Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-026-02904-5

Synthetic process modeling for ring rolling via physics-guided GANs and analytical transfer learning

Johannes Seitz[1] · **Qinwen Wang**[2] · **Luca Quagliato**[3,4] · **Tobias Moser**[1] · **Taeyong Lee**[3] · **Alexander Brosius**[2] · **Bernd Kuhlenkötter**[1]

Received: 17 December 2025 / Accepted: 4 June 2026

Abstract

Modeling metal forming processes, and specifically ring rolling as considered in this research, is crucial for process design and the development of new product variants. Although finite element simulations and analytical modeling are common practices to perform process predictions, they can still present limitations such as high computational time, adherence to reality, and modeling accuracy. Trying to address all the above, this research proposes a novel generative machine learning solution for the creation of accurate and physics-compliant synthetic data for the ring rolling process, enabling almost real time and accurate process modeling within latent space. The proposed data generation method is based on multivariate generative adversarial network (GAN) combined with a physics-guided neural network (PGNN) to incorporate constraints relevant to the ring rolling process and integrate them into an auxiliary loss granting physical consistency. The proposed GAN architecture is then annotated as a physics-guided auxiliary GAN (PG-A-GAN). The physical constraints introduced are relevant to an analytical slip-line model for the force, a surrogate model for the torque, volume consistency, and an analytical model for the time–diameter relationship. Considering the duality of the physical modeling approach for the force and torque, where data-driven machine learning is stirred by analytical models, the proposed approach is defined as analytical transfer learning. Findings reveal that employing PGNNs in the GANs learning process improves physical loss in data generation, combined with a slight increase in data distribution similarity with respect to experimental instances. To enable modeling of specific and customized rolling instances, a condition based on the final sought ring geometry was introduced, within an auxiliary classifier generative adversarial network (ACGAN) framework. The proposed architecture allows the generation of multivariate, physically constrained, rolling time series and highlights the feasibility of such a modeling approach within the latent space and might be extended to other manufacturing processes by adapting modeling and retraining the proposed architecture.

Keywords Generative adversarial networks (GANs) · Physics-informed neural networks (PINNs) · Transfer learning · Process modeling

Introduction

Ring rolling is a wide-spread manufacturing technology utilized to produce seamless rings employed in various key industrial sections, among them aerospace, heavy machinery, and energy (Allwood et al., 2005; Berti et al., 2015). Considering the complex interaction between the tools and the ring, and the incremental nature of the process itself, the last 50 years have seen a world-wide effort to shed light on these complex relationships by developing a large variety of models belonging to different categories (Allwood et al., 2005).

Up to the early 1990s, most of the work on the ring rolling process saw the predominance of analytical modeling,

✉ Johannes Seitz
seitz@lps.ruhr-uni-bochum.de

✉ Luca Quagliato
luca.quagliato@unitn.it

1 Chair of Production Systems, Ruhr-University Bochum, Bochum, Germany

2 Chair of Forming and Machining Technology, TUD Dresden University of Technology, Dresden, Germany

3 Department of Mechanical and Biomedical Engineering, Ewha Womans University, Seoul, Republic of Korea

4 Department of Industrial Engineering, University of Trento, Trento, Italy

Published online: 21 July 2026

coupled with experimental characterizations for validation purposes, for the prediction of the diameter expansion (Hua & Zhao, 1997), radial force and torque (Hawkyard et al., 1973), and pressure distribution (Mamalis et al., 1976). Afterwards, the rise of finite element analysis (FEA) as the investigation tool of choice allowed for the development of process simulations for the ring rolling process (Kim et al., 1990), followed by a full array of research on various aspects, including geometrical expansion, pressure distribution, and torque (Yea et al., 2003). It is worth noting that analytical, or empirical, models have often been used in conjunction with FEA analysis either for validation purposes, (Berti et al., 2015; Quagliato et al., 2018), for virtual modeling considering the influence of various process parameters on process outcomes (Wang et al., 2010), and for formability estimation (Kil et al., 2015). On top of that, it ought to be pointed out that most of the recent work on ring rolling focusses mostly on profiled rings (Mirandola et al., 2022), with a clear focus on the deformation behavior (Ge et al., 2025), process control (Lafarge et al., 2021), and force and torque prediction by machine learning (Raftar & Parvizi, 2018). Though scholars shifted their focus from flat to profiled ring rolling, most of the industrial compartment, and the relevant R&D departments, are still employing, researching, and improving the flat ring rolling process (Arthington et al., 2020; Hua et al., 2017). This fact highlights the need for this research and provides a background for the application scenarios envisioned for the developed model, presented further on.

As summarized so far, throughout the last five decades, scholars focused on the ring rolling process by employing analytical, empirical, FEA, and more recently machine learning modeling approaches with various aims, among which process analysis, force and torque prediction, and process control. In this regard, the introduction of machine learning led to huge progress in the field of manufacturing engineering (Quagliato et al., 2025) and especially metal forming (Cao et al., 2024). Accordingly, it is perhaps not surprising that some of this effort was oriented towards machine learning applications for the ring rolling process.

From a general perspective, machine learning (ML) has been employed in manufacturing engineering for more than a decade, though the last 5 years saw an exponential growth in research and publications. In this regard, the common approach involves the data collection, or *ad-hoc* database generation, for the training of the relevant models (Mirandola et al., 2021). Recent examples involve blank optimization design for the cross-rolling process based on the Artificial Neural Network (ANN) (Deng & Mao, 2015), process parameters optimization for the radial–axial ring rolling (RARR) process (Lee et al., 2023), and Differential Evolution (DE)-optimized ANN model for the prediction of force and torque in the RARR process (Parvizi & Raftar, 2019). Other interesting machine learning applications, relevant to the ring rolling process, see the utilization of a Radial Basis Function Neural Network (RBFNN), optimized through a Genetic Algorithm (GA), for the prediction of the ring throughout the process by analyzing the position of the tools (Wang et al., 2017). Besides that, combinations of different machine learning models in an architectural manner also showed to be beneficial on the training performance and shown recently for the fatigue life estimation of the mandrel of ring rolling mills (Moser et al., 2025).

