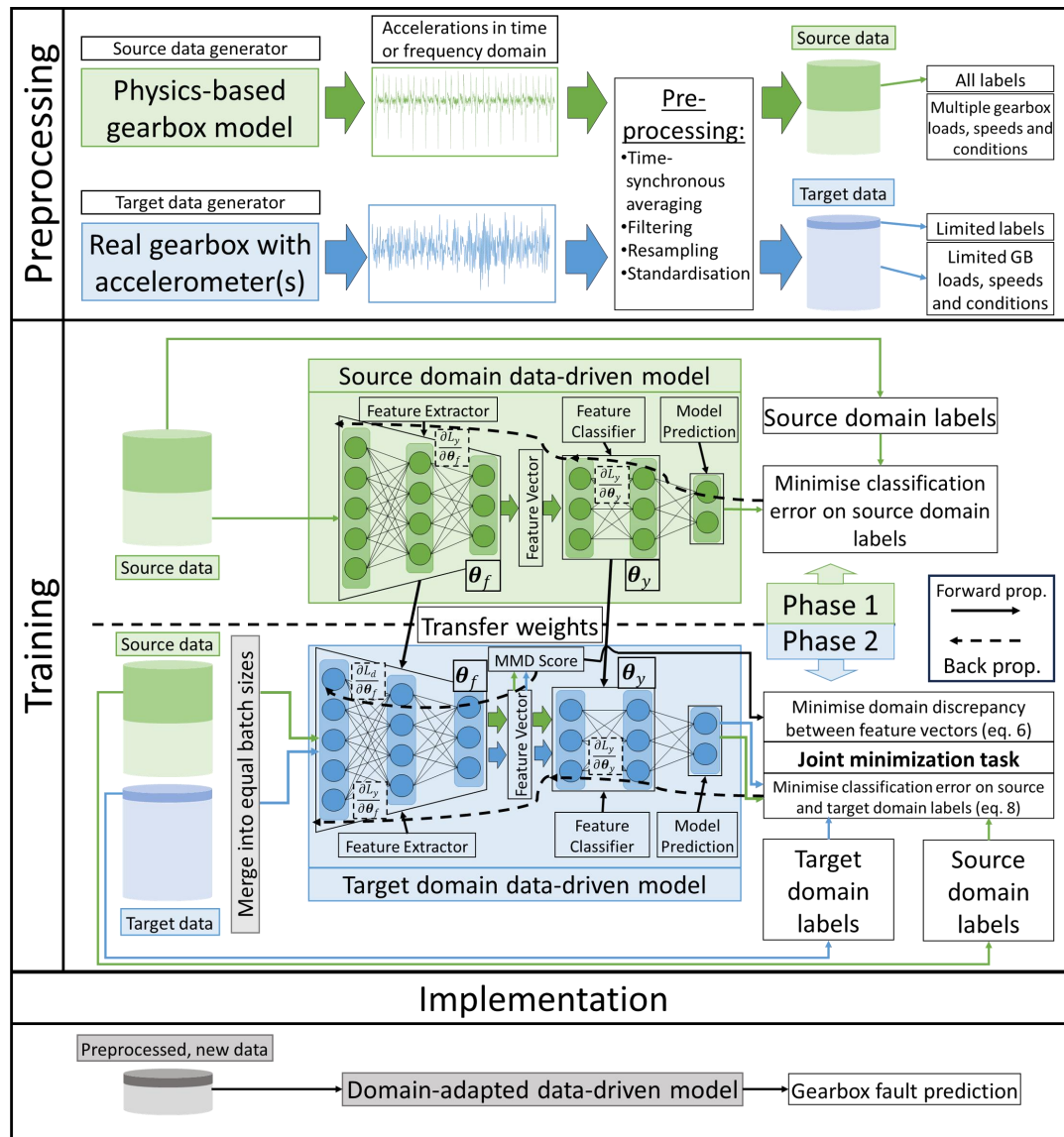


Fig. 1. The simulation-driven domain adaptation (SDDA) methodology uses physics-based simulated gearbox data in conjunction with real gearbox data to detect gearbox faults. Both simulated and real data are preprocessed uniquely for each application. For this paper, however, time-synchronous averaging, filtering, resampling, and standardisation steps are implemented (discussed in greater detail for each case study). Both time- and frequency-domain accelerometer data are considered in this work, and the relevant choice is clarified in the case study. The preprocessed data are compiled into source (simulated) and target (real) datasets. The source dataset has numerous load and speed conditions, each with associated health-state labels, whereas the target dataset has very limited operating conditions with limited labelled fault data samples, representing a few-shot learning problem. The model is trained in two phases. In phase 1, a data-driven model (DDM) is trained on the source domain data until the validation loss stops decreasing. Note the forward- and back-propagation steps. In phase 2, the DDM weights are transferred to an identical DDM with a modified loss function connected to the feature vector. Source and target data are batched together in equal-sized batches, which are fed through the feature extractor to create two sets of feature vectors – one for each data domain. These vectors are compared using a Multi-Kernel Maximum Mean Discrepancy (MK-MMD) loss function, forming the first half of the loss and representing the domain adaptation portion of the process. The feature vectors are passed together with all fault state labels to the feature classifier, creating a secondary loss between the model's fault prediction and known health label. Training is stopped when the MK-MMD score has been minimised and the classification validation loss stops decreasing. Take note of the altered forward- and back-propagation terms. At this stage, new, preprocessed, real gearbox data (or a testing set) may be passed to the domain-adapted DDM to predict whether gearbox faults are present in the input data. The physics-driven and data-driven models (DDMs) shown in this figure, are not prescribed; any applicable model may be used.

Given the relevant mass, damping, stiffness and force values, and an initial set of conditions for the displacement and velocity, [equation \(1\)](#) is solved using a Newmark algorithm [30] to obtain the unknown acceleration, velocity and displacements. The solved accelerations are the main quantities of interest, as they are analogous to the vibrations measured from accelerometers for real assets. A ten degrees of freedom lumped mass spur gearbox model is used in this work.

The model comprises a spur pinion and driven gear, a motor and load and a gearbox casing. The pinion and driven gear have radial  $x$  and  $y$  and axial rotation degrees of freedom. The motor and load each have an axial rotational degree of freedom and the gearbox casing has radial  $x$  and  $y$  displacement degrees of freedom. The layout and interaction between the ten degrees of freedom are shown in [Fig. 2](#).

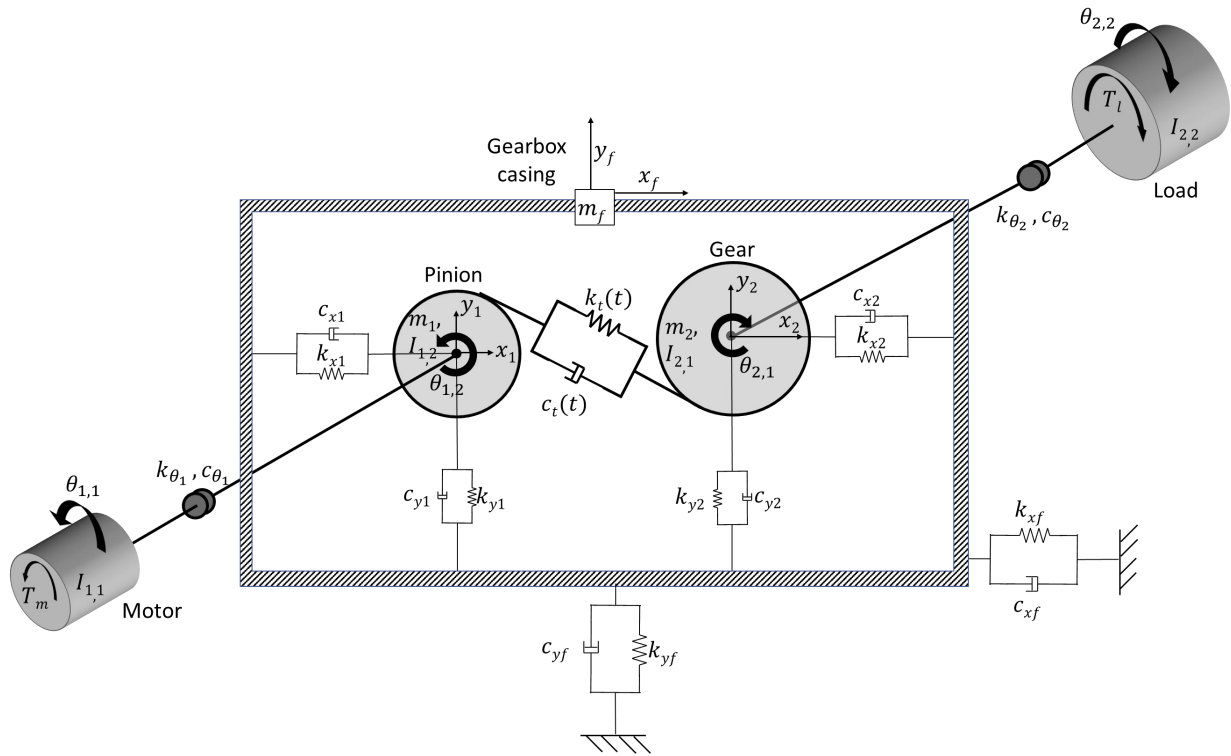


Fig. 2. Illustration of a physics-based gearbox model. There are ten degrees of freedom, namely four rotational for the motor ( $\theta_{1,1}$ ), load ( $\theta_{2,2}$ ), pinion gear ( $\theta_{1,2}$ ) and driven gear ( $\theta_{2,1}$ ), four translational for the pinion ( $x_1$  and  $y_1$ ) and gear ( $x_2$  and  $y_2$ ) and two translational degrees of freedom for the gearbox casing ( $x_f$  and  $y_f$ ). The pinion and driven gear have masses of  $m_1$  and  $m_2$ , respectively. The motor, pinion, driven gear, and load have rotational inertias of  $I_{1,1}$ ,  $I_{1,2}$ ,  $I_{2,1}$  and  $I_{2,2}$ , respectively. The motor-pinion and driven gear-load connections are modelled using stiffness coefficients,  $k_{\theta_1}$  and  $k_{\theta_2}$ , and damping coefficients,  $c_{\theta_1}$  and  $c_{\theta_2}$ . The pinion-driven gear connection is modelled using time-varying stiffness and damping coefficients,  $k_t(t)$  and  $c_t(t)$ , respectively. The pinion-casing and driven gear-casing connections are modelled in both  $x$ - and  $y$ -directions using stiffness and damping coefficients ( $k_{x1}$ ,  $k_{y1}$ ,  $c_{x1}$ , and  $c_{y1}$ ) for the pinion and driven gear ( $k_{x2}$ ,  $k_{y2}$ ,  $c_{x2}$ , and  $c_{y2}$ ), respectively. Similarly, casing-environment connections are modelled by  $x$ - and  $y$ -direction stiffness ( $k_{xf}$  and  $k_{yf}$ ) and damping ( $c_{xf}$  and  $c_{yf}$ ) coefficients.

### 2.1.2 Stiffness modelling of two interacting gears

Within the dynamic model, there are static and dynamic stiffnesses, such that:

$$\mathbf{K} = \mathbf{K}_s + \mathbf{K}_v, \quad (2)$$

where  $\mathbf{K}_s \in \mathbb{R}^{N \times N}$  and  $\mathbf{K}_v \in \mathbb{R}^{N \times N}$  refer to the static and varying stiffness matrices of the lumped parameter

model. Static stiffnesses (see  $k_{a,i}$ , where  $a = x, y$  or  $\theta$  and  $i = 1$  or  $2$  in [Fig. 2](#)) are stiffness values which do not change over the duration of the simulation and are summarised in [Appendix A.1](#). The varying stiffness (see  $k_t(t)$  in [Fig. 2](#)), however, varies over time and is modelled separately. The varying stiffness arises due to two main factors, namely the number of teeth in contact between two interfacing gears and the changing contact position between the two gears' teeth. The potential

energy method (PEM) [31] is utilised to simultaneously model both factors.

The PEM utilises beam theory to convert the interactions between teeth to theoretical stiffnesses as a function of gear rotation angle. Therefore, the varying stiffness value can only be updated as the dynamics are iteratively solved because  $\mathbf{K}_v$  depends on the solved gear rotational displacement. A comprehensive description of the stiffness modelling procedure is presented in van Eyk et al. [32].

### 2.1.3 Gear fault modelling

The true utility of a physics-based gearbox model becomes apparent when simulating faults, enabling deeper insights into how different faults impact a gearbox. In this work, we consider four typical gear tooth fault models presented in the literature<sup>2</sup>, namely root cracks [34], chips [35], spalls [36], and surface pits [37]. These fault models capture the effect of tooth faults by effectively reducing the values of three key entities, namely the surface line contact, the tooth cross-sectional area, and the second moment of area in the beam equations from the PEM. This finally reduces the meshing stiffness at specific angular positions as the gear rotates.

As previously mentioned, the focus of this work is not on the gearbox fault modelling process, and therefore only a summarised version of the gearbox modelling process is outlined in Fig. 3. A more comprehensive description of the fault modelling procedure is presented in van Eyk et al. [38]. In essence, various tooth faults with set geometries and locations are modelled. These faults are converted to angle-dependent stiffness functions using the PEM, resulting in the fault-altered stiffness function  $k_t(t)$  seen in Fig. 2. The resulting stiffness function is substituted into the overall stiffness matrix in equation 1. Rayleigh damping is assumed with the damping matrix proportional to a linear combination of the mass and stiffness matrices, using a damping ratio  $\zeta = 0.07$ . All matrices are substituted into equation 1 for each time step in the Newmark integration process. This results in simulated vibration signals for various faults.

In summary, the physics-based gearbox model enables the generation of vast amounts of representative source domain fault (and healthy) data under numerous known, labelled operating conditions, which may be used to pre-train a DDM as shown in Fig. 1.

## 2.2 Data-driven model – Convolutional neural network

The SDDA method utilises a convolutional neural network (CNN) as the data-driven model shown in Fig. 1. A CNN is a sensible data-driven model for gearbox fault assessment because of its ability to extract periodic features within data [39]. Since this work focuses on vibration data, the CNN architecture is used to recognise periodic time- or frequency-domain faults. The method is intentionally agnostic to the input domain or specific signal processing procedures, acknowledging that different assets may exhibit properties favouring particular domains or preprocessing techniques. Additionally, practical constraints such as memory availability may limit the choice of preprocessing strategies. Therefore, the remainder of this paper considers both the time- and frequency-domains, applying signal processing choices tailored to the characteristics of each problem.

The literature on CNNs is quite comprehensive as they have become fairly standard and widely used and are therefore not repeated here<sup>3</sup>. Instead, a conceptual explanation of the CNN is given to aid in later explanations of the domain adaptation procedure.

The CNN comprises three key parts: a feature extractor, a feature vector, and a feature classifier. The architecture used throughout this work is based on the work of Han et al. [3] and can be seen in Fig. 4. The dropout layers are used as a regularisation tool for the CNN and are set at a 10 % dropout probability. In some investigations, batch-normalisation layers are also added after dropout layers to aid loss minimisation convergence, but this is explicitly stated when used. All layers utilise the rectified linear unit (ReLU) activation function, except the output node, which utilises cross-entropy for binary or multi-class classification tasks. Adaptive moment estimation (Adam) is selected as the CNN optimiser with a default learning rate of 0.001.

The SDDA method relies on a feature vector for domain alignment. Therefore, although a CNN is described, the SDDA method may be implemented with any data-driven model that generates a feature vector.

<sup>2</sup> For a short but comprehensive review of various gear fault models, see van Eyk et al. [33].

<sup>3</sup> For a more comprehensive explanation of the CNN, refer to Lei et al. [1].

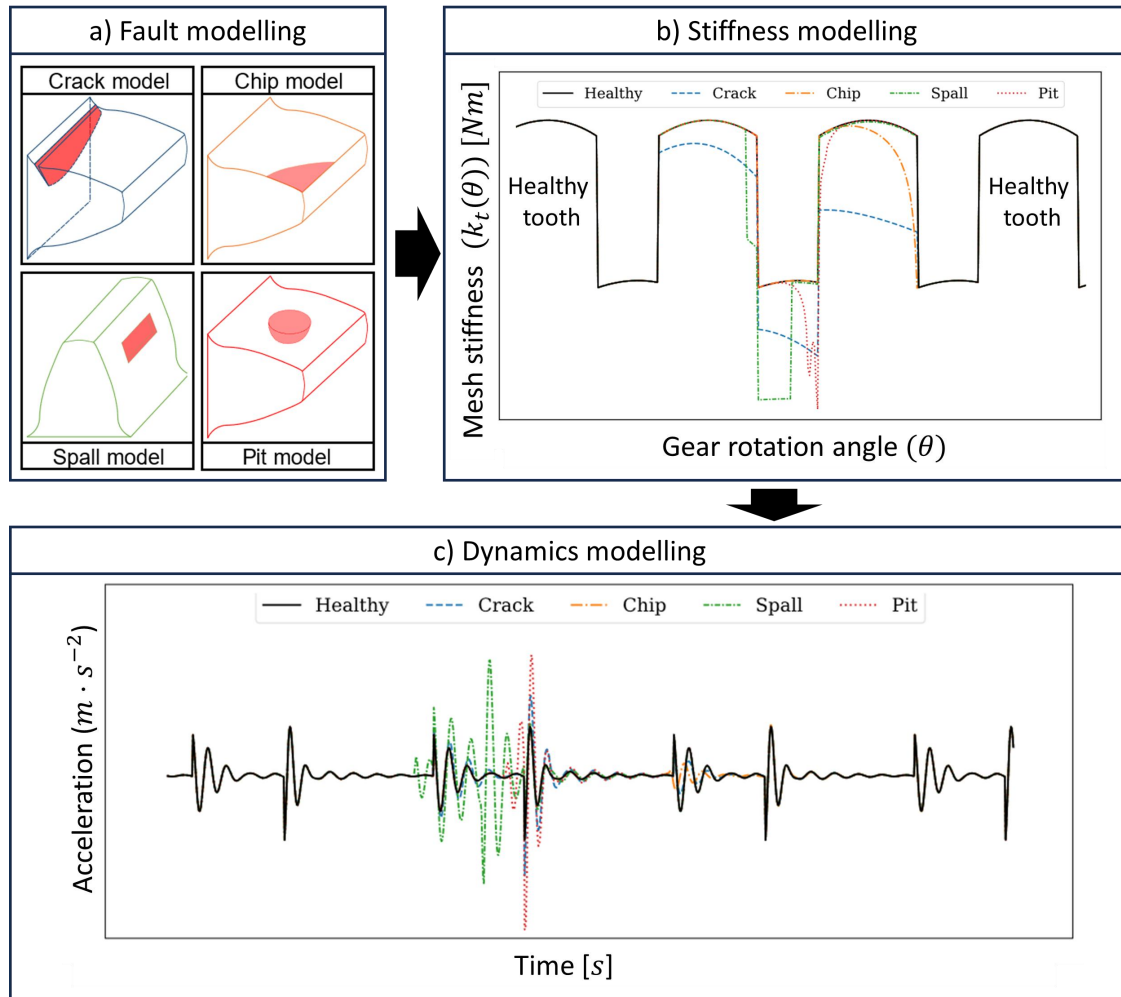


Fig. 3. Visualisation of how faults are introduced into the dynamic model. a) Four faults are modelled on the spur tooth, namely a crack, chip, spall, and pit. Different numerical fault models are utilised for spur gear teeth. The geometry of these faults may be varied to create a wide array of fault scenarios. b) The fault parameters from a) are converted to the corresponding reductions in mesh stiffnesses through the potential energy modelling method. c) A given meshing stiffness ( $k(\theta)$ ) is converted to a time-varying mesh stiffness, ( $k_t(t)$ ) – See Fig. 2. The time-varying mesh stiffness is incorporated into the varying stiffness matrix ( $\mathbf{K}_v$ ). The equation of motion of the ten DOF lumped parameter model in Fig. 2, namely  $\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{f}(t)$ , is solved using a Newmark algorithm. The resulting solution for the accelerations ( $\ddot{\mathbf{u}}(t)$ ) represents the simulated gearbox vibration signals over time, which are used to construct the source domain dataset. Depending on the input fault parameters, the corresponding output vibration signals vary.

### 2.3 Automatic feature calibration through domain adaptation

The foundation of the few-shot simulation-driven methodology evaluated in this work is rooted in transfer learning. Transfer learning facilitates the transfer of knowledge between the simulated source domain and real target domain data. Various transfer mechanisms are described in the transfer learning literature [1]. This work utilises a technique referred to as domain adaptation. Recent articles related to transfer learning in gearbox fault assessment successfully utilise some form

of domain adaptation [11,24,25]. It is therefore expected to be a strong candidate for evaluating few-shot simulation-driven gearbox fault assessment.

Domain adaptation attempts to align the feature vector distributions between the source and target domains. Simultaneously, the feature classifier attempts to create separable class mappings from the aligned feature vector distributions, given the numerous labels from the source domain and limited labels from the target domain [40]. The domain adaptation approach attempts to produce aligned feature distributions by adding a feature

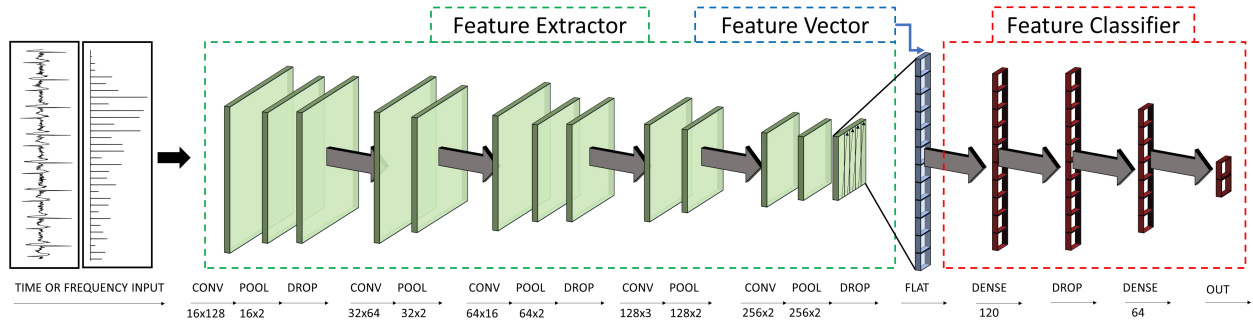


Fig. 4. CNN structure used in this work. Notice the three key components of the CNN, namely the feature extractor, the feature vector and the feature classifier. The feature extractor uses convolution, pooling and dropout layers to extract periodic features from the input data (input data may be time- or frequency-domain accelerometer data based on the application). The more convolution and pooling layers that are stacked, the richer the features become. Dropout layers are used to regularise network weights. At the final layer of convolution and pooling, the resulting set of matrices is flattened into a single vector, which is referred to as the feature vector. The feature vector is then passed through a normal multi-layer perceptron (MLP) network structure, which is referred to as the feature classifier. The feature classifier classifies the input feature vectors into the relevant gearbox condition.

discrepancy loss term to the DDM structure. The feature discrepancy loss is added at the feature vector layer, penalising large discrepancies between source and target feature embeddings. The domain adaptation structure results in a final loss term that simultaneously incentivises meaningful and aligned features between the source and target domains.

The feature discrepancy loss may be constructed through adversarial [19,41] or analytical [42,43] methods. The focus of this work is to determine whether explicit domain alignment improves few-shot simulation-to-real gearbox fault assessment relative to non-adaptation baselines. Accordingly, Maximum Mean Discrepancy (MMD) is used here as a representative and established analytical domain adaptation mechanism. This paper does not claim MMD to be superior to alternative domain adaptation strategies, it is merely selected based

on proven use [42,43]. Although alternative domain adaptation mechanisms may yield further improvements, the present study focuses on systematically evaluating an SDDA pipeline rather than optimising its individual sub-elements. MMD has been described in various works, such as those found in refs. [44,45] and therefore only a brief overview is given here.

MMD measures the similarity between two distributions by comparing all of their moments encoded in a reproducing kernel Hilbert space (RKHS). In the context of domain adaptation, the measured distributions correspond to the feature representations of source and target domains. Assuming equal batch sizes  $N$  between the source and target domains, the empirical MMD loss function is defined by Gretton et al. [44] using the biased estimator as:

$$\text{MMD}^2(\mathbf{X}_s, \mathbf{X}_t) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N k(\mathbf{x}_{s,i}, \mathbf{x}_{s,j}) + \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N k(\mathbf{x}_{t,i}, \mathbf{x}_{t,j}) - \frac{2}{N^2} \sum_{i=1}^N \sum_{j=1}^N k(\mathbf{x}_{s,i}, \mathbf{x}_{t,j}), \quad (3)$$

where  $\mathbf{X}_s \in \mathbb{R}^{N \times L}$  and  $\mathbf{X}_t \in \mathbb{R}^{N \times L}$  denote the batches of source and target domain feature matrices, respectively.  $L$  refers to the number of elements in the feature vector. The  $i^{\text{th}}$  and  $j^{\text{th}}$  source and target batch pairs are denoted by  $\mathbf{x}_{s,i} \in \mathbb{R}^L$  and  $\mathbf{x}_{t,j} \in \mathbb{R}^L$ , respectively. The function  $k(\cdot, \cdot)$  denotes a positive-definite kernel function. This work utilises the Gaussian kernel as outlined in Gretton et al. [44].

To extend this approach, a multi-kernel version of MMD (MK-MMD) is employed to capture feature discrepancies across multiple kernel scales. Long et al. [43] highlight that the selection of kernel bandwidth is

critical for effective domain adaptation; hence, a combination of kernels at different scales increases the likelihood of accurately identifying distributional differences, compared to a single potentially suboptimal kernel. In this work, a modified kernel, denoted  $k^*(\mathbf{x}, \mathbf{y})$  is defined as a convex combination of multiple Gaussian kernels:

$$k^*(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^K \beta_k k(\mathbf{x}, \mathbf{y}, \sigma_k), \quad (4)$$

where the weights satisfy  $\beta_k \geq 0$  and  $\sum_{k=1}^K \beta_k = 1$ . In this work, uniform weights  $\beta_k$  are employed for all kernels with standard deviations chosen as:

$$\sigma_k \in \{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1.5, 10, 35, 100, 10^3, 10^4, 10^5, 10^6\}. \quad (5)$$

The same biased empirical MMD estimator is then applied using this modified kernel:

$$\text{MK-MMD}^2(\mathbf{X}_s, \mathbf{X}_t) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N k^*(\mathbf{x}_{s,i}, \mathbf{x}_{s,j}) + \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N k^*(\mathbf{x}_{t,i}, \mathbf{x}_{t,j}) - \frac{2}{N^2} \sum_{i=1}^N \sum_{j=1}^N k^*(\mathbf{x}_{s,i}, \mathbf{x}_{t,j}) \quad (6)$$

The MK-MMD function is added to the feature vector layer, acting as a feature discrepancy loss function:

$$\mathcal{L}_d(\boldsymbol{\theta}_f) = \text{MK-MMD}(\mathbf{X}_s(\boldsymbol{\theta}_f), \mathbf{X}_t(\boldsymbol{\theta}_f)), \quad (7)$$

where  $\boldsymbol{\theta}_f$  represents the feature extractor weights.

The overall training loss function for the CNN modified with the MK-MMD method is defined by:

$$E(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y) = \mathcal{L}_{y,s}(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y) + \mathcal{L}_{y,t}(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y) + \lambda \mathcal{L}_d(\boldsymbol{\theta}_f), \quad (8)$$

with

$$\mathcal{L}_y(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y) = - \sum_{n=1}^N \sum_{c=1}^M y_{c,n} \log(p_{c,n}(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y)), \quad (9)$$

where  $M$ , refers to the number of classes,  $y$  refers to a binary indicator if class label  $c$  is the correct classification for batch sample  $n$ . The  $n^{\text{th}}$  CNN prediction for the  $c^{\text{th}}$  class is denoted by  $p_{c,n}$ .  $\mathcal{L}_y$  and  $\boldsymbol{\theta}_y$  refer to the feature classifier loss and weights respectively. Losses from the source and target domains are labelled with subscripts  $s$  and  $t$ , respectively. The  $\lambda$  parameter controls the relative loss contribution between the feature discrepancy loss and the classification loss. Ganin et al. [41] state that a small initial  $\lambda$  value causes the feature discrepancy loss to be less sensitive during early training. As training progresses, one may increase the  $\lambda$  value to cause the domain discrepancy loss contribution to increase.

To minimise the given training loss ( $E(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y)$ ),  $\boldsymbol{\theta}_f$  and  $\boldsymbol{\theta}_y$  are adjusted such that optimal parameters ( $\hat{\boldsymbol{\theta}}_f, \hat{\boldsymbol{\theta}}_y$ ) can be found:

$$(\hat{\boldsymbol{\theta}}_f, \hat{\boldsymbol{\theta}}_y) = \arg \min_{\boldsymbol{\theta}_f, \boldsymbol{\theta}_y} E(\boldsymbol{\theta}_f, \boldsymbol{\theta}_y) \quad (10)$$

The SDDA method requires two separate training phases (refer to Fig. 1). In the first phase, labelled simulated source domain examples are supplied to prime the network weights. This means that the feature discrepancy loss contribution is set to 0. In the second phase, identical batch sizes are created for the source and target domains. Both batches are passed through the

feature extractor. Source batches are always accompanied by labels, whereas target samples do not necessarily need labels. In this work, however, we always assume target domain labels, albeit with very limited data. Once the feature vectors are obtained for both domains, the feature discrepancy loss is determined. The feature discrepancy loss is used to update the feature extractor, aimed at producing domain-invariant features. The feature vectors are simultaneously passed through the feature classifier with their corresponding labels. This results in a classification loss for each domain, which is used to jointly update the feature classifier and feature extractor. This process is repeated over all batches of samples for multiple epochs until convergence is reached. In this work, we define convergence as the point in training where the validation set loss has reached a minimum.

In conclusion, the SDDA methodology provides a structured framework for studying the impact of scarce fault data in gearbox fault assessment. By leveraging physics-based gearbox simulations to generate representative source domain data and calibrating a CNN-based DDM through MK-MMD-based domain adaptation, SDDA enables fault diagnosis and detection with only a few labelled examples in the target domain. Unlike traditional transfer learning studies that assume extensive unlabelled or labelled target datasets, the present evaluation focuses on truly low-data scenarios, aligning well with industrial constraints. The method's end-to-end structure, combined with explicit feature-level domain alignment between simulated source and real target data, provides a practical basis for assessing when simulation-driven few-shot transfer is effective. In the following sections, we empirically evaluate SDDA through controlled numerical studies and two real-world case studies, offering a comprehensive analysis of its few-shot performance across varying conditions and fault types.

### 3 Comprehensive numerical investigation: Identifying factors that impede knowledge transfer in a simulation-driven method

While SDDA is a promising approach for few-shot gearbox fault assessment, the conditions under which simulation-driven domain adaptation may succeed or fail remain underexplored. When creating a physics-based model of a real gearbox, factors relating to the measurement of the real gearbox need to be taken into account. Factors such as sensor placement and noise will play a role in the type and quality of data being captured. Furthermore, factors relating to the physics-based modelling process need to be considered, such as modelling fidelity and similarity to the real gearbox. If these factors are neglected or modelled incorrectly, they may drastically increase the discrepancy between the simulated source and real target domain, complicating the transfer task.

This section, to the authors' knowledge, presents the first controlled numerical evaluation of how these domain discrepancy factors affect few-shot simulation-driven knowledge transfer for gearbox fault assessment. A numerical investigation based on simulated target domain data allows:

- Control over variables such as environmental noise, sensor noise, and extraneous vibrations, all of which might otherwise affect an investigation with real target domain data.
- Rapid changes between gearbox geometries and data measurement locations without altering the underlying properties of the gearbox.
- Greater control over the fault mechanisms, sizes, geometries, and locations compared to an experimental rig.

Although simulated data is used in the target domain, the investigations remain structured to mimic few-shot simulation-to-real transfer tasks by limiting the number of labelled simulated target domain samples. The investigation consists of multiple transfer tasks, each inducing a deliberate discrepancy between the source and target domains. The results of this section serve as a diagnostic lens to anticipate the strengths and limitations of simulation-driven domain adaptation before applying it to real-world case studies.

The comprehensive numerical investigation evaluates the impact of various discrepancies across three levels of comparison when transferring gearbox knowledge from a simulated to a real domain:

1. **Transfer method:** The knowledge transfer task may be evaluated using various methods. Four are investigated (see Fig. 6), the first two being non-

transfer methods and the latter two being transfer-based methods:

- Target-only training: Neglect simulated source data and train the DDM only on limited target domain data.
- Source-only training: Rely on simulated data for training the DDM, neglecting real samples.
- Weight fine-tuning: Pre-train the DDM on the simulated data and fine-tune the DDM on the limited target domain samples.
- Domain adaptation: Jointly train a DDM with a blend of simulated and real data, simultaneously minimising domain discrepancy and classification loss.

#### 2. Physics-based modelling considerations:

Different modelling decisions may greatly impact the difficulty of knowledge transfer between the source and target domains. These include model fidelity, sensor or process noise, sensor placement, and similarity of parameters between the simulation model and the real gearbox.

3. **Transfer type:** The type of transfer task being attempted may also give an indication of the transfer difficulty, as stated in Lei et al. [1]. Therefore, we subdivide the physics-based modelling considerations (point 2 above) into two subcategories: transfer across different machines (TDM) and transfer in the same machine (TIM). The model parameter similarity investigation between source and target gearboxes is a TDM task, whereas the model fidelity, noise and sensor location investigations are TIM tasks.

The proposed investigations are summarised in Table 1 with each row representing a full transfer investigation with a unique difference between the source and target domain. All investigations aim to correctly predict one of five health states from the target domain test set. The five classes represent a healthy, cracked, chipped, spalled, or pitted pinion gear tooth. Sets of rows are grouped to investigate the physics-based modelling considerations, outlined under the "Simulation Consideration" column. These simulation considerations are further grouped into the transfer type, namely TIM or TDM.

The first TIM case investigates the effect of model fidelity. The physics-based gearbox model is a low-fidelity estimate of the high-fidelity real gearbox. This case, therefore, investigates the impact that a

Table 1. Setup of transfer learning experiments. Machines A and B refer to the gearbox parameters associated with Luo et al. [46] and Meng et al. [47], respectively, and are discussed in greater detail in [section 3.1](#). Six, eight, or ten DOF variants are used. Gaussian noise is added to simulated signals with signal-to-noise ratios (SNRs) of -10 dB, 0 dB or 10 dB. Simulated vibrations are extracted at the pinion, driven gear (denoted as "gear") or on the gearbox casing DOFs. All signals are taken along the  $x$ -directions as seen in [Fig. 2](#). The source domain data refers to the vast labelled dataset which is used to train the data-driven model. The target domain refers to the domain with severely limited data, to which knowledge must be transferred.

| Transfer Experiment |                          | Source Domain |     |          |                      | Target Domain |     |          |                      |
|---------------------|--------------------------|---------------|-----|----------|----------------------|---------------|-----|----------|----------------------|
| Transfer Type       | Simulation Consideration | Machine       | DOF | SNR [dB] | Measurement Location | Machine       | DOF | SNR [dB] | Measurement Location |
| TIM                 | DOF                      | A             | 6   | 0        | Pinion               | A             | 10  | 0        | Pinion               |
|                     |                          | A             | 6   | 0        | Pinion               | A             | 8   | 0        | Pinion               |
|                     |                          | A             | 8   | 0        | Pinion               | A             | 10  | 0        | Pinion               |
|                     | Noise                    | A             | 10  | no noise | Pinion               | A             | 10  | 10       | Pinion               |
|                     |                          | A             | 10  | no noise | Pinion               | A             | 10  | 0        | Pinion               |
|                     |                          | A             | 10  | no noise | Pinion               | A             | 10  | -10      | Pinion               |
|                     | Location                 | A             | 10  | 0        | Pinion               | A             | 10  | 0        | Gear                 |
|                     |                          | A             | 10  | 0        | Pinion               | A             | 10  | 0        | Casing               |
|                     | TDM                      | A             | 6   | 0        | Pinion               | B             | 6   | 0        | Pinion               |
|                     |                          | A             | 8   | 0        | Pinion               | B             | 8   | 0        | Pinion               |

discrepancy in the DOFs between source and target domains has on the transfer learning calibration. The second TIM case investigates the effect of environmental noise. The simulated source domain data is free of noise and can cause a discrepancy between what is measured on the real gearbox. Therefore, experiments with varying degrees of Gaussian noise in the target domain are conducted to understand the impact of noise on the transfer task. The third TIM case investigates the effect of the measurement location. Pinion gear vibrations are often used for analyses in gearbox modelling. However, actual gearbox measurements may occur at a different sensor location, such as on the gearbox casing. We therefore explore the effect that differences between source and target sensor placements may have.