Although the contributions summarized in the last paragraph provide powerful methodologies for process control and optimization, they also require the establishment of a dataset on which to be trained and are also susceptible to difference in the features distributions when predicting either within or outside of the latent space. Modified versions of Physics-Informed Neural Networks (PINNs), defined as Expert-Informed Neural Networks (EINNs), can also be employed for an effective modeling also outside of the latent space, but require a time-consuming equation discovery phase, which might be impractical for some industrial applications (Modanloo et al., 2025).

To address these potential limitations, Transfer Learning (TL) approaches are being currently used in various manufacturing modeling applications (Li & Zheng, 2020), and the ring rolling process is no exception. In this regard, TL has been used in the ring rolling process to investigate ring roundness by classification tasks employing experimental and synthetic data (Seitz et al., 2023) generated by Generative Adversary Neural Networks (GANs). TL also showed to be highly beneficial in scenarios where full retraining is not possible or would lead to poor performance due to the lack of sufficient data in the target domain (Fernandez et al., 2025) and to overcome limitations in data sharing among different institutions (Guo et al., 2024). Indeed, TL can reduce the computational effort and time required for an efficient deployment of effective machine learning solutions with minimum retraining, but a crucial aspect is represented by the dataset employed for the initial training or for the subsequent domain adaptation by TL. In this regard, the combined application of synthetic data generation for the initial pre-training and TL for the fine tuning was recently demonstrated for the case of part classification in manufacturing engineering (Ruediger-Flore et al., 2023). In this sense, high quality synthetic data are the most viable option to reduce the data generation, and post-processing, but require careful design to avoid introducing bias in the dataset, subsequently reflected in the training process.

One of the most employed approaches for synthetic data generation is based on the auxiliary classifier generative adversarial network (ACGAN), an improved version of GANs where both generator and discriminator are trained to predict the correctness likelihood as well as a target

variable class. The latter characteristic influences the generator in creating more meaningful samples and sharpens the discriminator in its evaluation of the correctness of the generated sample (Petrik et al., 2023). ACGAN modelling has been already employed in different industrial applications, such as additive manufacturing for the generation of reliable melt pool images in laser power bead fusion additive manufacturing (Petrik et al., 2023), working patterns generation in metal-melting induction furnaces and continuous casting machines (Bazarbaev et al., 2022), and ring quality prediction in the RARR process (Fahle et al., 2022). It is worth pointing out that other than ACGANs also other methodologies have been employed for synthetic data generation, such as variational autoencoder (VAE), (Ferré et al., 2025), and conditional variational autoencoder (CVAE), (Liu et al., 2024). However, VAE is a purely probabilistic auto-encoding model without adversarial loss or label conditioning whereas CVAE maximizes a variational lower bound on the data likelihood, yielding blurrier outputs but providing a solid probabilistic encoder–decoder model. On top of that, also copulas (Restrepo et al., 2023) and diffusion-based models (Yi et al., 2024) have been proposed in the recent literature as alternatives to generate synthetic data, but the structure of the conditional part of the framework can make the integration of customizable constraints based on user knowledge or physical laws non-trivial, as intended in this research.

As summarized so far, ring rolling is a key manufacturing process and has been investigated by various employing analytical modeling, FEA, and, more recently, machine learning approaches. As concerns the data-driven machine learning methodologies, although they demonstrated high prediction accuracy, they are still limited by the need for database generation and pre-processing. To alleviate this burden, synthetic data generation approaches have been proposed in the literature, but their data-driven architecture makes them prone to lose focus on the overall governing behavior of the system, ruled by physics-bounded rules.

To this end, recent reviews highlight physics-informed machine learning (PIML) as a rapidly expanding direction across manufacturing, emphasizing its role in improving generalization and reliability when data are limited or operating conditions shift (Ra et al., 2026). In the structure–property context of the additive-manufacturing process, Faegh et al. (2025) synthesize how embedding mechanistic knowledge into learning pipelines helps connect sparse experiments with high-dimensional process signatures in a physically meaningful way. Focusing specifically on metal additive manufacturing, Farrag et al. (2025) discussed how physics constraints can reduce the dependence on exhaustive datasets and enable more trustworthy prediction and control under varying process regimes. Beyond reviews, the differentiable PIML formulation proposed by Oddiraju et al. (2025) and (Modanloo et al., 2025) exemplifies how hybrid physics–ML models can be structured to support fast, optimization-ready surrogates for well-established yet complex manufacturing processes. However, most PIML efforts in manufacturing focus on prediction and control, whereas the complementary problem of generating physics-consistent synthetic datasets, particularly to support data augmentation and transfer learning across domains has received far less attention despite growing interest in generative backbones such as diffusion and transformer-based sequence models (Ogoke et al., 2024; Sikder et al., 2025).

Building on the needs summarized in the previous paragraphs, this work introduces a novel Generative Adversarial Neural Network (GAN) enhanced with a physics-guided auxiliary loss, enriching the adversarial learning framework with the principles of Physics-Informed Neural Networks (PINNs). By embedding key process constraints directly into the discriminator, the model generates synthetic ring rolling data that is not only statistically realistic but also physically consistent. The approach delivers sub-millisecond runtimes, orders of magnitude faster than finite element simulations, making it possible to produce vast, high-fidelity datasets for transfer learning, process modeling, and data augmentation. Moreover, both unconditioned and conditioned synthetic data generation were implemented, showing the reliability and effectiveness of the proposed solution in both scenarios. Beyond its immediate application to ring rolling, the proposed framework establishes a generalizable pathway for integrating physics into generative models for manufacturing science both in unconditioned and conditioned GAN applications.

In this research, the training dataset, composed of 129 tangential cold ring rolling (TCRR) experiments, is presented in Sect. "Tangential cold ring rolling experiments" and represents the dataset employed at different stages throughout this research. Additional information relevant to the material characterization, Sect. "Material characterization", and the TCRR FEA model employed for analytical transfer learning and surrogate model tuning, are reported in Sect. "Finite element model implementation". In this research, the term analytical transfer learning (ATL) is used operationally to describe the adaptation of pre-defined analytical equations for the ring rolling process by learning data-driven correction factors for their coefficients. This adaptation is achieved by using machine input and ring geometry to tune trainable parameters that reduce the mismatch between the analytical formulation and the target experimental domain. In this sense, ATL is related to the general idea of transfer learning (Pan & Yang, 2010), in that a pre-existing model structure is adapted to a specific target setting. However, ATL is not introduced here as a new general transfer-learning theory, rather as a task-specific hybrid modeling concept for the

present work, combining analytical process knowledge with data-driven surrogate correction.

Due to the variety of approaches and modelling techniques employed in this research, a summary of the content organization of chapter 3 is provided in the following paragraphs. First, after a summary of the works related to this research, presented in Sect. "Fundamentals of GAN modeling by deep convolutional neural network (DCNN)", the theoretical background and the implementation for the baseline DCGAN, for unconditioned modeling, and ACGAN, when a condition on the synthetic data generation is imposed, are presented in Sect. "Deep convolutional GAN (DCGAN) model implementation". The custom-made observer module enabling for the selection of the optimal generator model for the DCGAN benchmark model, based on the properties of the generated data distribution, is reported in Sect. "Observer for the DCGAN model". Coming now to the proposed methodology, the physical modelling methodology for rolling force and torque, volume consistency and the time-diameter relationship is presented in Sect. "Physics-informed and analytical transfer learning loss modeling" to establish the prerequisites for physics integration in the data generation process through physics-guided auxiliary GAN (PG-A-GAN) and includes also the analytical transfer learning modeling. The aim of Sect. "Physics-informed and analytical transfer learning loss modeling", core of the proposed approach, is twofold. On the one hand, it presents the implementation of the analytical models for ring outer diameter prediction and volume consistency throughout the process time. On the other hand, it also focusses on the surrogate models for the rolling force and torque, built to correct the analytical predictions to have a higher adherence to the real (experimental) data distribution, and based on both experimental and finite element results. The last section of chapter 3, namely Sect. "Benchmark, physics-guided, and conditional implementations", summarizes the key results of the analytical transfer learning process, in terms of setting and tuning of the force and torque surrogate models, the analysis steps between the benchmark DCGAN and the proposed PG-A-GAN, and lastly the conditional synthetic data generation. It ought to be pointed out that the conditional synthetic data generation is applied only by employing the proposed PG-A-GAN, resulting in the definition of an auxiliary conditioned PG-A-GAN, or AC-PG-A-GAN.

The results chapter 4 is subdivided into three sections, each one relevant to one of the key modeling aspects introduced in chapter 3. First, Sect. "DCGAN with custom-observer synthetic data generation (benchmark model)" presents the results of the baseline and unconditioned DCGAN and is intended to provide a state-of-the-art (SOTA) benchmark for the proposed modeling approach. This is followed by the presentation of the results for the unconditioned PG-A-GAN, together with the comparison with the performance of the benchmark DCGAN, supplemented by the custom-made observer module

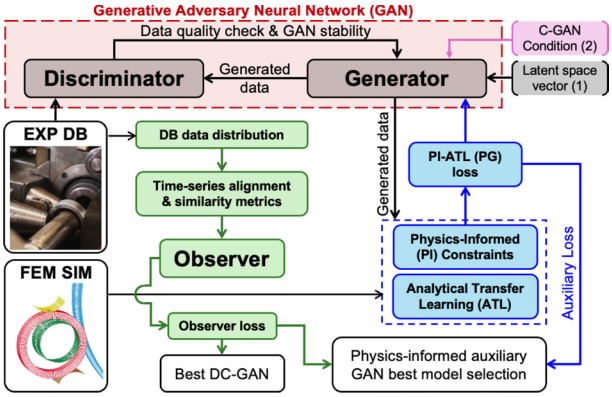


Fig. 1 Schematic diagram of this research. The DCGAN (Deep Convolutional Generative Adversarial Network) represents the benchmark for the proposed physics-guided auxiliary GAN (PG-A-GAN) and includes physics-informed loss, analytical transfer learning loss, and observer loss. Note, that (1) represents the initialization of the generator while (2) the condition which can be applied for the generation of specific synthetic data (e.g. a condition on the final geometry)

of Sect. "Observer for the DCGAN model", as summarized in Sect. "PG-A-GAN: synthetic data generation". The last section of chapter 4 is represented by the application of the AC-PG-A-GAN for the real-time generation of rolling time series for differing ring dimension conditions within latent space. The methodology and structure of the paper are presented in Fig. 1. The proposed integration of physics-guided constraints into DCGANs, together with the custom observer for model selection and the conditional synthetic data generation via the AC-PG-A-GAN, represents a step forward in process modeling and synthetic data generation for tangential cold ring rolling. As a remark it ought to be pointed out that, beyond this specific case, the PG-A-GAN framework proposed in this research offers a transferable methodology that can be extended to other manufacturing processes.