The TDM case investigates the effect of having two completely different gearboxes. It may not always be possible to develop a sufficiently representative physics-based model to model the necessary dynamics in the target domain. Therefore, this investigation aims to determine the impact of limited model similarity between a source and target domain gearbox.

The following subsections describe the construction of the source and target domain datasets for the various transfer experiments outlined in [Table 1](#). This is followed by an explanation of the parameters used during the numerical investigation. Finally, the results are shown and discussed.

### 3.1 Data-processing: Construction of the source and target domains

To construct the source and target domain datasets, two distinct gearbox models with different degrees of freedom, types of noise and measurement locations are constructed. The two gearboxes are referred to as Machine A and Machine B and are constructed using the gearbox parameters from Luo et al. [46] and Meng et al. [47], respectively. The major difference between Model A and B is the differences in the stiffness values, damping values, and gear parameters (see [Appendix A.2](#)). Both models use 16 and 48 teeth for the pinion and gear, respectively. For Model A, six, eight, and ten DOF versions are utilised. For Model B, only six and eight DOF versions are utilised. The ten DOF version is based on the lumped parameter model given in [Fig. 2](#). The eight DOF version neglects the effects of gearbox casing DOFs ( $x_f$  and  $y_f$ ). The six DOF version further neglects the effect of the motor and load DOFs ( $\theta_{1,1}$  and  $\theta_{2,2}$ ). The corresponding modelling matrices required for each case are shown in [Appendix A.1](#).

Both gearbox models are simulated across five pinion gear fault classes, namely healthy, cracked, chipped, spalled, or pitted. The fault parameters utilised to construct the various fault classes may be found in [Appendix A.2](#). Faults are induced on a randomly selected pinion tooth.

For both models, the dynamics are simulated at five loads, namely at 5, 10, 15, 20, and 25 N.m. These loads are simulated for 1.1 s using a simulation frequency of 200 kHz. The first 100 ms are discarded as transient state. Both models are simulated at an input shaft speed of 20 Hz. For the six and eight DOF models, the pinion ( $x_1$ ) and gear ( $x_2$ ) horizontal accelerations are extracted. For the ten DOF model, the gearbox casing horizontal acceleration ( $x_f$ ) is additionally extracted.

The resulting vibrations are infused with Gaussian noise signatures such that the new signal results in one of three signal-to-noise ratios (SNR (dB)), namely -10 dB, 0 dB, and 10 dB, where:

$$\text{SNR (dB)} = -20 \log_{10} \left( \frac{A_{\text{sig}}}{A_{\text{noise}}} \right), \quad (11)$$

with  $A_{\text{sig}}$  representing the root mean square amplitude of the clean, target domain signal, and  $A_{\text{noise}}$  representing the root mean square amplitude of Gaussian noise.

For a given load, noise signature and fault class, the source and target domain signals are standardised relative to the corresponding healthy class signal. The assumption is that in practice, newer signals will be standardised relative to older, healthier signals. This simulation process is repeated 50 times over, resulting in 1250 unique, labelled samples – 5 potential fault classes at 5 different loads with 50 simulations each. Each simulation is unique because the Gaussian noise signatures added to each simulation are randomly sampled. The set of 1250 signals is converted to the frequency domain because it removes any temporal information related to the faults. This is advantageous, as

it removes the necessity to train the data-driven model to recognise faults in multiple temporal locations. Due to memory constraints, the first 10000 frequency points, which conveniently equate to frequencies up to 10000 Hz, are used for all investigations. Healthy frequency spectra examples from the eight DOF models from both machines are illustrated in Fig. 5.

### 3.2 Setup of numerical investigations

Recall that four DDMs were selected for comparison. These models are all trained to classify between five gearbox health states using the CNN architecture described in section 2. All investigations use the same source domain training set of 1250 samples and a severely restricted target domain training set of only 54 samples (50 healthy samples and one per fault class), mimicking a few-shot fault diagnosis scenario. Training is performed using the Adam optimiser with a learning rate of 0.001, unless otherwise specified. The models are evaluated on a common target domain test set comprising 1196 samples (1250 samples minus 54 for training). The four DDMs differ primarily in their use of source and target domain data, and their approach to training and domain alignment. Each model is summarised below:

- **Target-only** (Fig. 6a): This model is trained solely on the 54 labelled target domain samples. It provides a lower-bound benchmark that reflects the information contained in the limited target data alone. Due to the extremely small training set, a validation split is not possible. Training is halted when the DDM reaches maximum training accuracy.
- **Source-only** (Fig. 6b): This model is trained exclusively on the complete source domain

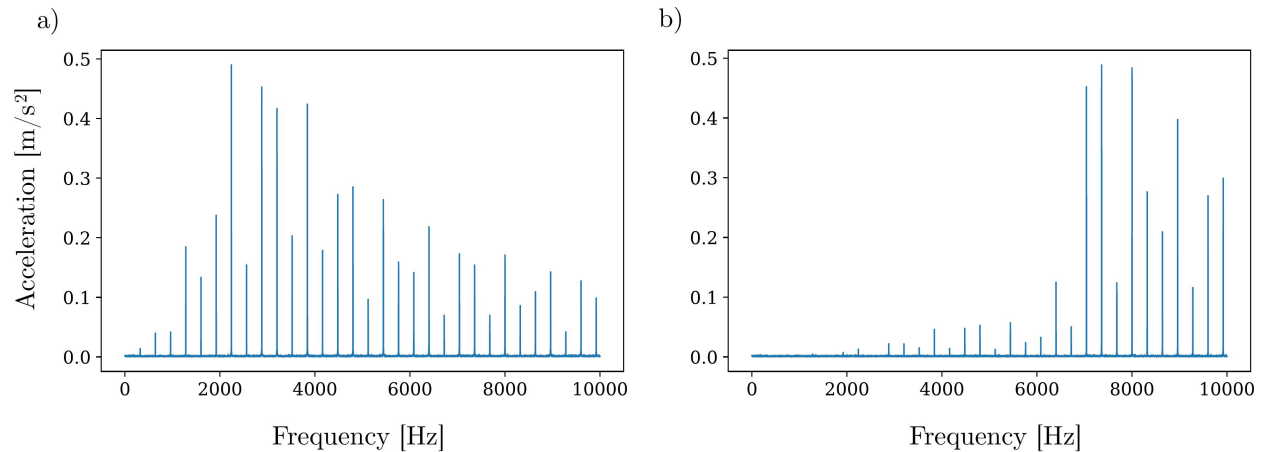


Fig. 5. Healthy frequency spectra for the eight DOF variants of a) Model A and b) Model B running at 20 Hz with 10 dB Gaussian noise added in the time-domain. Clear differences between the frequency spectra of the two models are visible, which makes the transfer task between these two machines non-trivial. These differences may be attributed to the vastly different stiffness and damping values used between the two models.

dataset. It serves as an indicator of domain similarity: if the source domain is representative of the target domain, this model is expected to generalise well to the unseen target test set. A 20 % validation split from the source data is used, and training is terminated once the validation loss stops decreasing.

- **Weight Fine-Tuning (WFT)** (Fig. 6c): The WFT model begins with the same training procedure as the source-only model. Once pre-training is complete, the model is fine-tuned using the limited target domain training data. During fine-tuning, the learning rate is reduced to 0.0001. Training is stopped when the target domain training loss ceases to decrease. This approach serves as a strong baseline for transfer learning performance.
- **Simulation-Driven Domain Adaptation (SDDA)** (Fig. 6d): The SDDA model jointly

utilises data from both the source and target domains, incorporating a domain adaptation loss based on the MK-MMD criterion (see section 2.3). Because the target domain contains only 54 samples, samples are duplicated across health states to balance the number of samples per class with the source domain, resulting in 250 samples per fault class. Pre-training is first performed on the source domain using a 20 % validation set, with the domain discrepancy term  $\lambda$  in equation (8) initially set to zero. After convergence, training continues for 300 epochs using joint batches of 30 samples (15 from each domain), while  $\lambda$  is incrementally increased by 0.1 every 10 epochs to gradually introduce the domain adaptation loss.

A summary of the aforementioned DDMs’ investigation parameters is shown in Table 2

3.3 Results

The average five-class classification results from running the transfer learning investigations in Table 1 over 15 repetitions are shown in Fig. 7. More detailed,

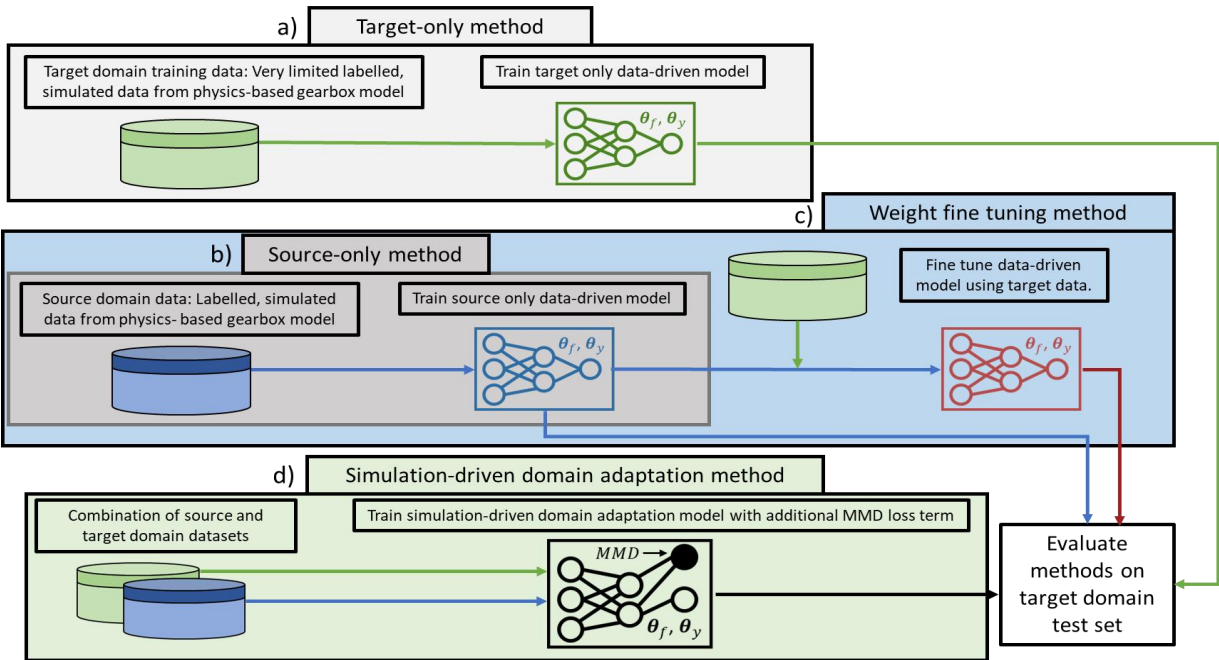


Table 2. Summary of the setup used to conduct the numerical investigations. Here, the columns indicate the source-only, target-only, weight fine-tuning, and simulation-driven domain adaptation methods. The target domain model is trained on the limited target domain data. b) The source-only model represents the case where a data-driven model is trained on only the source domain data but applied to the target domain data. This represents the expected performance of an uncalibrated simulation-to-real model. c) The weight fine-tuning model is an extension of the source-only method, in the sense that it starts from the source-only trained model, and further fine-tunes the weights with the limited target domain data. A comprehensive overview of the process is illustrated in Fig.

| Model         | Source | Target | Source & target | SDDA           |
|---------------|--------|--------|-----------------|----------------|
| Optimizer     | Adam   | Adam   | Adam            | Adam           |
| Learning rate | 0.001  | 0.001  | 0.001 → 0.0001  | 0.001 → 0.0001 |

| Training data    | Source                                                | Target | Source & target |
|------------------|-------------------------------------------------------|--------|-----------------|
| Data type        | Frequency domain: Pinion $x$ , gear $x$ or casing $x$ |        |                 |
| Data format      | 10000 data points to capture frequencies below 10 kHz |        |                 |
| Health states    | Healthy, crack, chip, spall, pit                      |        |                 |
| Load cases       | 5, 10, 15, 20, 25 Nm                                  |        |                 |
| Speed cases      | 20 Hz motor input                                     |        |                 |
| Validation split | 20 %                                                  | None   | 20 % → None     |

per-class results are given in [Table B.1](#) and [Table B.2](#) in [Appendix B](#). The goal of the comprehensive numerical investigations was to determine how common factors that can cause domain discrepancy affect simulation-to-real knowledge transfer in few-shot gearbox fault assessment. These impacts are most sensibly evaluated in reference to the four methods.

The source-only method results are good indicators of the level of domain discrepancy present for the transfer tasks. Model fidelity ([Fig. 7a](#)) does not contribute significantly to domain discrepancy because the source-only model was able to correctly classify results with accuracies above 97.5 %. [Fig. 7b](#)) suggests that low noise (SNR = 10 dB) scenarios represent no domain discrepancy, as expected. However, for more severe noise cases (SNR = 0 dB and SNR = -10 dB), larger domain discrepancies are present, falling to classification accuracies as low as 29.1 %. For the pinion to gear transfer case ([Fig. 7c](#)), no domain discrepancy was observed as the source-only model achieved 100 % classification accuracy. The pinion to casing transfer case ([Fig. 7c](#)) resulted in a larger domain discrepancy, with the source-only method only achieving 61.9 % classification accuracy. This makes intuitive sense as the pinion and gear directly mesh with each other, and therefore, measured vibrations for these two components are very similar, likely having a linear relationship. The casing, however, is not directly in contact with the pinion and vibrations are filtered through various paths to the casing, likely resulting in a non-linear relationship between pinion and casing vibrations. Finally, when observing the two different machine transfer cases ([Fig. 7d](#)), the strongest cause for domain discrepancy was

observed, with source-only accuracies of 25 % and 29.6 % for both TDM tasks. These results are in line with expectations because Machines A and B have different parameters, and therefore their domains of data are expected to be different. These results, as a collective, showcase the increased difficulty in performing a TDM transfer.

The target-only method represents a baseline of the level of knowledge contained in the limited target domain samples. This model represents the baseline that the transfer learning methods should, at minimum, match. Therefore, results from this model are not considered in isolation, but rather in comparison to the transfer learning methods.

The WFT results showed that this method could match or outperform the target-only method in all cases. This shows that the WFT method can incorporate information from the source domain to increase target domain classification performance. Interestingly, the same cases which presented difficulties for the source-only model presented difficulties for the WFT model, bringing the accuracy down to the same levels as the target-only method (see [Fig. 7d](#)). This is expected, as the WFT model mainly relies on the source domain pre-training step, causing the WFT model to overfit on this relatively different domain of data. Therefore, one can conclude that the WFT model is only effective where domain discrepancies are small, but may become ineffective with larger domain discrepancies.

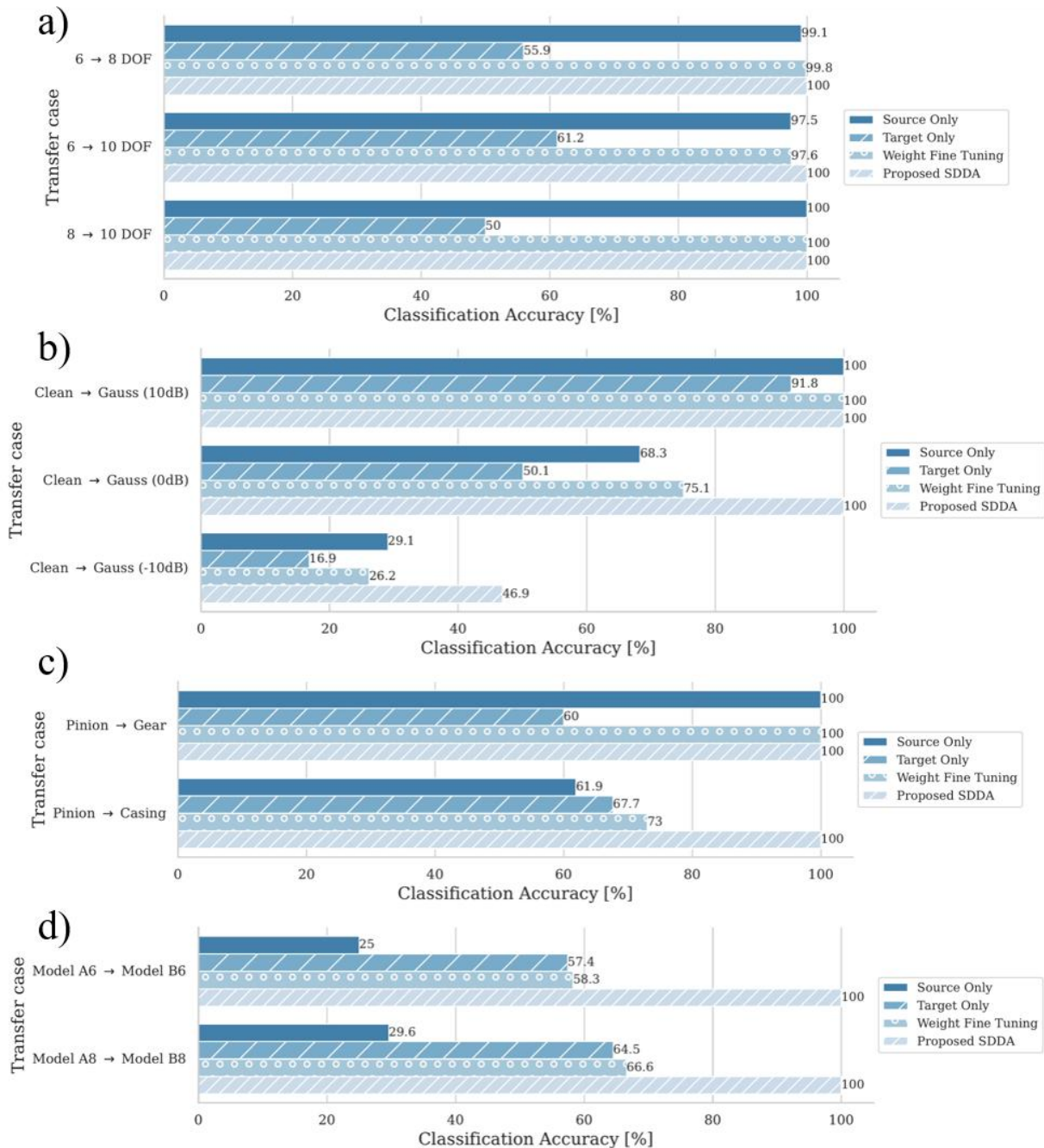


Fig. 7. Compiled results of the numerical investigation. The three TIM investigations are given by a) change in DOF, b) change in noise, c) change in vibration location and the TDM investigation is given by d) change in machine.

The SDDA method showcased 100 % accuracy on all but one transfer task and outperformed all other methods in all transfer tasks. In only a single case, namely the SNR = -10 dB noise case from Fig. 7b), the model achieved only 46.9 % accuracy. It is important to note that an SNR of -10 dB is quite severe, and decreased performance is expected. The results as a collective

indicate that the SDDA method is much less sensitive to larger domain discrepancies when compared to the WFT method. The explicit minimisation of domain discrepancy proved to be a sensible few-shot approach for transferring knowledge across different domains in comparison to weight fine-tuning.

Considering the three investigation criteria outlined at the start of this section, the following conclusions are drawn:

- **Transfer method:** The SDDA method outperforms the other transfer learning approach on all datasets. This indicates that the explicit minimisation of domain discrepancy through DA is a sensible way to achieve high classification accuracy for few-shot fault diagnosis.
- **Transfer type:** TDM transfer tasks are seen to be significantly more difficult when compared to their TIM counterparts. This is due to the increased discrepancy between the source and target domains arising from two different models' data being used between the source and target domains. Although the TDM task had a demonstrably larger impact on transfer performance, it was not large enough to impede the SDDA approach, which still achieved 100 % classification accuracy.
- **Physics-based modelling considerations:** Three modelling factors caused large domain discrepancies and deserve a short discussion below.

Firstly, a mismatch in modelling parameters between the source and target domain models resulted in the largest domain discrepancies. Therefore, when possible, the physics-based model must be calibrated to better match reality. However, as demonstrated in this study, the SDDA method could automatically calibrate for these differences, allowing flawless fault classification. Secondly, sensor location was seen to be an important contributor to causing domain discrepancies. Therefore, it is very important to understand where the target domain data is measured before constructing the physics-based counterpart. Finally, large domain discrepancies were caused by low SNR ratios. This factor, however, cannot necessarily be improved from physics-based modelling, and better target domain data acquisition and signal processing are likely more effective. It is, however, likely sensible to include some noise in the simulated source domain data to ensure the DDM does not overfit to clean data, allowing for a more robust DDM model that is less sensitive to noisy measurements.

The numerical investigations show that SDDA consistently outperforms the alternative approaches considered in this study, especially under severe domain discrepancy conditions. It exhibits improved robustness to noise, reduced physics-based modelling fidelity, and sensor misalignment, while remaining viable in cross-machine (TDM) transfer scenarios. These findings provide controlled evidence that explicit feature-level domain alignment is beneficial for few-shot simulation-driven fault classification when real-world fault data are sparse or difficult to collect. However, while the numerical studies isolate and control individual sources of domain shift, real-world systems often present

multiple overlapping discrepancies. To further evaluate whether these numerical trends extend to practical simulation-to-real conditions, we next investigate SDDA's performance.

## 4 Case study 1: PHM23 dataset

Having established the SDDA workflow and its behaviour under controlled numerical discrepancies, we now evaluate its performance in a real-world scenario where good modelling alignment between simulated and real domains is possible. This first case study investigates the fault detection performance of SDDA using the PHM23 dataset [48], focusing on few-shot learning conditions. Importantly, the physics-based simulation is configured to approximate the real gearbox's structure, sensor placement, and operating conditions, thereby reducing domain discrepancy. This setup allows us to assess the effect of low sample availability and benchmark SDDA's fault detection capability in a relatively well-aligned simulation-to-real transfer scenario.

The PHM23 experiments were conducted on a single-stage spur gearbox setup. A layout of the setup is provided in Fig. 8, with important gearbox parameters listed in Table A.3.

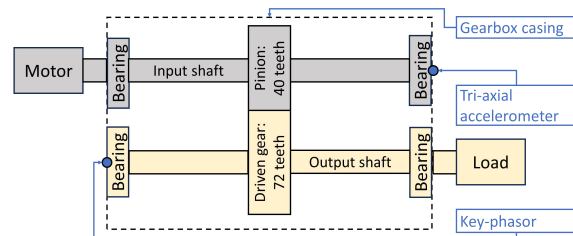


Fig. 8. Schematic of PHM23 gearbox. The gearbox is a single-stage spur gearbox with 40 pinion and 72 driven gear teeth. The gearbox has a tri-axial accelerometer mounted to the casing on the output side. A key phasor is located on the output shaft.

The test rig was subjected to a comprehensive set of operating conditions, including varying loads, rotational speeds and levels of pitting damage. For the purposes of this case study, all labelled combinations were considered, thereby constraining the dataset to a defined set of discrete load and speed combinations. Specifically, the rotational speeds are selected from  $\Omega = \{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000\}$  rpm and the applied load at the output shaft are selected from  $T = \{50, 100, 200, 300, 400, 500\}$  Nm. This results in a total of 60 unique load-speed operating conditions, each with varying levels of pitting damage.

Pits were induced with a drill bit through a lubrication cover, without the necessity to disassemble the gearbox. The pits were reported to potentially be distributed on more than one tooth. Pits were induced at increasing levels up to level 10. Therefore, 11 fault intensities exist, with 0 being healthy and 10 being the worst-case pitting. Note that fault intensities 5, 7, 9, and 10 do not have labelled examples and are excluded from the dataset. Therefore, the remaining 7 fault intensities are considered.

Accelerometer and key-phaser data were recorded for every load, speed and pitting combination. The positions of the tri-axial accelerometer and key-phaser are shown in Fig. 8. Recorded signals contain three columns of accelerometer data and one column for key-phasers. Signals were sampled at 20480 Hz for multiple pinion rotations. However, due to the fixed sampling rate, for lower rpm runs, fewer rotations were captured in comparison to higher rpm runs, because the lower rpm runs require more time to complete a single revolution. The dataset, therefore, contains varying sample lengths due to the varying rpm values. For a machine learning task, all data inputs must be identical and cannot use varying input data. Therefore, all data samples are processed to result in time-domain signals that:

- Represent a single gear rotation<sup>4</sup>.
- Retains high-frequency information from the original data.
- Are invariant to phase effects, which could occur due to misaligned key phasers.
- Are invariant to fault location.

These properties are selected to ensure that the resulting models are robust and do not overfit certain representations of the input data. Due to memory restrictions, the dataset is further restricted to only use the measured x-vibrations. The numerical investigations emphasised the importance of matching the sensor locations between source and target datasets. The simulated data will therefore only utilise the simulated x-vibrations. The following sections outline the data processing to obtain the relevant source and target domains, which will in turn be used to conduct the outlined experiments. Finally, the results are shown and discussed.

#### 4.1 Data-processing: Construction of the source and target domain

The source domain data is simulated at 200 kHz using the ten DOF gearbox model from Luo et al. [46] outlined in section 3.1. Simulation periods are varied according to the motor speed, similar to the PHM23 dataset

samples. The numerical investigations indicated that model fidelity had little effect on knowledge transfer; therefore, the existing ten DOF model is deemed sufficient for this investigation. The model parameters are changed to match those given in Table A.3. Where parameters are not defined, the default values from Appendix A.2 for the Luo et al. [46] gearbox are used. Simulations are conducted under the same 420 load and speed combinations considered for the PHM23 dataset. The PHM23 dataset contains multiple runs for all combinations of operating and fault conditions. These repetitions of runs under identical conditions are replicated in the simulations.

Pitting faults are generated using the same mechanism outlined in the numerical study (section 3), which utilised the model from Meng et al. [37]. Due to the lack of knowledge of the number and location of pits induced on the real gearbox dataset, numerical pits are induced by visual comparison to the original dataset. It is observed that the pits appear to be distributed along more than one tooth. Therefore, we generate pits on neighbouring teeth. The approach is inspired by the work of Liang et al. [49]. The number of pits ( $N_t$ ) for a given tooth ( $t$ ) based on fault intensity ( $I$ ) is assumed to have a linear relationship in the model:

$$N_t(I) = SF(t) \times \frac{294 \times I}{8} \quad (12)$$

where  $SF$  is the scaling factor to determine the number of pits for neighbouring teeth:

$$SF = \begin{cases} 1 & \text{if } t = T \\ \frac{1}{1.2} & \text{if } t = T - 1 \text{ or } t = T + 1 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

where  $T$  represents the main fault tooth. The authors estimated the real fault to be located around  $T = 13$ . Since the PHM23 dataset hides the fault generation mechanism, fault parameter choices here are fairly arbitrary and were visually calibrated to best match the shape of vibrations in the fault region for fault intensity 8 – the highest available labelled fault intensity. Any remaining mismatches in assumed vs unknowable real conditions need to be learned by the DDM, representing a realistic industrial scenario.

The pitting parameters are given in Table A.4. As described before, the PHM dataset is captured at 20 kHz. Therefore, to mimic this behaviour, after simulating all operating conditions at all fault levels, the source domain data is low-pass filtered at 8 kHz and resampled to 20 kHz to better match the target domain data. At this stage, the source and target datasets are similar in structure, and

<sup>4</sup> The data challenge does not specify which gear contained the pits. Therefore, to ensure that a fault signal will be present within a single rotation, the conservative choice is the gear with more teeth (driven gear).

further explanations of pre-processing apply to both domains of data.

Both domains contain varying-length samples with multiple rotations. Furthermore, the PHM23 dataset may still contain noise within the measured samples. Therefore, to simultaneously account for both factors, time synchronous averaging (TSA) is applied to all samples in the PHM23 dataset, resulting in samples that have less noise and represent a single rotation only. For the simulated dataset, TSA is not necessary as the data is generated in a noise-free environment. Thus, a single rotation is selected from the multiple simulated rotations to represent a simulated TSA equivalent.

The machine learning models used in this investigation require samples of equal length, and therefore, resampling is necessary for all samples. All signals are resampled to 16384 ( $2^{14}$ ) points to maximise the retention of high-frequency content within memory constraints. To address the desired fault location invariance and phase invariance outlined earlier, we apply two modifications to the process which generates single rotations. Firstly, we shift key-phasors by one quarter of a rotation. This process is repeated four times, leading to four unique locations where faults can occur. Secondly, the shifted key-phasor locations are randomly shifted by an integer number between -100 and 100. Considering the resampled signals contain 16384 points, this results in a new phasor location that deviates from the true value by a maximum of 0.6 %. The random shifts aim to develop a dataset which does not contain phase-related information. If a phasor sensor is slightly shifted between runs, the vibration signal may contain phase information which reflects this change, and false anomalies may be triggered<sup>5</sup>. Finally, for each dataset, signals are standardised relative to the dataset's mean and standard deviation. A visual comparison of a selection of standardised source and target domain samples is given in Fig. 9. The shape of healthy and faulty regions for the simulated and measured gearbox signals matches fairly well. Although in certain instances (such as the first row of Fig. 9) the amplitudes are not too similar, the fault location and shape remain similar. The simulated data is therefore deemed to be of acceptable quality to act as a simulated source domain.

Each processed signal is accompanied by two pieces of information that are used during model training. Firstly, the load and speed information is retained as scaled values between 0 and 1. Secondly, fault intensity is retained as the classification label.

<sup>5</sup> During experimentation, the phenomenon of overfitting to phase was seen, hence the application of slight shifts in key-phasor locations. These shifts successfully removed any phase-related overfitting.

## 4.2 Case study setup

A set of experiments is set up with varying numbers of target domain fault data samples. The goal for all DDMs is to obtain the highest fault detection accuracy on the target domain testing set. The DDMs in this study are trained to use time-domain data as input.

A summary of the source and target domain training and validation data is given in Table 3. For the source domain, 3000 healthy and 3000 faulty samples are generated at various speeds (100 to 1000 rpm as outlined earlier), loads (50 to 500 Nm as outlined earlier), pitting intensities (0 to 8) and permutations of key-phasors (index shifts randomly sampled from -100 to 100). A further validation set of 250 healthy and 250 faulty samples is generated. For the target domain, samples are purposefully restricted for the few-shot learning task. Only 320 unique healthy signals are selected, and at most, 128 fault signals. The healthy samples are split into training and validation sets of 240 and 80 samples, respectively. These samples are duplicated to match the size of the source domain, namely 3000 healthy training samples and 250 healthy validation samples. The number of available fault signals varies between 4, 8, 16, 32, 64 and 128 samples, which are split using a 67-33 % training-validation split. Rounding may cause slight deviations from this exact ratio. The fault samples are duplicated to match the source domain training and validation sets of 3000 and 250 samples, respectively. Considering that there are 1440 unique, labelled fault signals in the considered PHM23 dataset (six load cases, ten speed cases, six fault intensities (the seventh being healthy), and four possible fault locations), 4 and 128 labelled target domain fault samples represent 0.28 % and 8.9 % of the possible fault representations, respectively. The remaining target samples, which were not selected for training or validation, serve as the test set.

In this study, the four models from the numerical investigations are evaluated, namely the source-only, target-only, WFT and SDDA models based on the CNN architecture outlined in Fig. 4. Learning rates and optimisers are identical to those described in the numerical investigation. For this case study, three modifications are made to the CNN architecture. Firstly, batch normalisation layers are added after dropout layers, as they were seen to aid in convergence. Secondly, the scaled load and speed values that accompany each signal are concatenated with the feature vector. This is expected to aid the feature classifier. Finally, kernels and pooling layers are slightly changed to fit the given input size of 16384. The updated kernel and pooling layers are given in Table 4.

To provide a broader performance baseline, two additional models – Principal Component Analysis (PCA) and an Auto-Encoder (AE) – are also implemented using only target domain data, representing

typical linear and non-linear fault detection methods.

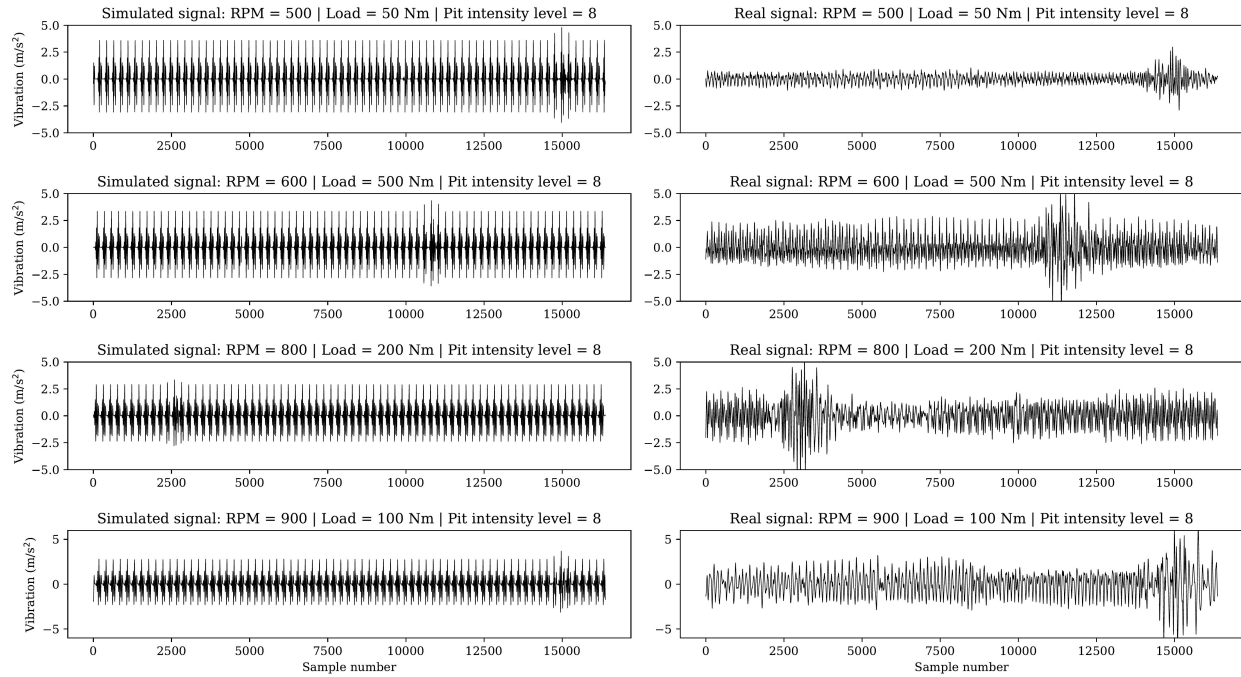


Fig. 9. Comparison between a subset of the simulated and real data. All signals have already been standardised. The maximum fault intensity was chosen, as it is the easiest to visually discern. Various load, speed and fault location combinations are plotted.