Experimental and finite element analyses

The aim of this first section is to provide the details relevant to the experiments and finite element simulations employed at different stages in this research. To this end, a summary of the ring rolling experiments is summarized in Sect. "Tangential cold ring rolling experiments" and is followed by the properties' characterization for the ring material, Sect. "Material characterization", employed in the definition of the finite element model, detailed in Sect. "Finite element model implementation".

Tangential cold ring rolling experiments

Tangential cold ring rolling (TCRR) experiments were carried out on a retrofitted UPWS 31,5.2 from Profiroll

Technologies GmbH, as shown in Fig. 2a. A simplified schematic of the process and the active tools is presented in Fig. 2b. During the forming operation, the main roll advances toward the mandrel at a defined feed rate while rotating at a specified speed resulting in a reduction of the ring thickness while the inner and outer diameters expand. In the employed TCRR process, the mandrel is free to rotate with the ring along its axis while the guide rolls are used to stabilize the ring throughout the process. The forming process terminates when the main roll gets in contact with the mandrel. The rolling machine employed in the experiments is equipped with different sensors to record the process parameters, *id est* the main roll displacement, ring growth, rolling force and rolling torque.

The displacement of the main roll is recorded with a linear displacement transducer, the ring growth is measured using two metering rolls and a potentiometer, while both rolling force and torque are measured using strain gauges. All measurement signals are acquired via a data acquisition device (HBM QuantumX). An example of the above-mentioned experimental outputs for a ring with an initial outer diameter of 63 mm, 6.5 mm thickness, and 19 mm height, and processed with a mandrel feed rate of 0.2 mm/s, a main roll rotational speed of 100 rpm, and 15 bar of pressure for the guide rolls are reported in Fig. 2c.

As summarized at the end of the previous chapter, and depicted in Fig. 1, the discriminator verifies to adherence to reality of the synthetic data provided by the generator based on a user-defined dataset. In this research, this dataset has been defined considering a total of 129 instances extracted from the experimental results carried out employing the set-up described in Fig. 2a.

Each instance is composed of six features, namely time [s], outer diameter [mm], ring thickness [mm], rolling force [kN], rolling moment [Nm], and centering force [bar]. The summary of the experimental levels employed for the generation of the dataset is provided in Table 1 while the final geometry is controlled by the roll gap.

Material characterization

In this investigation, all the rings were manufactured by employing the 100Cr6 steel (1.3505, AISI 52100), a material commonly used in the bearing industry due to its high hardness and wear resistance, and the ring preforms were cut from a seamless, soft annealed steel tube. The 100Cr6 steel mechanical properties were determined by performing compression tests on cylindrical samples at room temperature, Fig. 3a.

The compression tests were conducted on cylindrical specimens (Ø5.8 mm × 6.3 mm) following the guidelines of DIN 50106. The plates of the testing machine were precision turned and polished to achieve a parallelism of 0.005 mm. To minimize frictional effects between the specimen and platens, a thin PTFE film was placed on the platens. The specimens were positioned centrally on the lower platen, and the upper platen was equipped with

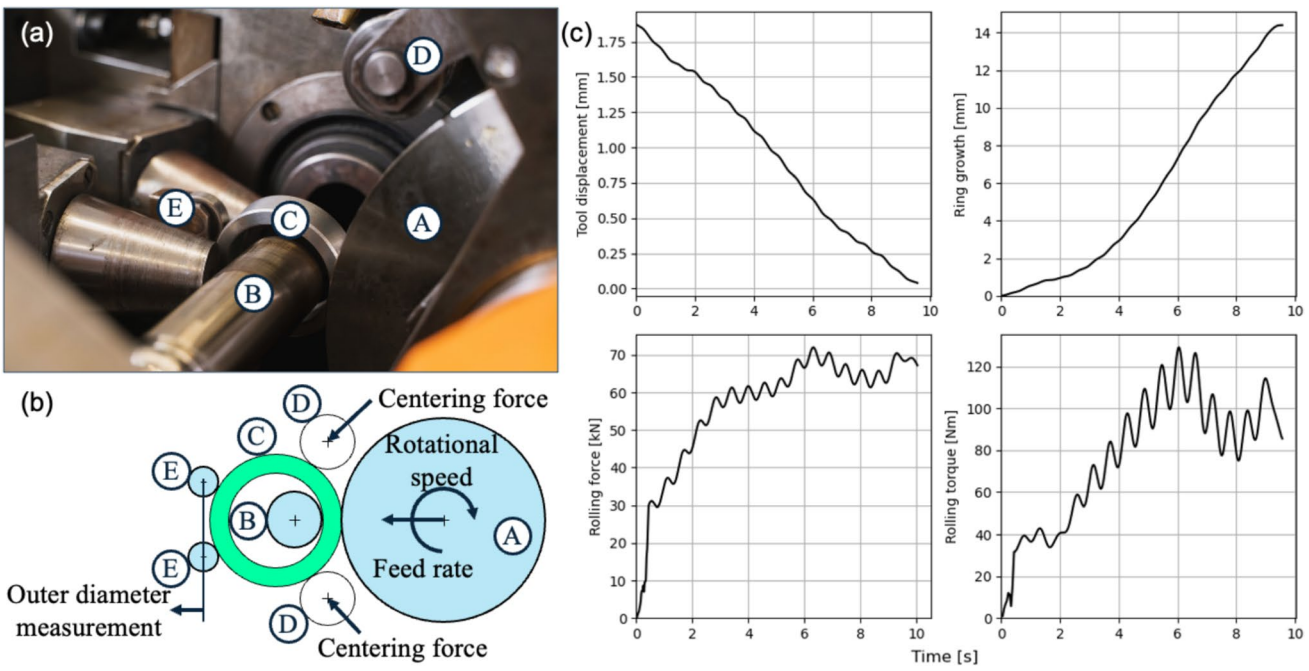


Fig. 2 **a** Tangential cold ring rolling (TCRR) at the TU Dresden and **b** corresponding schematic illustration. **c** Example of experimental outputs for a ring with an initial outer diameter of 63 mm

Table 1 Factors and levels employed in the TCRR experiments for the definition of the experimental dataset

Factor	Levels	Mean	SD
Initial ring outer diameter (mm)	60, 60.75, 61, 61.5, 62.25, 63	61.417	1.08
Initial ring thickness (mm)	5, 5.375, 5.5, 5.75, 6.125, 6.5	5.708	0.54
Ring height (mm)	19	19	-
Roll gap (mm)	4.2, 4.4	4.3	0.141
Mandrel feeding speed (mm/s)	0.05, 0.1, 0.15, 0.2, 0.25, 0.3	0.175	0.094
Main roll rotational speed (rpm)	50, 63, 80, 100, 125	83.6	29.788
Guide roll pressure (bar)	7, 10, 15	10.667	4.041

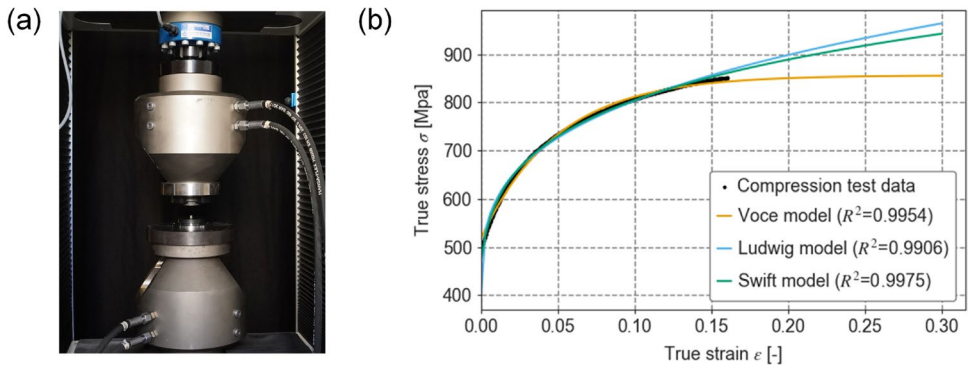


Fig. 3 Fitting of the 100Cr6 steel true plastic strain by various flow stress models with relevant R^2 correlation

a spherical seat to ensure automatic alignment, so that the loading axis coincides with the specimen axis. Two different strain rates were applied: 0.00025 s^{-1} for determining the Young's modulus, and 0.0067 s^{-1} for capturing the plastic behavior. The tests were carried out on a 250 kN universal testing machine from Hegewald & Peschke. Deformation of the compression samples was recorded using an ARAMIS 4 M system for digital image correlation. Various flow stress models were employed for fitting the experimental true stress–strain curves, as shown in Fig. 3, and for each method the R^2 value was calculated to assess the fitting performance. Among them, the Voce and Swift models exhibited the highest correlation, and, for this reason, the Voce model was considered and calibrated to the following constants: $\sigma_p = 518.64 + 337.78 \bullet (1 - e^{-20.27 \cdot \varepsilon_p})$. As a final remark, since the temperature during cold TRR does not exceed 100 °C, thermal effects were neglected.

Finite element model implementation

In this research, 3D finite element (FE) simulations were implemented for the calibration of the friction coefficient during the TCRR process, to be subsequently employed for the empirical torque model considered in the definition of the analytical transfer learning scheme, as shown in Fig. 1. To this end, the commercial software LS-DYNA version 14.0.0 was employed considering a standard explicit algorithm solution and by modeling only half of the ring to minimize the computational time. In the FE simulations, the ring is discretized into approximately 13 k solid elements using brick solid element with 8 nodes and one integration point, ELFORM 1 in LS-DYNA, while a mass time scaling factor of 2 was applied to further reduce computational effort. The contact conditions between the ring and the tools were modeled using the Coulomb model with a friction factor of 0.15, calibrated to match the experimental results. The simulation was implemented for an average of approximately 330 core hours, utilizing 32 cores on an Intel® Xeon® Platinum 8470 processor within the high-performance computing system of the Technical University Dresden.

Each FE simulation was optimized to minimize the differences between experimental and numerical results for the outer diameter expansion, rolling force, and rolling torque by adapting the friction coefficient. In other words, starting with the above-mentioned value of 0.15, the friction conditions allowing for the best representation of the 3 selected target outputs, and for each simulation, were inversely calibrated by FE analyses employed the model presented in this section. An example of the FE results, superimposed on the relevant experimental results, for a 63 mm outer diameter, 6.5 mm thickness, and 19 mm height ring is shown in Fig. 4 in terms of outer diameter, rolling force and rolling torque over the process time.