Table 3. Breakdown of the training and validation data for case study 1. The → symbol represents target domain samples that are duplicated to match the number of samples in the source domain.

| Dataset                          | Number of healthy training samples | Number of fault training samples | Number of healthy validation samples | Number of fault validation samples |
|----------------------------------|------------------------------------|----------------------------------|--------------------------------------|------------------------------------|
| Source (Simulated data)          | 3000                               | 3000                             | 250                                  | 250                                |
| Target (4 PHM23 fault samples)   | 240 → 3000                         | 2 → 3000                         | 80 → 250                             | 2 → 250                            |
| Target (8 PHM23 fault samples)   | 240 → 3000                         | 5 → 3000                         | 80 → 250                             | 3 → 250                            |
| Target (16 PHM23 fault samples)  | 240 → 3000                         | 10 → 3000                        | 80 → 250                             | 6 → 250                            |
| Target (32 PHM23 fault samples)  | 240 → 3000                         | 21 → 3000                        | 80 → 250                             | 11 → 250                           |
| Target (64 PHM23 fault samples)  | 240 → 3000                         | 42 → 3000                        | 80 → 250                             | 22 → 250                           |
| Target (128 PHM23 fault samples) | 240 → 3000                         | 85 → 3000                        | 80 → 250                             | 43 → 250                           |

Table 4. Updated CNN architecture for PHM23 case study. 'Conv', 'Pool', 'FV' and 'FC' refer to convolutional layers, max pooling layers, feature vector and fully-connected layers, respectively. Notice the '+2' in the feature vector, which corresponds to the concatenation of scaled speed and load values with the extracted signal features.

| Layer | Conv1   | Pool1 | Conv2   | Pool2 | Conv3   | Pool3 | Conv4  | Pool4 | Conv5   | Pool5 | FV      | FC1 | FC2 | Output |
|-------|---------|-------|---------|-------|---------|-------|--------|-------|---------|-------|---------|-----|-----|--------|
| Size  | 8 × 128 | 4     | 16 × 64 | 4     | 32 × 16 | 4     | 64 × 3 | 4     | 256 × 2 | 4     | 100 + 2 | 120 | 64  | 9      |

The PCA model is optimised by choosing the number of components that maximises the classification accuracy of the validation set. 240 components were found to be optimal for reconstruction. For comparison purposes, the AE model bottleneck layer was therefore also constrained 240 components. The AE uses a multi-layer perception (MLP) structure from the input space directly to the latent space, and the opposite MLP structure to go from the feature vector back up to the predicted input signal. Both models are trained on the 240 healthy source domain samples (as outlined in Table 3), and the validation set of 80 healthy samples is used to set an anomaly threshold. Due to the large cost associated with misclassifying faults, a high threshold value (TV) is determined at a 99.9 % significance level under a Gaussian data distribution assumption:

$$TV = \mu_L + 3\sigma_L \quad (14)$$

where  $\mu_L$  and  $\sigma_L$  represent the mean and standard deviation of the reconstruction loss from the healthy data validation set, respectively.

For the first four models, the goal is multi-class classification based on one-hot encoded labels of the 7 fault intensity levels. Therefore, they are trained as fault intensity classifiers. However, to make their results comparable to the two anomaly detectors, their class predictions are converted to either a health or fault prediction. A fault severity threshold of 1 is chosen, which means that only 0 labels are classified as healthy, otherwise, it is automatically classified as a fault. Similarly, the anomaly detectors use the defined threshold value, TV, to determine whether a sample is anomalous and, by extension, represents a fault.

All models are trained and evaluated over the six target domain training data scenarios seen in Table 3. This process is repeated 15 times to gather statistically significant results. Note that the few-shot labelled fault-sampling process is completely random for each repetition, avoiding any overfitting to training data or data leakage between training and testing sets. The resulting mean model predictive accuracy with one standard deviation and true negative rate is given in Table 6. A visual representation without the true negative rate is shown in Fig. 10. Model classification accuracy is defined as the true negative rate and true positive rate combined with equal weighting. A random guessing model would therefore obtain a 50 % accuracy,

and any effective model should outperform this lower bound. In the context of this study, the true negative rate is defined as the proportion of healthy samples predicted correctly. The true negative rate is relevant in the context of this study because false classifications of healthy signals as faulty may cause unnecessary downtime for a gearbox. A summary of the training parameters used in this case study is seen in Table 5.

### 4.3 Results

The aim of this case study was to analyse the few-shot fault detection performance of the SDDA method on a simulation-to-real task. It is helpful to contextualise the results of the SDDA method by first analysing the performance of the non-transfer learning methods. The source- and target-only methods represent useful baselines in their respective domains.

The source-only method showed a greater than 50 % average on all the fault detection tasks, which is above random guessing. This indicates that there is some degree of overlap between the simulated and real datasets. However, the overlap is not aligned, as the true negative rate for the source-only model was quite low. This implies the source-only model classifies most samples as faults, and only rarely correctly classifies healthy samples. The target-only model performed poorly when only a few target samples were available, and steadily increased with increasing number of labelled faults. This is expected and serves as the lower bound for a transfer-learning approach.

The anomaly detectors may be used as a baseline that describes how much information is encoded in the few healthy target samples. Both methods performed similarly to the source-only model but with improved true negative rates. The mediocre results indicate that the healthy target domain samples are not different enough from most fault samples, leading to poor anomaly detection. Importantly, the anomaly detectors were seen to outperform the target-only method for extremely low data cases. This behaviour was not unexpected because the target-only method was trained for a different purpose (multi-class fault classification) compared to the AE and PCA models (anomaly detection). The target-only method showcased increased performance when fault samples were increased, a property that the anomaly detectors do not have.

Table 5. Summary of the parameters used in case study 1. Here, the columns represent parameters for the source-only, target-only, principal component analysis, autoencoder, weight fine-tuning and simulation-driven domain adaptation methods.

| Model            | Source                                           | Target | PCA                   | AE    | WFT             | SDDA  |
|------------------|--------------------------------------------------|--------|-----------------------|-------|-----------------|-------|
| Optimiser        | Adam                                             |        |                       |       |                 |       |
| Learning rate    | 0.001                                            | 0.001  | N.A.                  | 0.001 | 0.001 → 0.0001  | 0.001 |
| Training data    | Source                                           | Target | Target (Healthy only) |       | Source & target |       |
| Data type        | Time domain: Driven gear $x$ accelerations       |        |                       |       |                 |       |
| Data format      | 16384 Data points representing a single rotation |        |                       |       |                 |       |
| Fault type(s)    | Pits: Multiple intensities                       |        |                       |       |                 |       |
| Validation split | Source: 8.3 %   Target: 33 %                     |        |                       |       |                 |       |

Table 6. PHM23 case study results. The results are presented in the following format:  $a \pm b(c)$ , where  $a$  represents the mean model fault detection accuracy,  $b$  represents one standard deviation of model fault detection accuracy and  $c$  represents the % of healthy target domain test samples identified as healthy (true negative rate). Values in **bold** font indicate the best performer based on model accuracy.

| Model              | Model fault detection accuracy (%) for different numbers of target domain fault samples available |                                                      |                                                      |                                                     |                                                      |                                                      |
|--------------------|---------------------------------------------------------------------------------------------------|------------------------------------------------------|------------------------------------------------------|-----------------------------------------------------|------------------------------------------------------|------------------------------------------------------|
|                    | 4                                                                                                 | 8                                                    | 16                                                   | 32                                                  | 64                                                   | 128                                                  |
| PCA                | 58.67 $\pm$ 0.75<br>(99.45)                                                                       | 58.61 $\pm$ 0.59<br>(99.67)                          | 58.46 $\pm$ 0.60<br>(99.65)                          | 58.73 $\pm$ 0.79<br>(98.89)                         | 58.72 $\pm$ 0.60<br>(99.56)                          | 59.41 $\pm$ 0.82<br>(98.81)                          |
| Auto-Encoder       | 57.19 $\pm$ 0.59<br>(99.51)                                                                       | 57.09 $\pm$ 0.66<br>(99.37)                          | 57.20 $\pm$ 0.47<br>(99.23)                          | 57.56 $\pm$ 0.46<br>(99.27)                         | 57.39 $\pm$ 0.69<br>(98.97)                          | 57.33 $\pm$ 0.58<br>(99.03)                          |
| Source             | 58.10 $\pm$ 0.31<br>(66.87)                                                                       | 58.05 $\pm$ 0.31<br>(66.97)                          | 58.13 $\pm$ 0.38<br>(67.07)                          | 58.13 $\pm$ 0.52<br>(67.04)                         | 57.94 $\pm$ 0.25<br>(66.62)                          | 58.00 $\pm$ 0.42<br>(66.86)                          |
| Target             | 51.04 $\pm$ 0.66<br>(97.82)                                                                       | 52.74 $\pm$ 2.34<br>(98.48)                          | 53.27 $\pm$ 1.13<br>(99.27)                          | 57.06 $\pm$ 3.95<br>(99.71)                         | 62.34 $\pm$ 5.13<br>(98.05)                          | 72.70 $\pm$ 6.14<br>(94.62)                          |
| Weight Fine-Tuning | 61.12 $\pm$ 1.36<br>(91.20)                                                                       | 63.87 $\pm$ 0.79<br>(91.41)                          | 65.10 $\pm$ 1.62<br>(91.57)                          | 67.54 $\pm$ 1.80<br>(93.69)                         | 71.13 $\pm$ 1.70<br>(94.93)                          | 74.76 $\pm$ 2.02<br>(95.77)                          |
| SDDA               | <b>64.02 <math>\pm</math> 2.38</b><br><b>(98.98)</b>                                              | <b>65.98 <math>\pm</math> 2.21</b><br><b>(98.76)</b> | <b>67.04 <math>\pm</math> 1.92</b><br><b>(96.42)</b> | <b>72.23 <math>\pm</math> 2.78</b><br><b>(97.4)</b> | <b>75.20 <math>\pm</math> 2.11</b><br><b>(97.23)</b> | <b>81.19 <math>\pm</math> 1.89</b><br><b>(95.19)</b> |

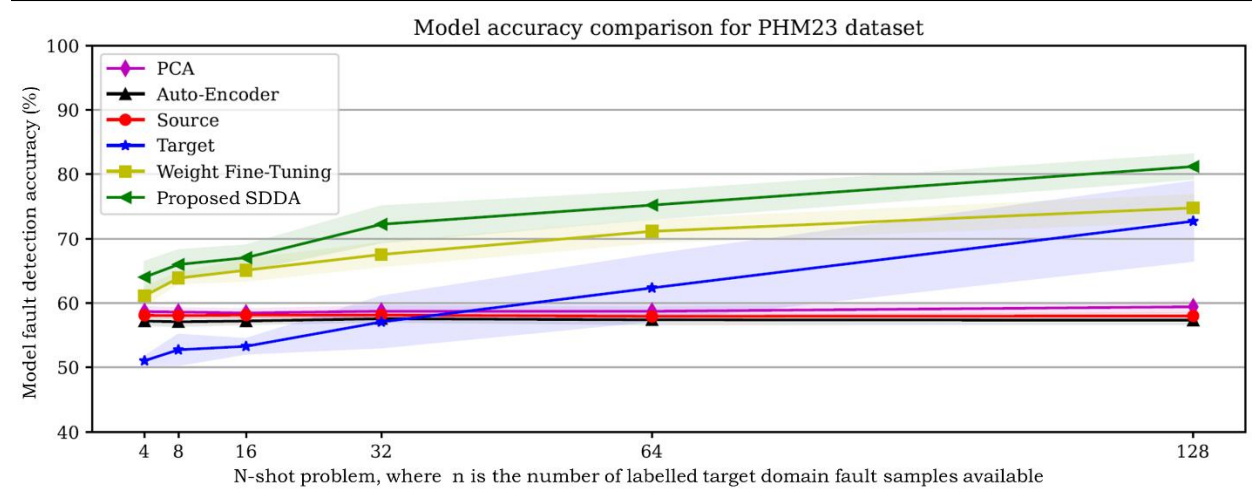


Fig. 10. Results of case study experiments. Solid lines indicate the 15-run mean results, with a one standard deviation band around them. Note that the PCA, auto-encoder and source methods had consistent results, leading to small standard deviation bands. These methods also have a fixed, low fault diagnosis accuracy. The target-only method, as expected, increases in performance as the number of target domain training faults is increased. WFT and DA outperform all other models at any number of available fault samples; however, DA consistently outperforms WFT.

Looking at the transfer learning methods, we observe that the WFT model outperformed the anomaly detectors as well as the target-only method, again indicating some overlap between the simulated source and real target domain. As the results from the comprehensive numerical investigations showed, if the source-only model contains some level of transferable knowledge, the WFT method may improve upon the target-only method. The SDDA model outperformed the WFT method, indicating that explicit feature-level domain alignment improved the transfer of knowledge from the simulated source domain to the real target domain in this case study. With only four target fault samples, or 0.28 % of possible permutations, SDDA correctly classified samples with approximately 65 % accuracy. As the target domain data increases, so too does the SDDA classification accuracy, with 128 samples, or 8.9 % of the possible permutations, leading to approximately 80 % accuracy.

All models except the source-only method had good true negative rates, indicating that they generally do not misclassify unseen healthy samples as faults. However, considering that the source-only method's accuracy is above 50 %, we can conclude that there is some degree of overlap between the simulated source and real target domains, such that a model only trained on source data can correctly identify both healthy and fault samples.

In conclusion, this case study evaluates SDDA under relatively well-aligned simulation-to-real transfer conditions. The PHM23 experiments illustrate that, even with minimal target domain fault data, as low as 0.28 % of the possible permutations, SDDA achieves higher fault detection accuracy than the source-only, target-only, weight fine-tuning, PCA, and auto-encoder baselines considered in this study. Importantly, SDDA maintains high true negative rates, which is important for real industrial settings where false alarms may cause unnecessary downtime. These findings indicate that, when simulation fidelity and sensor alignment are reasonably achievable, explicit domain alignment can improve few-shot simulation-driven fault assessment. The next case study explores a more challenging scenario, where the simulation model differs substantially from the target domain, thereby testing SDDA's robustness in high-discrepancy simulation-to-real conditions.

## 5 Case study 2: PHM09 dataset

While the first case study evaluated SDDA under relatively well-aligned simulation and real gearbox domains, real-world applications may present modelling challenges that involve large discrepancies between simulated and real systems. This second case study

evaluates SDDA under a high-discrepancy simulation-to-real scenario using the PHM09 dataset [50], where the simulated and real gearboxes differ in gearbox layout, sensor characteristics, and failure modes. This setup tests whether explicit feature-level domain alignment can improve few-shot fault detection when the simulated source domain is only partially representative of the real target system.

The PHM09 experiments were conducted on a multi-stage spur and helical gearbox. The spur version is used in this case study and has a 5:1 reduction ratio. A schematic of the setup is shown in Fig. 11. The PHM09 gearbox setup was run under eight different fault conditions, as seen in Table 7. These conditions are captured for two high and two low loads at various input shaft rotation speeds. This study only considers the 30 Hz rotation samples<sup>6</sup>. Therefore, this study only considers four data samples based on the four load cases at 30 Hz input speed. Each load case represents 120 input pinion rotations and 24 output shaft rotations per sample at the chosen 30 Hz motor speed.

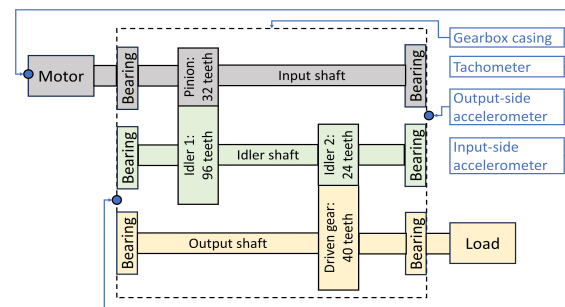


Fig. 11. Schematic of PHM09 gearbox. The gearbox has three shafts, namely an input, idler and output shaft. All shafts are fitted with spur gears. The input shaft has a pinion gear with 32 teeth, connecting it to the idler input side through a gear with 96 teeth. The idler output side contains a gear with 24 teeth, which connects it to the output shaft through a gear with 40 teeth. The gearbox has an input and output-side accelerometer which are mounted on the gearbox casing. The motor shaft is connected to a tachometer, which generates 10 pulses per revolution.

The signals were synchronously captured at 200 kHz, but due to three sensors being used, only equates to a sample frequency of 66.6 kHz per sensor. The three sensors consist of an input-side and output-side accelerometer and a pinion tachometer with 10 pulses per revolution.

<sup>6</sup> Further investigations into case study 1's results not shown in this work, indicated that the SDDA model's accuracy was insensitive to changes in gearbox speed. Therefore, simplifying this dataset to a single speed case was deemed a fair way to speed up simulations whilst simultaneously lowering the memory requirements.

Each data sample contains 266656 data points. As before for case study 1, we only utilise the

simulated data is run for a 30 Hz input motor and contains examples over five loads from low to high, namely 5, 10, 15, 20, and 25 Nm. A range of five loads

Table 7. Summary of PHM09 gearbox dataset classes. The faults are indicated in bold typeface to aid readability.

| Label | Gear           |         |                  |               | Bearing      |             |              |         | Shaft   |         |                  |               |
|-------|----------------|---------|------------------|---------------|--------------|-------------|--------------|---------|---------|---------|------------------|---------------|
|       | 32 T           | 96 T    | 48 T             | 80 T          | IN-IS        | IN-ID       | IN-OS        | OUT-IS  | OUT-ID  | OUT-OS  | Input            | Output        |
| 1     | Healthy        | Healthy | Healthy          | Healthy       | Healthy      | Healthy     | Healthy      | Healthy | Healthy | Healthy | Healthy          | Healthy       |
| 2     | <b>Chipped</b> | Healthy | Eccentric        | Healthy       | Healthy      | Healthy     | Healthy      | Healthy | Healthy | Healthy | Healthy          | Healthy       |
| 3     | Healthy        | Healthy | <b>Eccentric</b> | Healthy       | Healthy      | Healthy     | Healthy      | Healthy | Healthy | Healthy | Healthy          | Healthy       |
| 4     | Healthy        | Healthy | <b>Eccentric</b> | <b>Broken</b> | <b>Ball</b>  | Healthy     | Healthy      | Healthy | Healthy | Healthy | Healthy          | Healthy       |
| 5     | <b>Chipped</b> | Healthy | <b>Eccentric</b> | <b>Broken</b> | <b>Inner</b> | <b>Ball</b> | <b>Outer</b> | Healthy | Healthy | Healthy | Healthy          | Healthy       |
| 6     | Healthy        | Healthy | Healthy          | <b>Broken</b> | <b>Inner</b> | <b>Ball</b> | <b>Outer</b> | Healthy | Healthy | Healthy | <b>Imbalance</b> | Healthy       |
| 7     | Healthy        | Healthy | Healthy          | Healthy       | <b>Inner</b> | Healthy     | Healthy      | Healthy | Healthy | Healthy | Healthy          | <b>Keyway</b> |
| 8     | Healthy        | Healthy | Healthy          | Healthy       | Healthy      | <b>Ball</b> | <b>Outer</b> | Healthy | Healthy | Healthy | <b>Imbalance</b> | Healthy       |

T – Teeth; IN – Input side; OUT – Output side; IS – Input shaft; ID – Idler shaft; OS – Output shaft

pinion-side accelerometer data in this study due to memory restrictions.