The reason for implemented FE simulations for the TCRR process is twofold. On the one hand, FE results offer a precious insight into the contact conditions among mandrel, main roll, and the ring, crucial for the definition of a reliable contact length to be employed in the force estimation model, as defined in subsequent Sect. "Physics-informed and analytical transfer learning loss modeling". On the other hand,

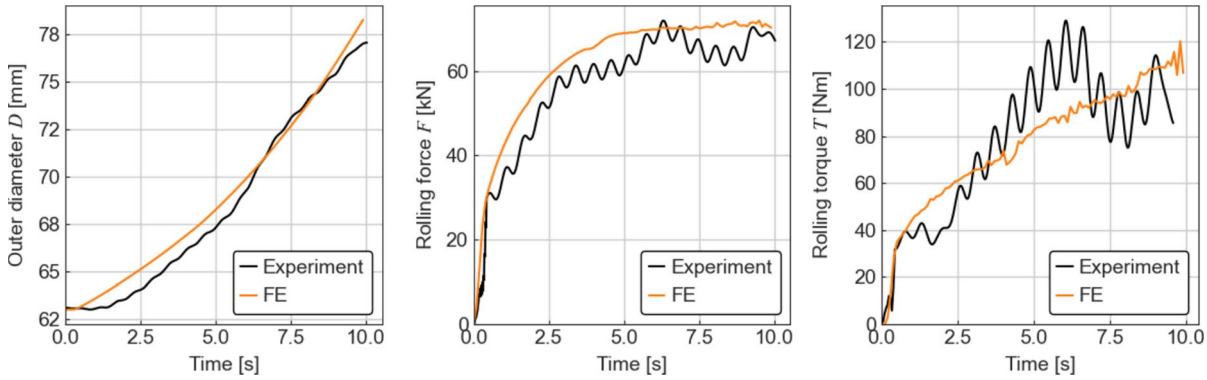


Fig. 4 Examples of experimental and FE results for a selected instance

the calibration of the friction coefficient, as described in the previous paragraph, is also essential for the estimation of the rolling torque from the rolling force, also summarized in Sects. "Physics-informed and analytical transfer learning loss modeling" and "Benchmark, physics-guided, and conditional implementations".

Process modeling and synthetic data generation

This chapter presents first a summary of the work relevant to this research in terms of GANs including both deep convolutions (DCGANs) and ACGANs. It ought to be remarked that DCGANs are employed for unconditioned synthetic data generation while ACGANs include one or more class labels to steer data generation towards user-selected conditions. In other words, for the case of the TCRR process and the research hereby considered, applying an ACGAN instead of a DCGAN implies considering a condition, for instance on the final geometry of the ring, when generating synthetic data.

The introduction to GANs architecture presented in Sect. "Fundamentals of GAN modeling by deep convolutional neural network (DCNN)" is required for the understanding of how the DCGANs model, presented in Sect. "Deep convolutional GAN (DCGAN) model implementation", was coupled with a custom-developed observer unit, as shown in Fig. 1 and explained in Sect. "Observer for the DCGAN model", and employed as the benchmark algorithm in this research. This is followed by the introduction to the concept of physics-guided GANs, its implementation in this research, and afterwards by the presentation of the details on both physics-guided and analytical transfer learning models employed as auxiliary losses for the definition of the proposed physics-guided auxiliary GAN (PG-A-GAN), as per Sect. "Physics-informed and analytical transfer learning loss modeling". Finally, Sect. "Benchmark, physics-guided, and conditional implementations" is devoted to a summary of the key modeling approaches employed in this research to provide a pathway guiding the understanding and interpretation of the results chapter 4. In summary, this chapter presents the fundamentals associated with GANs, the current state of the art, and introduces the proposed extension through physics-guided auxiliary loss and the necessary results to implement the unconditioned PG-A-GAN and the condition label-driven AC-PG-A-GAN.

Fundamentals of GAN modeling by deep convolutional neural network (DCNN)

The modeling approach employed in this research is a generative adversarial neural network (GAN) (Goodfellow et al., 2014) with a deep convolutional network (DC) (Krizhevsky et al., 2017) architecture within the GAN. In a second step, a condition for ring geometries is implemented using ACGANs (Odena et al., 2017). A GAN is a framework for data generation that combines two models in an adversarial process. The generator (G) predicts samples ($\hat{x}$) reflecting the data distribution from a noise vector (latent space vector) $p_z(z)$ and the discriminator (D) estimates the probability of the data's origin. During adversarial training, the discriminator hyperparameters are optimized to maximize the correct assigning of the labels or, in other words, whether the data being tested is real or synthetically generated. In this regard, the probability that the sample x is from the dataset and not from the generator distribution p_g, is described by $D(x)$.

In the adversarial training, the generator's objective is to fool the discriminator by making $D(G(x))$ as close as possible to 1 so that the discriminator minimizes $1 - D(G(z))$. In the ideal case, represented by the GAN equilibrium, the discriminator loss is equal to 0.5, which happens if the generator's output distribution exactly matches the real data distribution. The adversarial game can be summarized by the value function V, (Goodfellow et al., 2014), that takes logarithmic loss functions into account and is defined to:

$$\min\max V(D,G) = \mathbb{E}_{x\sim p_{data}(x)} \underbrace{\left[\log\left(D(x)\right)\right]}_{Discriminator} + \mathbb{E}_{z\sim p_z(z)} \underbrace{\left[\log\left(1-D(G(z))\right)\right]}_{Generator} \quad (1)$$

For modeling the generator and discriminator, architectures capable of time series inputs are required. For this reason, in this research, the deep convolutional neural networks (DCNNs) (Krizhevsky et al., 2017) have been employed, not least due to their proven efficiency in dealing with time series inputs for GAN training in radial-axial ring rolling (Fahle et al., 2022). The employed DCNN architecture (Lecun et al., 2015) is characterized by a stack of convolutional, pooling and fully connected layers. The convolutional layers apply trainable kernels via discrete convolutions to extract local patterns into a feature map, while pooling layers downsample the data dimensionality with the aim of retaining all necessary information. Following the convolutional layers, fully connected layers can be attached as well and represent the output of the model, thus its final predictions. Utilizing the DCNN for univariate time series predictions (single feature as input) means that each convolutional kernel slides only along the time axis (Ismail Fawaz et al., 2019). For a univariate time series X, the convolution output at time t, denoted as C_t, can be expressed as (Ismail Fawaz et al., 2019):

$$C_t = f\left(\omega * X_{t-\frac{l}{2}:t+\frac{l}{2}} + b\right) \quad (2)$$

where ω is the convolutional kernel (filter) of corresponding length l, b is the bias term, f is the non-linear activation function, and $X_{t-\frac{l}{2}:t+\frac{l}{2}}$ denotes the window of l consecutive times steps centered in t. Applying n multiple kernels, or filters, results in n separate feature maps, that are further processed in the DCNN.

To extend from univariate to multivariate time series (i.e., multiple features per time step), each filter acquires an additional dimension equal to the number of input channels. In other words, in case of dealing with m-features, each kernel will have a (l, m) shape, so that each step convolves across all m-features simultaneously.

After each convolution, a pooling operation (either average or max pooling) is typically applied to reduce the temporal dimension of each feature map, preserving only the most crucial information while also reducing dimensionality by downsampling. In this regard, techniques such as batch-normalization for each channel can be applied for quick convergence during the training process (Ismail Fawaz et al., 2019).

Deep convolutional GAN (DCGAN) model implementation

The implementation of the GAN framework with the DCNN modeling presented in the previous section is obtained using the deep learning library *TensorFlow* with its graph execution (Abadi et al., 2015). The TCRR time series described in Sect. "Tangential cold ring rolling experiments" are pre-processed by applying *Min–Max* normalization and scaling them to 100 timesteps, to ensure constant input shapes. As previously mentioned, the dimension of the multivariate data is equal to six and inherits the following features: time [s], rolling force [kN], rolling moment [Nm], centering force [bar], outer diameter [mm], and the ring thickness [mm] for a total of 129 instances, split into training (70%), validation (15%), and test data (15%). The outer diameter is captured as the metering roll displacement, having a fixed minimal value of 63 mm to maintain comparability between geometries. A conversion remains possible using the thickness and the fixed height of the ring. In the DCGAN architecture, Fig. 5a, the number of hidden blocks in the discriminator was designed to progressively capture local patterns, whereas the generator employs a decreasing number of hidden blocks to produce six time series as output. Each block consists of two 1D convolutional layers, *LeakyReLU* as the non-linear activation function, batch normalization (generator), and dropout (discriminator) while backpropagation is performed using the Adaptive Moment Estimation (Adam) optimizer (Kingma & Ba, 2015). The optimization of both generator and discriminator was initially carried out manually by adjusting the depth of both blocks to allow for stable training. In this regard, maintaining a balanced adversarial game is crucial, as an overly dominant discriminator can hinder the generator's learning process, leading to mode collapse or trivial outputs, whereas a weak discriminator fails to provide meaningful gradients, preventing the generator from improving (Arjovsky & Bottou, 2017). The target is to identify the correct balance between both blocks, as schematized in Fig. 5b, which represents the region where to select the stable model for further optimization.

Following an initial manual tuning, hyperparameters were further optimized via Bayesian optimization using the *Optuna* framework (Akiba et al., 2019). The score function for the optimization is designed by comparing data distributions of the generated data (p_g) and the origin dataset ($D(x)$), to rely not only on the generator and discriminator losses, as schematized in Fig. 5c.

In the DCGAN, temporal dependencies are captured by generating the complete trajectory in a single forward pass rather than predicting it step-by-step. Specifically, the generator maps a latent vector to a full-length multivariate