The following sub-section describes how the source and target domains are constructed for the transfer learning tasks. This is followed by an explanation of the experiments that will be conducted. Finally, the results are shown and discussed.

## 5.1 Data-processing: Construction of the source and target domain

This study specifically focuses on the case where the simulated and real gearbox geometries differ. It is therefore convenient to utilise the existing ten DOF numerical gearbox outlined in [section 2](#) as the source domain model. This model represents a single-stage spur gearbox, which is different from the target domain dual-stage spur gearbox.

The source domain is simulated at 200 kHz using the single-stage ten DOF numerical gearbox model. The gearbox parameters are the same as those found in [Table A.1](#), except that the number of pinion and driven gear teeth are changed to 32 and 96 respectively, to match the first stage of the PHM09 gearbox<sup>7</sup>. The source data is generated under identical load and speed values as the target data. Simulated pinion-side gearbox casing  $y$ -vibrations (refer to [Fig. 2](#)) are captured over the same number of steady-state rotations as the target domain. The choice to use the casing  $y$ -vibrations is not trivial, as the numerical investigations revealed that the sensor location between the source and target domains is very important. Since the PHM09 setup measures vibrations on the casing, we simulate vibrations on the casing. The

was simulated because the underlying loads used in the source domain dataset are unknown and simply reported as "low" and "high" in the dataset description. Simulated samples are filtered and resampled to 66 kHz to match the target domain. This process is repeated for all five classes, namely healthy, crack, chip, spall, and pit faults.

The target domain was not time-synchronously averaged in this case study, due to the limited number of rotations available for training at each given load and fault case. Consequently, to compensate for the noise in the target domain, Gaussian noise at SNR = 0 dB is added to the simulated source domain data to force DDMs to learn robust fault features.

Both simulated and real datasets contain healthy and fault classes. The real dataset has eight distinct health states (seen in [Table 7](#)), whereas the numerical gearbox model has the five health states outlined above. Due to the discrepancy in health states, the transfer task for this case study is simplified from fault classification to fault detection, which allows us to relabel all fault states under a single label in the source and target domains.

At this stage, all source and target samples are similar in shape and can be processed together. For both domains, the 4 s measurements are reworked into datasets as follows:

The full 4 s measurement results in 266 656 points being captured. A window of 12000 points is taken and added to the database associated with each domain with the corresponding label from the 4 s signal. This window is slid forward in time by 2540 points, and the process is repeated. This results in 100 uniquely sampled windows per health state, per load. Take note that 12000 points capture exactly one rotation of the output shaft of the PHM09 gearbox, implying any faults on this shaft should

<sup>7</sup> In the early stages of this work, matching of the first stage was not done and this resulted in poor transfer results. Therefore, this step in the process is of paramount importance.

be visible in the captured signal. For both domains, signals are standardised for each load, using the mean and standard deviation of all healthy signals (at the given load) for the standardisation process.

For the source domain, with 5 load cases, 5 classes, and 100 windows, 2500 unique samples are obtained. For the target domain, with 4 load cases, 8 classes, and 100 windows, 3200 unique samples are obtained. A subset of these datasets is used for training, with the remainder being used for testing. The breakdown of the data is seen in [Table 8](#).

## 5.2 Case study setup

This case study utilises the time domain signals for fault detection. Examples from the healthy and fault classes for each domain are shown in [Fig. 12](#). The same six DDMs that were evaluated in case study 1 are reused in this study. The only difference is that the source, target, WFT and SDDA models use the CNN structure outlined

in [section 2](#). The PCA and AE models are identical to the previous case study, except that this case study uses a newly optimised 300 components for reconstruction.

The goal of the models is to correctly identify faults in the target domain from limited time-domain signals in the target training set. Experiments are run for the case where 0, 1, 2, 4, 8, and 16 faults are available in the target domain training set. Due to the low number of available target domain faults, a validation set is not used for training convergence. Rather, the training loss is used to gauge training convergence. After initial testing, 150 epochs were seen to result in model training losses that converged. Faults are chosen at random from the available 7 fault classes indicated in [Table 7](#). In all cases, 160 unique healthy samples are made available. The remaining target domain samples are used as a test set. The experiments are run five times over to gather statistically significant results. A summary of the parameters used in this case study is seen in [Table 9](#).

Table 8. Breakdown of the training and test data for case study 2. The → symbol represents samples that are duplicated to increase the number of samples for a given dataset. The source domain has 2500 unique samples distributed across all training/testing sets, and the target domain has 3200 unique samples distributed across all training/testing sets.

| Dataset                         | Number of healthy training samples | Number of fault training samples | Number of healthy testing samples | Number of fault testing samples |
|---------------------------------|------------------------------------|----------------------------------|-----------------------------------|---------------------------------|
| Source (Simulated data)         | 500 → 2600                         | 2000 → 2600                      | n.a.                              | n.a.                            |
| Target (0 PHM09 fault samples)  | 160 → 2600                         | 0                                | 240                               | 2800                            |
| Target (1 PHM09 fault samples)  | 160 → 2600                         | 1 → 2600                         | 240                               | 2799                            |
| Target (2 PHM09 fault samples)  | 160 → 2600                         | 2 → 2600                         | 240                               | 2798                            |
| Target (4 PHM09 fault samples)  | 160 → 2600                         | 4 → 2600                         | 240                               | 2796                            |
| Target (8 PHM09 fault samples)  | 160 → 2600                         | 8 → 2600                         | 240                               | 2792                            |
| Target (16 PHM09 fault samples) | 160 → 2600                         | 16 → 2600                        | 240                               | 2784                            |

Table 9. Summary of the parameters used in case study 2. Here, the columns represent parameters for the source-only, target-only, principal component analysis, autoencoder, weight fine-tuning and simulation-driven domain adaptation methods.

| Model         | Source | Target | PCA                                                                   | AE    | WFT            | SDDA  |
|---------------|--------|--------|-----------------------------------------------------------------------|-------|----------------|-------|
| Optimiser     |        |        | Adam                                                                  |       |                |       |
| Learning rate | 0.001  | 0.001  | N.A.                                                                  | 0.001 | 0.001 → 0.0001 | 0.001 |
| Training data | Source | Target | Target (Healthy only)                                                 |       |                |       |
| Data type     |        |        | Time domain: Casing y                                                 |       |                |       |
| Data format   |        |        | 12000 Data points equating to 1 output shaft rotation                 |       |                |       |
| Fault type(s) |        |        | Source: Crack, chip, pit, spall   Target: See <a href="#">Table 7</a> |       |                |       |
| Load cases    |        |        | Source: 5, 10, 15, 20, 25 Nm   Target: 2 Low, 2 High                  |       |                |       |
| Speed cases   |        |        | 30 Hz motor input                                                     |       |                |       |

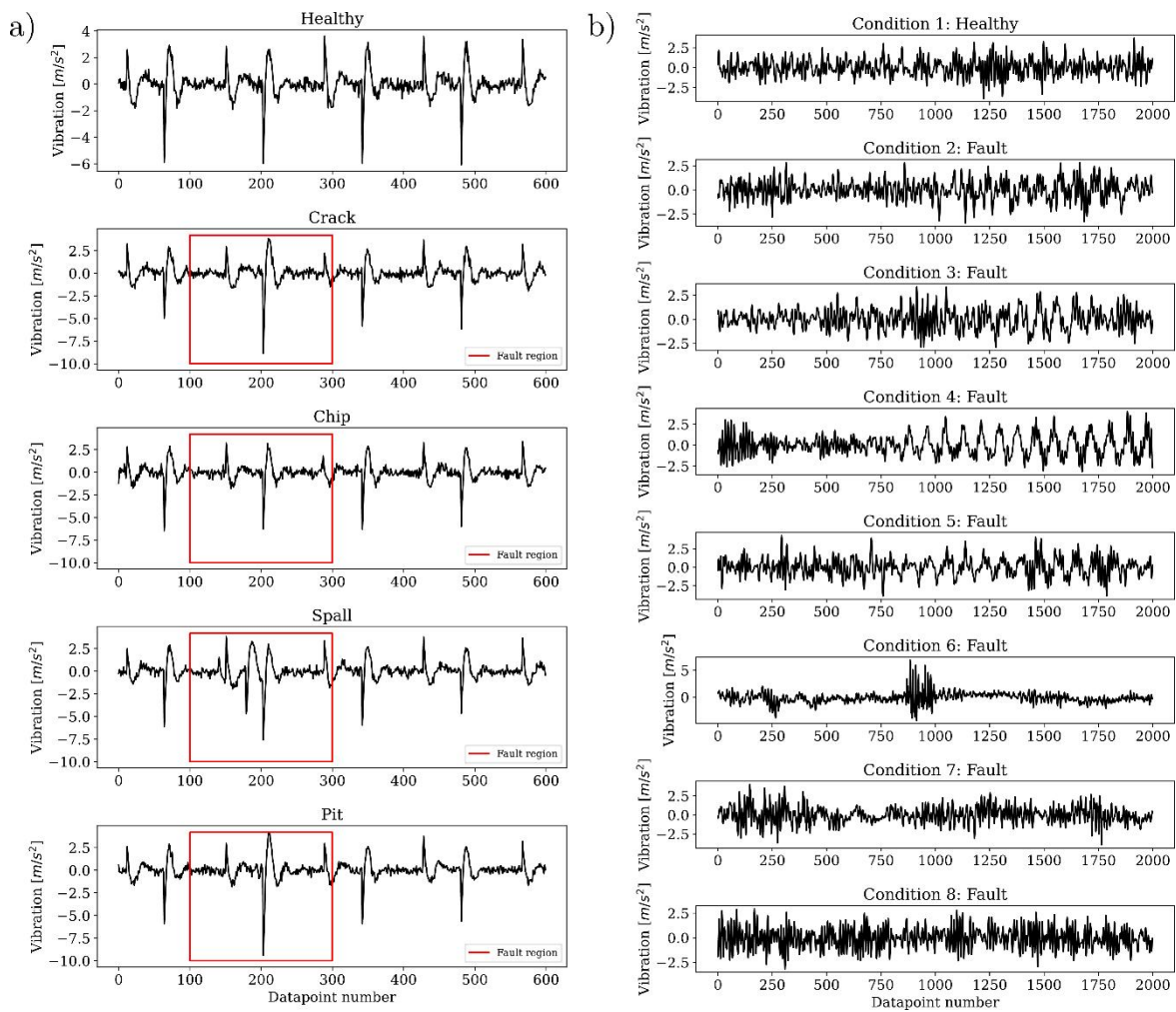


Fig. 12. a) and b) represent examples of signals used in the PHM09 case study. a) Simulated gearbox signals used for the source domain training. The boxes show the impact of various faults on the time-domain signal. b) Real measured signals on the PHM09 gearbox used in the target domain.

### 5.3 Results

The results across the different models, for different numbers of available target domain faults, are seen in Fig. 13. The performance of the SDDA method under this few-shot mismatched simulation-to-real study is best interpreted in relation to the other models.

The source-only model had 50 % fault detection accuracy across all experiments. Considering that the true negative rate was 0 %<sup>8</sup>, this implies that the source-only model classified all unseen test samples as faults. This result shows that the degree of misalignment between the source and target domains is severe. The target-only model had increasing fault detection performance with increased fault training samples, as

one would expect. However, this performance was quite low and a target-only approach will not work for this few-shot problem. The WFT fault detection accuracies are very similar to the target-only method, indicating that no extra knowledge was gained from the vastly misaligned source domain. This behaviour was also seen in the comprehensive numerical investigations when the transfer case was difficult (see Fig. 7d)).

The SDDA approach showed improved performance in comparison with the other models. With one fault example available, roughly 60 % fault detection accuracy was achieved, increasing to almost 90 % with only 16 fault examples available. This indicates that explicit feature-level domain alignment can extract useful information from the simulated source domain, even when direct source-only transfer and weight fine-tuning are ineffective. The zero-shot case, where no faults are available in the target domain, proved to be too challenging for the SDDA approach. No model could

<sup>8</sup> The true negative rates (% of healthy samples identified as healthy) for all models across all available fault samples were 100 %, except the source-only method, which achieved a 0 % true negative rate over all available fault samples.

identify faults when no faults were previously present in the target domain. This failure is expected because the present SDDA formulation aligns feature distributions but still relies on labelled target-domain fault examples to anchor the target fault region in the learned feature space. In the zero-shot case, only healthy target-domain labels are available; consequently, the model has no reference for where real target-domain fault samples should lie after alignment. Source-domain fault features may therefore be aligned with target-domain regions that do not correspond to separable real fault behaviour, especially when the simulated and real fault mechanisms differ. This suggests that zero-shot simulation-to-real transfer requires additional constraints beyond the present MK-MMD objective. One promising direction is to embed physics more directly into the data-driven model or adaptation objective, for example through physics-informed labels, loss terms, or constraints that encourage physically meaningful feature alignment. Jia et al. [51] provide a related example in cross-machine bearing fault diagnosis, in which physical information is incorporated into an unsupervised domain adaptation framework to improve diagnostic reliability and interpretability.

The AE and PCA models were found to be ineffective on this dataset, achieving only 50 % accuracy. Upon further investigation, it was observed that fault examples had reconstruction losses similar to those of the healthy samples. Therefore, all fault samples were classified as healthy samples.

The poor fault detection results from the anomaly detection methods indicate that insufficient knowledge is encoded in the target health samples. The source-only results showcase that a large discrepancy is present between the physics-based and real gearbox models. However, the SDDA method can successfully utilise source domain information to significantly increase fault detection accuracy over other methods.

In conclusion, this case study evaluates SDDA under significant domain mismatch. Unlike weight fine-tuning and non-transfer baselines, which failed to leverage useful knowledge from the simulated domain in this case, SDDA demonstrated meaningful few-shot fault detection performance even when the alignment between simulated and real gearbox data was poor. These results are consistent with the earlier numerical investigation: explicit domain adaptation can enable the transfer of useful latent fault representations across dissimilar systems, provided that at least a small number of target-domain fault examples are available. However, the zero-shot setting remains unresolved and requires further investigation.

Together with the findings from Case Study 1, these results indicate that SDDA is a practical candidate workflow for simulation-driven few-shot gearbox fault assessment when historical fault data are scarce and perfect simulation alignment is not achievable.

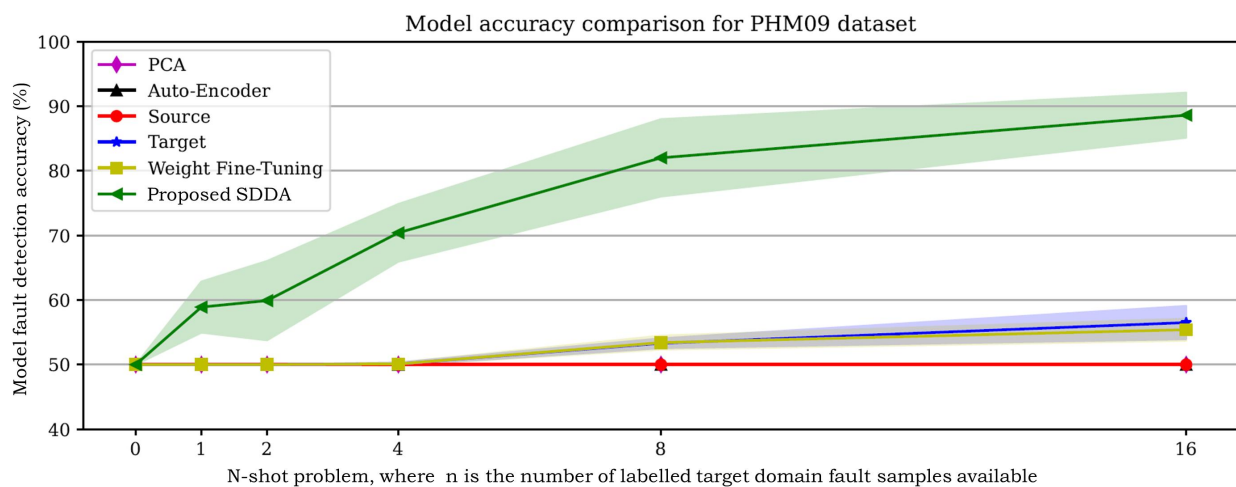


Fig. 13. Fault detection accuracy for different models with different numbers of labelled target domain faults. The target-only and source-only models represent methods which are not calibrated. Weight fine-tuning and the SDDA approaches represent the transfer learning methods. Finally, anomaly detection methods are represented by PCA and an autoencoder. Both these models utilise a bottleneck vector size of 300.

## 6 Conclusions and future work

This paper evaluated a simulation-driven domain adaptation (SDDA) workflow for end-to-end, few-shot gearbox fault assessment. The work focused on simulation-to-real scenarios where very limited labelled fault data are available in the target domain, which represents a realistic and challenging condition in industrial gearbox monitoring. This study examined how established components, namely physics-based simulated source data, a CNN-based data-driven model, and MK-MMD-based feature alignment, can be combined and evaluated for few-shot gearbox fault assessment. In this setting, SDDA addresses the scarcity of target-domain fault data by leveraging simulated source-domain data and explicitly reducing the feature-level discrepancy between simulated and real domains.

A central contribution of this paper is the controlled numerical investigation of factors that may cause simulation-to-real discrepancies, including mismatched gearbox modelling parameters, sensor placement, sensor noise, and physics-based modelling fidelity. These investigations provided a structured environment in which transfer performance could be evaluated for several models under isolated domain misalignment scenarios. The results showed that sensor placement, physics-based modelling accuracy, and extreme sensor noise were among the strongest causes of domain misalignment and poor fault diagnosis performance. Despite these large domain misalignments, SDDA remained robust in the numerical study, achieving flawless fault diagnosis accuracy in all scenarios except the extreme sensor noise case.

Two simulation-to-real fault detection case studies were then conducted to evaluate SDDA under real-world conditions. The first case study, using the PHM23 dataset, represented a relatively well-aligned simulation-to-real scenario in which the physics-based gearbox model was configured to approximate the real gearbox. In this setting, SDDA outperformed the source-only, target-only, weight fine-tuning, PCA, and auto-encoder baselines considered in this study, achieving improved fault detection accuracy even with minimal labelled target-domain fault data. The second case study, based on the PHM09 dataset, represented a more challenging scenario where the simulated and real systems were substantially mismatched. In this high-discrepancy setting, SDDA remained the only method considered that delivered meaningful few-shot fault detection performance. These results indicate that simulated data, when coupled with explicit domain adaptation, can provide useful information for gearbox fault assessment even when the simulated and real domains are not perfectly aligned.

The broader implication of these findings is that physics-informed simulated data can serve as more than a supplementary source of training data for gearbox fault assessment, provided that domain discrepancies are explicitly addressed. The results suggest that feature-level domain alignment can improve few-shot learning performance by allowing useful fault-related representations to be transferred from simulated source data to scarce real target data. This is particularly relevant for industrial condition monitoring applications, where real fault data are rare, costly, and often impractical to collect in sufficient quantities.

Several open questions remain. First, future work should explore the trade-off between simulation fidelity and SDDA performance. It is not presently clear whether relatively simple phenomenological gearbox models are sufficient, or whether higher-fidelity simulations would unlock further improvements in fault assessment performance. This question is especially important for unseen or evolving gear faults. Lower-fidelity models may be sufficient when the objective is to reproduce dominant vibration signatures associated with known fault types, such as impulsive responses, modulation patterns, and stiffness-related sidebands. However, faults that initiate, propagate, or interact with other components may require higher-fidelity models that capture contact mechanics, crack or pitting evolution, load-dependent stiffness changes, casing transfer paths, and multi-component dynamics. Future work should therefore evaluate not only whether higher-fidelity simulations improve classification accuracy, but also whether the added physical detail improves transfer to fault classes or degradation stages that are not explicitly labelled in the target domain. Second, the zero-shot case, where no fault labels exist in the target domain, remains unresolved and deserves focused investigation. Future work should examine whether embedding physics more directly into the data-driven model, for example, through physics-informed labels, loss terms, or physically constrained adaptation objectives, can provide the missing target fault structure required for zero-shot simulation-to-real transfer. Third, future studies should investigate whether the SDDA workflow can be extended to unseen or evolving fault classes, and whether such extensions require higher-fidelity fault-evolution models or can be achieved with lower-fidelity models combined with stronger physics-informed learning constraints. Fourth, additional experimental datasets and industrial case studies are needed to further assess the generality of the observed trends. Finally, future work should include a broader benchmark of domain adaptation mechanisms to determine whether alternative mechanisms provide measurable advantages over MK-MMD under varying levels of label scarcity and simulation-to-real domain shift.

Overall, this study provides evidence that simulation-driven domain adaptation is a promising candidate workflow for few-shot gearbox fault assessment. The results show that meaningful gearbox fault diagnosis and detection can be achieved in sparse-data regimes through the strategic use of physics-based simulation and explicit domain alignment. These findings support further development of data-efficient condition monitoring systems for gearboxes and other industrial machines where labelled fault data are rare and costly to obtain.

## 7 Conflict of interest

The authors declare that they have no conflicts of interest related to this work.

## References

- [1] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mechanical Systems and Signal Processing*, vol. 138, p. 106587, 2020.
- [2] T. Han, C. Liu, W. Yang, and D. Jiang, "A novel adversarial learning framework in deep convolutional neural network for intelligent diagnosis of mechanical faults," *Knowledge-Based Systems*, vol. 165, pp. 474-487, 2019.
- [3] T. Han, C. Liu, W. Yang, and D. Jiang, "Learning transferable features in deep convolutional neural networks for diagnosing unseen machine conditions," *ISA Transactions*, vol. 93, pp. 341-353, 2019.
- [4] M. Zhang, W. Lu, J. Yang, D. Wang, and L. Bin, "Domain adaptation with multilayer adversarial learning for fault diagnosis of gearbox under multiple operating conditions," in *2019 Prognostics and System Health Management Conference (PHM-Qingdao)*, pp. 1-6, 2019.
- [5] C. Qiu, T. Tang, T. Yang, and M. Chen, "Learning to generalize with latent embedding optimization for few- and zero-shot cross domain fault diagnosis," *Expert Systems with Applications*, vol. 254, p. 124280, 2024.
- [6] X. Li, Y. Hu, M. Li, and J. Zheng, "Fault diagnostics between different type of components: A transfer learning approach," *Applied Soft Computing*, vol. 86, p. 105950, 2020.
- [7] C. Lin, Y. Kong, Q. Han, T. Wang, M. Dong, H. Liu, and F. Chu, "An information fusion-based meta transfer learning method for few-shot fault diagnosis under varying operating conditions," *Mechanical Systems and Signal Processing*, vol. 220, p. 111652, 2024.
- [8] J. M. Ha and O. Fink, "Domain knowledge-informed synthetic fault sample generation with health data map for cross-domain planetary gearbox fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 202, p. 110680, 2023.
- [9] D. She, Z. Yang, Y. Duan, X. Yan, J. Chen, and Y. Li, "A meta transfer learning method for gearbox fault diagnosis with limited data," *Measurement Science and Technology*, vol. 35, no. 8, p. 086114, 2024.
- [10] D. She, Z. Yang, Y. Duan, and M. G. Pecht, "A meta transfer learning-driven few-shot fault diagnosis method for combine harvester gearboxes," *Computers and Electronics in Agriculture*, vol. 227, p. 109605, 2024.
- [11] L. Chen, G. Feng, D. Zhen, H. Jing, X. Liang, and F. Gu, "Cross-condition fault diagnosis of planetary gearboxes driven by data-model fusion based on improved domain-adversarial transfer learning," *Shock and Vibration*, vol. 2025, no. 1, p. 9982177, 2025.
- [12] Z. Yang, Y. Duan, D. She, and M. G. Pecht, "A meta transfer learning fault diagnosis method for gearbox with few-shot data," *Measurement Science and Technology*, vol. 36, no. 2, p. 025007, 2025.
- [13] X. Li, H. Jiang, K. Zhao, and R. Wang, "A deep transfer nonnegativity-constraint sparse autoencoder for rolling bearing fault diagnosis with few labeled data," *IEEE Access*, vol. 7, pp. 91216-91224, 2019.
- [14] H. Shao, Z. Ding, J. Cheng, and H. Jiang, "Intelligent fault diagnosis among different rotating machines using novel stacked transfer auto-encoder optimized by PSO," *ISA Transactions*, vol. 105, pp. 308-319, 2020.
- [15] Z. Li, Z. Cao, K. Luo, and H. Fu, "A novel method for fault diagnosis of rolling bearings based on domain-adversarial partial transfer learning," in *2020 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan)*, pp. 414-419, 2020.
- [16] K. Hu, Y. Cheng, J. Wu, and H. Zhu, "Cross-domain intelligent diagnostics for rotating machinery using domain adaptive and adversarial networks," *Journal of Industrial Information Integration*, vol. 42, p. 100722, 2024.
- [17] C. Sobie, C. Freitas, and M. Nicolai, "Simulation-driven machine learning: Bearing fault classification," *Mechanical Systems and Signal Processing*, vol. 99, pp. 403-419, 2018.

- [18] C. Liu, A. Mauricio, J. Qi, D. Peng, and K. Gryllias, "Domain adaptation digital twin for rolling element bearing prognostics," Annual Conference of the PHM Society, vol. 12, no. 1, pp. 1-10, 2020.
- [19] C. Liu and K. Gryllias, "Simulation-driven domain adaptation for rolling element bearing fault diagnosis," IEEE Transactions on Industrial Informatics, vol. 18, no. 9, pp. 5760-5770, 2022.
- [20] D. Li, Y. Zhao, and Y. Zhao, "A dynamic-model-based fault diagnosis method for a wind turbine planetary gearbox using a deep learning network," Protection and Control of Modern Power Systems, vol. 7, no. 1, p. 22, 2022.
- [21] Z. Xiong, M. Li, Y. Tang, S. Xiao, and M. Song, "Research on fault diagnosis method of deep transfer learning driven by simulation data," Vibroengineering Procedia, vol. 43, pp. 21-26, 2022.
- [22] M.-M. Song, Z.-C. Xiong, J.-H. Zhong, S.-G. Xiao, and Y.-H. Tang, "Research on fault diagnosis method of planetary gearbox based on dynamic simulation and deep transfer learning," Scientific Reports, vol. 12, no. 1, p. 17023, 2022.
- [23] M. Song, Z. Xiong, J. Zhong, S. Xiao, and J. Ren, "Fault diagnosis of planetary gearbox based on dynamic simulation and partial transfer learning," Biomimetics, vol. 8, no. 4, p. 361, 2023.
- [24] P. Zhu, L. Deng, B. Tang, Q. Yang, and Q. Li, "Digital twin-enabled entropy regularized wavelet attention domain adaptation network for gearboxes fault diagnosis without fault data," Advanced Engineering Informatics, vol. 64, p. 103055, 2025.
- [25] F. Jiang, W. Lin, Z. Wu, S. Zhang, Z. Chen, and W. Li, "Fault diagnosis of gearbox driven by vibration response mechanism and enhanced unsupervised domain adaptation," Advanced Engineering Informatics, vol. 61, p. 102460, 2024.
- [26] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," ACM Computing Surveys, vol. 53, no. 3, Art. no. 63, pp. 1-34, 2020.
- [27] O. Matania, L. Bachar, V. Khemani, D. Das, M. H. Azarian, and J. Bortman, "One-fault-shot learning for fault severity estimation of gears that addresses differences between simulation and experimental signals and transfer function effects," Advanced Engineering Informatics, vol. 56, p. 101945, 2023.
- [28] O. D. Mohammed and M. Rantatalo, "Gear fault models and dynamics-based modelling for gear fault detection – A review," Engineering Failure Analysis, vol. 117, p. 104798, 2020.
- [29] L. van Eyk, "A hybrid gearbox condition monitoring methodology using transfer learning for calibration," M.Eng. dissertation, University of Pretoria, Pretoria, South Africa, 2021.
- [30] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, The finite element method: Its basis and fundamentals, 7th ed. Oxford, UK: Butterworth-Heinemann, 2013.
- [31] D. C. H. Yang and J. Y. Lin, "Hertzian damping, tooth friction and bending elasticity in gear impact dynamics," Journal of Mechanisms, Transmissions, and Automation in Design, vol. 109, no. 2, pp. 189-196, 1987.
- [32] L. van Eyk, P. S. Heyns, and S. Schmidt, "A generalised stiffness model for incorporating multiple spur gear tooth faults - part 1: Stiffness modelling," in Modelling and simulation of complex systems for sustainable energy efficiency II, Applied Condition Monitoring, vol. 23. Cham, Switzerland: Springer, 2025, pp. 200-215.
- [33] L. van Eyk, P. S. Heyns, and S. Schmidt, "A short review of gear fault modelling in a hybrid modelling context," in Modelling and simulation of complex systems for sustainable energy efficiency, Applied Condition Monitoring, vol. 20. Cham, Switzerland: Springer, 2022, pp. 242-258.
- [34] Z. Chen and Y. Shao, "Dynamic simulation of spur gear with tooth root crack propagating along tooth width and crack depth," Engineering Failure Analysis, vol. 18, no. 8, pp. 2149-2164, 2011.
- [35] X. Tian, "Dynamic simulation for system response of gearbox including localized gear faults," Master's thesis, University of Alberta, Edmonton, Canada, 2004.
- [36] F. Chaari, W. Baccar, M. S. Abbes, and M. Haddar, "Effect of spalling or tooth breakage on gearmesh stiffness and dynamic response of a one-stage spur gear transmission," European Journal of Mechanics - A/Solids, vol. 27, no. 4, pp. 691-705, 2008.
- [37] Z. Meng, F. Wang, and G. Shi, "A novel evolution model of pitting failure and effect on time-varying meshing stiffness of spur gears," Engineering Failure Analysis, vol. 120, p. 105068, 2021.

- [38] L. van Eyk, P. S. Heyns, and S. Schmidt, "A generalised stiffness model for incorporating multiple spur gear tooth faults - part 2: Fault modelling," in *Modelling and simulation of complex systems for sustainable energy efficiency II*, Applied Condition Monitoring, vol. 23. Cham, Switzerland: Springer, 2025, pp. 216-228.
- [39] F. Jia, Y. Lei, N. Lu, and S. Xing, "Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization," *Mechanical Systems and Signal Processing*, vol. 110, pp. 349-367, 2018.
- [40] Z. Chen, G. He, J. Li, Y. Liao, K. Gryllias, and W. Li, "Domain adversarial transfer network for cross-domain fault diagnosis of rotary machinery," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 8702-8712, 2020.
- [41] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1-35, 2016.
- [42] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv:1412.3474*, 2014.
- [43] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32nd International Conference on Machine Learning*, *Proceedings of Machine Learning Research*, vol. 37, pp. 97-105, 2015.
- [44] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Advances in Neural Information Processing Systems*, vol. 19, pp. 513-520, 2007.
- [45] H. Zheng, R. Wang, Y. Yang, J. Yin, Y. Li, Y. Li, and M. Xu, "Cross-domain fault diagnosis using knowledge transfer strategy: A review," *IEEE Access*, vol. 7, pp. 129260-129290, 2019.
- [46] Y. Luo, N. Baddour, and M. Liang, "Dynamical modeling and experimental validation for tooth pitting and spalling in spur gears," *Mechanical Systems and Signal Processing*, vol. 119, pp. 155-181, 2019.
- [47] Z. Meng, G. Shi, and F. Wang, "Vibration response and fault characteristics analysis of gear based on time-varying mesh stiffness," *Mechanism and Machine Theory*, vol. 148, p. 103786, 2020.
- [48] PHM Society, "PHM North America 2023 conference data challenge," 2023. [Online]. Available: <https://data.phmsociety.org/phm2023-conference-data-challenge/>.
- [49] X. H. Liang, Z. L. Liu, J. Pan, and M. J. Zuo, "Spur gear tooth pitting propagation assessment using model-based analysis," *Chinese Journal of Mechanical Engineering*, vol. 30, no. 6, pp. 1369-1382, 2017.
- [50] PHM Society, "2009 PHM challenge competition data set," 2009. [Online]. Available: <https://phmsociety.org/public-data-sets/>.
- [51] N. Jia, W. Huang, C. Ding, J. Wang, and Z. Zhu, "Physics-informed unsupervised domain adaptation framework for cross-machine bearing fault diagnosis," *Advanced Engineering Informatics*, vol. 62, p. 102774, 2024.
- [52] F. Chaari, W. Bartelmus, R. Zimroz, T. Fakhfakh, and M. Haddar, "Gearbox vibration signal amplitude and frequency modulation," *Shock and Vibration*, vol. 19, no. 4, pp. 635-652, 2012.
- [53] O. D. Mohammed, M. Rantatalo, and J. O. Aidanpää, "Dynamic modelling of a one-stage spur gear system and vibration-based tooth crack detection analysis," *Mechanical Systems and Signal Processing*, vol. 54-55, pp. 293-305, 2015.

## Appendix A Additional information for the physical modelling process

The following sub-sections represent the necessary modelling quantities to reproduce the simulated gearbox signals from this work. Although this work does not contain a comprehensive write-up of the physics-based modelling process, a complete version is given in [32,38].

### A.1 Dynamic Matrices

Below are the relevant matrices that are solved for the dynamic simulations. All matrices or vectors are given for the ten DOF model. For the eight DOF model, simply remove rows and columns 9 and 10 (indices start at 1). For the six DOF model, additionally, remove rows and columns 3 and 8.

Firstly, the displacement vector for the ten DOF model, as illustrated in Fig. 2, is defined by:

$$\mathbf{u} = [x_1 \ y_1 \ \theta_{1,1} \ \theta_{1,2} \ x_2 \ y_2 \ \theta_{2,1} \ \theta_{2,2} \ x_f \ y_f] \quad (\text{A.1})$$

The velocity and acceleration vectors follow the same order as above, but are derived in time. Together, the displacement, velocity and acceleration vectors are solved using an applicable solver.

The mass matrix is defined as a diagonal matrix, with the mass ( $m$ ) and inertial ( $I$ ) quantities defined as illustrated in Fig. 2:

$$\mathbf{M} = \text{diag}[m_1 \ m_1 \ I_{1,1} \ I_{1,2} \ m_2 \ m_2 \ I_{2,1} \ I_{2,2} \ m_f \ m_f] \quad (\text{A.2})$$

The constant stiffness matrix is defined as follows, with the values of the entries defined in Table A.1 from Appendix A.2:

$$= \begin{bmatrix} k_{x1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -k_{x1} & 0 \\ 0 & k_{y1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -k_{y1} \\ 0 & 0 & k_{\theta 1} & -k_{\theta 1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -k_{\theta 1} & k_{\theta 1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k_{x2} & 0 & 0 & 0 & -k_{x2} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{y2} & 0 & 0 & 0 & -k_{y2} \\ 0 & 0 & 0 & 0 & 0 & 0 & k_{\theta 2} & -k_{\theta 2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -k_{\theta 2} & k_{\theta 2} & 0 & 0 \\ -k_{x1} & 0 & 0 & 0 & -k_{x2} & 0 & 0 & 0 & k_{xf} + k_{x1} + k_{x2} & 0 \\ 0 & -k_{y1} & 0 & 0 & 0 & -k_{y2} & 0 & 0 & 0 & k_{yf} + k_{y1} + k_{y2} \end{bmatrix} \quad (\text{A.3})$$

To construct the varying stiffness matrix, a projection matrix ( $\mathbf{P}$ ) is used. The projection matrix is necessary to

account for the pressure angle ( $\alpha_0$  in Appendix A.2) between the two gears:

$$\mathbf{P} = [\sin\alpha_0 \ \cos\alpha_0 \ 0 \ R_{b1} \ -\sin\alpha_0 \ -\cos\alpha_0 \ R_{b2} \ 0 \ 0 \ 0] \quad (\text{A.4})$$

Paired with the displacement values, the varying stiffness matrix can be found from

$$\mathbf{K}_v = k_t(t)\mathbf{u}\mathbf{P}^T\mathbf{P} \quad (\text{A.5})$$

where  $k_t(t)$  varies over time depending on the gear rotation angle. The result gives an  $N \times N$  matrix which is added to  $\mathbf{K}_s$  to form the final time-varying stiffness

term,  $\mathbf{K}$ . The  $k_t(t)$  value is found after faults are applied and stiffnesses are solved, based on the current gear rotation angle ( $\theta_{1,2}$  in Fig. 2).

The damping matrices are also constructed from constant and varying components. The constant matrix is found from:

$$\mathbf{C}_c = \begin{bmatrix} c_{x1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_{x1} & 0 \\ 0 & c_{y1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_{y1} \\ 0 & 0 & c_{\theta 1} & -c_{\theta 1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -c_{\theta 1} & c_{\theta 1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{x2} & 0 & 0 & 0 & -c_{x2} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{y2} & 0 & 0 & 0 & -c_{y2} \\ 0 & 0 & 0 & 0 & 0 & 0 & c_{\theta 2} & -c_{\theta 2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -c_{\theta 2} & c_{\theta 2} & 0 & 0 \\ -c_{x1} & 0 & 0 & 0 & -c_{x2} & 0 & 0 & 0 & c_{xf} + c_{x1} + c_{x2} & 0 \\ 0 & -c_{y1} & 0 & 0 & 0 & -c_{y2} & 0 & 0 & 0 & c_{yf} + c_{y1} + c_{y2} \end{bmatrix} \quad (\text{A.6})$$

The quantities in this matrix are outlined in Table A.1 from Appendix A.2.

$$\mathbf{C}_v = c_t(t)\mathbf{u}\mathbf{P}^T\mathbf{P} \quad (\text{A.7})$$

where  $c_t(t)$  is defined proportional to the stiffness:

The varying damping matrix is constructed with the same projection matrix as for  $\mathbf{K}_v$ :

$$c_t(t) = \frac{c_m}{k_m}k_t(t) \quad (\text{A.8})$$

The mean meshing stiffness ( $k_m$ ) is calculated by evaluating a full gear rotation

$$k_m = \frac{1}{2\pi} \int_0^{2\pi} k_t(\theta_{1,2}) d\theta_{1,2} \quad (\text{A.9})$$

and  $c_m$  is defined by

$$c_m = 2\zeta \sqrt{k_m \frac{m_1 m_2}{m_1 + m_2}} \quad (\text{A.10})$$

where  $\zeta = 0.07$  for this work and  $m_1$  and  $m_2$  are found in Table A.1 from Appendix A.2.

Finally, the force vector is set up as follows:

$$\mathbf{f} = [0 \quad 0 \quad T_m \quad 0 \quad 0 \quad 0 \quad 0 \quad T_l \quad 0 \quad 0] \quad (\text{A.11})$$

The load and motor torque values are seen in Table A.1 from Appendix A.2.

## A.2 Gearbox parameters

The quantities necessary to fully define the dynamic model, along with stiffness modelling quantities are given in Table A.1. The fault-related modelling quantities are given in Table A.2.

Table A.1. Parameters utilised in this work for Machine A and Machine B. Machine B only has eight DOFs, and therefore some quantities are not applicable, labelled as n.a. Machine A's parameters are based on those found in Luo et al. [46] and Machine B's parameters are based on those found in Meng et al. [47].

| Symbol                           | Description                                  | Machine A                  | Machine B                  | Units             |
|----------------------------------|----------------------------------------------|----------------------------|----------------------------|-------------------|
| $Z_1$                            | Number of pinion gear teeth                  | 16                         | 16                         | –                 |
| $Z_2$                            | Number of driven gear teeth                  | 48                         | 48                         | –                 |
| $m_1, m_2$                       | Tooth module                                 | 3.175                      | 2                          | mm                |
| $L_1, L_2$                       | Tooth width                                  | 16                         | 20                         | mm                |
| $\alpha_0$                       | Pressure angle                               | 20                         | 20                         | °                 |
| $E$                              | Young's modulus                              | 200                        | 200                        | GPa               |
| $\nu$                            | Poisson's ratio                              | 0.3                        | 0.3                        | –                 |
| $R_h$                            | Hub radius                                   | 9.76                       | 9.76                       | mm                |
| $m_1$                            | Mass of pinion block                         | 1.272                      | 0.3083                     | kg                |
| $m_2$                            | Mass of gear block                           | 3.5367                     | 0.4439                     | kg                |
| $m_f$                            | Mass of gearbox casing                       | 18.509                     | n.a.                       | kg                |
| $I_{1,1}$                        | Mass moment of inertia of motor <sup>9</sup> | 0.016107                   | 0.0043                     | kg.m <sup>2</sup> |
| $I_{1,2}$                        | Mass moment of inertia of pinion block       | 0.0001751                  | $0.96 \times 10^{-4}$      | kg.m <sup>2</sup> |
| $I_{2,1}$                        | Mass moment of inertia of gear block         | 0.006828                   | $2 \times 10^{-4}$         | kg.m <sup>2</sup> |
| $I_{2,2}$                        | Mass moment of inertia of load <sup>10</sup> | 0.005153                   | 0.0045                     | kg.m <sup>2</sup> |
| $T_m$                            | Motor torque                                 | $T_l \times \frac{16}{48}$ | $T_l \times \frac{16}{48}$ | N.m               |
| $T_l$                            | Load torque                                  | 5/10/15/20/25              | 5/10/15/20/25              | N.m               |
| $k_{x1}, k_{x2}, k_{y1}, k_{y2}$ | Bearing stiffnesses                          | $8.5364 \times 10^7$       | $6.56 \times 10^8$         | N/m               |
| $k_c$                            | Coupling stiffness <sup>11</sup>             | 330                        | $1 \times 10^4$            | N.m/rad           |
| $k_{xf}$                         | Gearbox casing x-stiffness                   | $1.9912 \times 10^8$       | n.a.                       | N/m               |
| $k_{yf}$                         | Gearbox casing y-stiffness                   | $2.036 \times 10^8$        | n.a.                       | N/m               |
| $c_{x1}, c_{x2}, c_{y1}, c_{y2}$ | Bearing damping                              | $2.134 \times 10^4$        | $1.8 \times 10^3$          | N.s/m             |
| $c_c$                            | Coupling damping <sup>12</sup>               | 23.1                       | 10                         | N.m rad/s         |
| $c_{xf}$                         | Gearbox casing x-damping                     | 1995.56                    | n.a.                       | N.s/m             |
| $c_{yf}$                         | Gearbox casing y-damping                     | 2005.80                    | n.a.                       | N.s/m             |

<sup>9</sup> The motor inertia for Machine B was adopted from Chaari et al. [52].

<sup>10</sup> The motor inertia for Machine B was adopted from Chaari et al. [52].

<sup>11</sup> The coupling stiffness value for Machine B was adopted from Mohammed et al. [53].

<sup>12</sup> The coupling stiffness value for Machine B was adopted from Mohammed et al. [53].

Table A.2. Values used for generating faults for Machine A and B. Fault Value (N) indicates the faulty value utilised for Machine N.  $\mathcal{N}(a, b)$  refers to normal distribution with mean and standard deviation  $a$  and  $b$ , respectively.  $\mathcal{U}(c, d)$  refers to a uniform distribution with lower and upper sampling bounds of  $c$  and  $d$ , respectively. For a more detailed outline of parameters and their interpretation, refer to [38].

| Crack Parameters           |                |               |                                  |                                  |               |
|----------------------------|----------------|---------------|----------------------------------|----------------------------------|---------------|
| Quantity Name              | Symbol in [38] | Healthy Value | Fault Value (A)                  | Fault Value (B)                  | Unit          |
| Left Side Crack Depth      | $q_0$          | 0             | 2.693                            | 1.697                            | mm            |
| Right Side Crack Depth     | $q_2$          | 0             | 2.693                            | 1.697                            | mm            |
| Crack Angle                | $\nu$          | 90            | 90                               | 90                               | °             |
| Crack Width                | $W_c$          | 0             | 16                               | 20                               | mm            |
| Chip Parameters            |                |               |                                  |                                  |               |
| Quantity Name              | Symbol in [38] | Healthy Value | Fault Value (A)                  | Fault Value (B)                  | Unit          |
| Chip Depth                 | $b$            | 0             | 2.42                             | 1.52                             | mm            |
| Chip Width                 | $c$            | 0             | 16                               | 20                               | mm            |
| Spall Parameters           |                |               |                                  |                                  |               |
| Quantity Name              | Symbol in [38] | Healthy Value | Fault Value (A)                  | Fault Value (B)                  | Unit          |
| Spall Length               | $l_s$          | 0             | 740                              | 466                              | $\mu\text{m}$ |
| Spall Width                | $w_s$          | 0             | 15.2                             | 19                               | mm            |
| Spall Depth                | $h_s$          | 0             | 1                                | 1                                | mm            |
| Spall Start Location       | $x_1$          | 0             | 3.87                             | 2.437                            | mm            |
| Pitting Parameters         |                |               |                                  |                                  |               |
| Quantity Name              | Symbol in [38] | Healthy Value | Fault Value (A)                  | Fault Value (B)                  | Unit          |
| Mean Pitting Line          | $\bar{u}$      | 0             | 4.39                             | 2.76                             | mm            |
| Standard Pitting Deviation | $\sigma_u$     | 0             | 296                              | 186                              | $\mu\text{m}$ |
| Sampled Pit Location       | $u_i$          | n.a.          | $\mathcal{N}(\bar{u}, \sigma_u)$ | $\mathcal{N}(\bar{u}, \sigma_u)$ | mm            |
| Sampled Pit Angular Size   | $\theta_i$     | n.a.          | $\mathcal{U}(30, 30)$            | $\mathcal{U}(30, 30)$            | °             |
| Sampled Pit Radius         | $r_i$          | n.a.          | $\mathcal{U}(80, 80)$            | $\mathcal{U}(80, 80)$            | $\mu\text{m}$ |
| Number of Samples          | N.A.           | 0             | 2100                             | 1800                             | –             |

Table A.3. Known gearbox parameters for the PHM23 experimental setup [48].

| Gear parameter                    | Pinion gear | Driven gear |
|-----------------------------------|-------------|-------------|
| Tooth number                      | 40          | 72          |
| Module                            | 3 mm        | 3 mm        |
| Base circle diameter              | 112.763 mm  | 202.974 mm  |
| Pitch diameter                    | 120 mm      | 216 mm      |
| Pressure angle                    | 20°         | 20°         |
| Addendum coefficient              | 1           | 1           |
| Coefficient of top clearance      | 0.25        | 0.25        |
| Diametral pitch                   | 8.4667      | 8.4667      |
| Engaged angle                     | 19.7828°    | 19.7828°    |
| Circular pitch                    | 9.42478 mm  | 9.42478 mm  |
| Addendum                          | 4.5 mm      | 3.588 mm    |
| Dedendum                          | 2.25 mm     | 3.162 mm    |
| Addendum modification coefficient | 0.5         | 0.196       |
| Addendum modification             | 1.5 mm      | 0.588 mm    |
| Fillet radius                     | 0.9 mm      | 0.9 mm      |
| Tooth thickness                   | 5.8043 mm   | 5.1404 mm   |
| Tooth width                       | 85 mm       | 85 mm       |
| Theoretical centre distance       | 168 mm      | 168 mm      |
| Actual centre distance            | 170.002 mm  | 170.002 mm  |

Table A.4. Values used to generate pitting faults in the PHM23 case study. The fault value indicates the maximum damage case.

| Quantity Name              | Symbol in [38] | Healthy Value | Fault Value                      | Unit          |
|----------------------------|----------------|---------------|----------------------------------|---------------|
| Mean Pitting Line          | $\bar{u}$      | 0             | 3.03                             | mm            |
| Standard Pitting Deviation | $\sigma_u$     | 0             | 757                              | $\mu\text{m}$ |
| Sampled Pit Location       | $u_i$          | n.a.          | $\mathcal{N}(\bar{u}, \sigma_u)$ | mm            |
| Sampled Pit Angular Size   | $\theta_i$     | n.a.          | 90                               | °             |
| Sampled Pit Radius         | $r_i$          | n.a.          | 425                              | $\mu\text{m}$ |
| Number of Samples          | N.A.           | 0             | 294                              | —             |

## Appendix B Additional comprehensive numerical investigation results

This section presents some additional results from the comprehensive numerical investigation from [section 3](#). These are shown for the source- and target-only methods in [Table B.1](#) and calibrated methods in [Table B.2](#).

Table B.1. Per class and total classification accuracies for the source- and target-only methods. Given accuracies were calculated from 15 repetitions of the experiments. The total column reports the mean classification accuracy with one standard deviation.

| Uncalibrated techniques |                     |                                         |         |       |      |       |     |          |                                         |       |      |       |     |           |
|-------------------------|---------------------|-----------------------------------------|---------|-------|------|-------|-----|----------|-----------------------------------------|-------|------|-------|-----|-----------|
|                         |                     | Source-only classification accuracy [%] |         |       |      |       |     |          | Target-only classification accuracy [%] |       |      |       |     |           |
|                         | Transfer Experiment | Transfer case                           | Healthy | Crack | Chip | Spall | Pit | Total    | Healthy                                 | Crack | Chip | Spall | Pit | Total     |
| TIM                     | DOF                 | 6 to 8                                  | 100     | 98    | 100  | 97    | 100 | 99.1±1.8 | 100                                     | 40    | 29   | 60    | 60  | 55.9±16   |
|                         |                     | 6 to 10                                 | 100     | 100   | 98   | 90    | 100 | 97.5±7.4 | 100                                     | 60    | 46   | 41    | 66  | 61.2±15   |
|                         |                     | 8 to 10                                 | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                                     | 30    | 20   | 40    | 70  | 50.0±14   |
|                         | Noise               | SNR=10 dB                               | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                                     | 97    | 75   | 100   | 91  | 92.3±9.9  |
|                         |                     | SNR=0 dB                                | 53      | 45    | 44   | 100   | 64  | 61.4±30  | 100                                     | 53    | 29   | 59    | 36  | 53.5±34   |
|                         |                     | SNR=-10 dB                              | 7       | 0     | 17   | 84    | 21  | 26.3±8.8 | 100                                     | 0     | 0    | 0     | 0   | 16.8±0.33 |
|                         | Location            | Pinion to gear                          | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                                     | 40    | 30   | 58    | 80  | 60.0±19   |
|                         |                     | Pinion to casing                        | 100     | 90    | 27   | 100   | 0   | 61.9±12  | 100                                     | 75    | 40   | 50    | 80  | 67.7±9.8  |
| TDM                     | Machine             | Mod A-6 to Mod B-6                      | 75      | 0     | 10   | 0     | 50  | 25.0±18  | 100                                     | 66    | 30   | 30    | 70  | 57.4±9.6  |
|                         |                     | Mod A-8 to Mod B-8                      | 90      | 0     | 0    | 0     | 70  | 29.6±9.8 | 100                                     | 50    | 40   | 90    | 50  | 64.5±16   |

Table B.2. Per class and total classification accuracies for the weight fine-tuning and simulation-driven domain adaptation methods. Given accuracies were calculated from 15 repetitions of the experiments. The total column reports the mean classification accuracy with one standard deviation.

| Transfer learning calibrated techniques |                     |                    |         |       |      |       |     |          |                                  |       |      |       |     |          |
|-----------------------------------------|---------------------|--------------------|---------|-------|------|-------|-----|----------|----------------------------------|-------|------|-------|-----|----------|
| Fine-tuning classification accuracy [%] |                     |                    |         |       |      |       |     |          | SDDA classification accuracy [%] |       |      |       |     |          |
|                                         | Transfer Experiment | Transfer case      | Healthy | Crack | Chip | Spall | Pit | Total    | Healthy                          | Crack | Chip | Spall | Pit | Total    |
| TIM                                     | DOF                 | 6 to 8             | 100     | 99    | 100  | 100   | 100 | 99.8±0.5 | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | 6 to 10            | 100     | 100   | 98   | 90    | 100 | 97.5±7.3 | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | 8 to 10            | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         | Noise               | SNR=10 dB          | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | SNR=0 dB           | 100     | 67    | 77   | 87    | 78  | 81.0±30  | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | SNR=-10 dB         | 100     | 4     | 1    | 28    | 2   | 23.9±11  | 100                              | 20    | 20   | 98    | 5   | 46.9±9.0 |
|                                         | Location            | Pinion to gear     | 100     | 100   | 100  | 100   | 100 | 100±0    | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | Pinion to casing   | 100     | 80    | 51   | 60    | 80  | 73.0±13  | 100                              | 100   | 100  | 100   | 100 | 100±0    |
| TDM                                     | Machine             | Mod A-6 to Mod B-6 | 100     | 70    | 30   | 30    | 70  | 58.3±9.3 | 100                              | 100   | 100  | 100   | 100 | 100±0    |
|                                         |                     | Mod A-8 to Mod B-8 | 100     | 50    | 50   | 90    | 50  | 67.6±14  | 100                              | 100   | 100  | 100   | 100 | 100±0    |