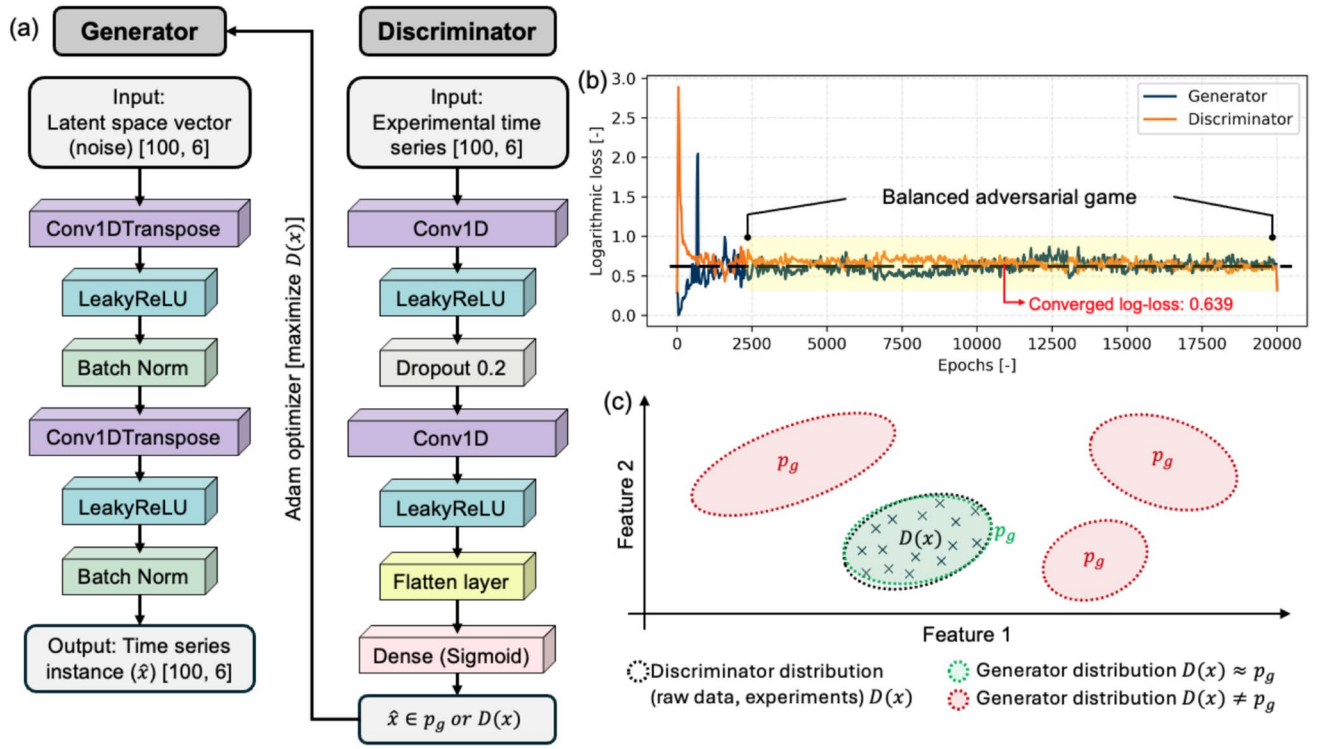


Fig. 5 **a** Implemented DCGAN model framework of the implementation in Python using the library TensorFlow (Abadi et al., 2015) and the class *tensorflow.keras.layers*. **b** Exemplification of the adversary game between generator and discriminator for the identification of the stable training region and **c** resulting variation of the generator features distribution with progressive matching of that of the discriminator

time series of 100 time steps. The temporal structure is learned through the stacked 1D convolutional layers, Fig. 5b, whose increasing receptive field enables the network to represent both local dynamics and longer-range evolution within the rolling sequence. As the convolutional filters operate across all feature channels, the model can also learn coupled temporal relationships between multiple process variables and target quantities.

Observer for the DCGAN model

To improve the selection process for the DCGAN best benchmark model, an additional module was designed in this research to evaluate the synthetic instances constructed by the generator in the optimal region where the $\hat{x} \in D(x)$. This module, subsequently annotated as the *observer*, is based on three elements: the mean curves, the correlations, and dynamic time warping (DTW), here after described in detail. To compare the quality of the generated time series (instances) with respect to the original dataset (TCRR experiments), a mean curve is calculated for each one of the six features and represents the relevant feature trend, as shown in Eq. (3), where n is the number of generated instances and $f(t)$ is a generated synthetic instance.

$$Mean\_curve = \frac{\sum_0^n f(t)}{n} \tag{3}$$

The output dimension of Eq. (3) is a [1, 6] matrix, meaning one row with the mean magnitudes across all predicted instances for 100 timesteps and, as previously mentioned, the number of time steps in each series is set to 100 to uniform the input/output shape. An example of the mean curve within the overall generated instances is shown in Fig. 6a for the case of the normalized radial force.

After the estimation of the mean curves for both generator and discriminator distributions for all six features, the similarity score between the experimental dataset ($D(x)$) and generator distribution (p_g) is calculated as the mean squared error (MSE) between these two domains during the training process and is defined as MC_{score} as in Eq. (4). It ought to be remembered that (4) is the average score across all six features.

$$MC_{score} = \left\| \mathbf{D(x)} - \mathbf{p_g} \right\|_2^2 \tag{4}$$

The second term in the observer score is represented by the features correlation (Pearson, $r_{i,j}$), which ensures that the data interdependencies are well reflected in the synthetic

data as they are in the raw data. For this comparison, the MSE of the dataset correlations and the generator correlations are calculated employing the open-source library *NumPy* (Harris et al., 2020). An example of the features correlation is depicted in Fig. 6b while the relevant score, $Feat\text{-}Corr_{score}$, is estimated as in Eq. (5).

$$Feat - Corr_{Score} = \frac{\sum_{n=0}^{n} \sum_{i,j} (r_{ij}^{D(x)} - r_{ij}^{p_g})^2}{n} \tag{5}$$

The last contribution in the observer score is based on the dynamic time warping (DTW) distance, a method for identifying and evaluating patterns within two mutually distorted time series. In this regard, mutually distorted time series refers to the comparison between true and generated data, carried out by nonlinearly stretching or compressing segments so that corresponding features are optimally matched. The lower the required adjustments the better the DTW score. From a practical standpoint, the first step is represented by the calculation of the smallest Euclidean distance d between two generic points i and j belonging to two distinct time series, A and B, in Eq. (6) (Keogh & Pazzani, 2000), as shown in Fig. 7.

$$DTW_{score} = min\left\{ \frac{\sqrt{\sum_{k=1}^{K} d(i_k, j_k)}}{K} \right\} \tag{6}$$

This is followed by the establishment of the warping path describing the optimal alignment of the individual time points of the time series, with consideration for the following constraints (Keogh & Pazzani, 2000):

1. *Boundary conditions*: The distances $d_0(0,0)$ and $d_k(m,n)$ describe the start and end point of the path. The variable k describes the number of distances determined on the path. The variables m and n describe the last time point of the time series A and B respectively.
2. *Monotonicity condition*: $i_{k-1} \leq i_k$ und $j_{k-1} \leq j_k$ ensures, that the time points within the path are in the correct chronological order.
3. *Step size condition*: $(i_k - i_{k-1}, j_k - j_{k-1} \in \{(1,1),(1,0),(0,1)\}$ prevents individual temporal steps from being skipped.

In this research, the implementation of the DTW is achieved with the library DTAIDistance (Meert et al., 2020) and is partly based for the visualization on (Stent, 2024). An example of the application of the DTW is shown in Fig. 7a, where two time series, namely A and B, are shifted of an offset of 0.2 from one another, resulting in the shortest path calculation and point to point comparison shown in Fig. 7b and c, respectively. The combination of the three scores presented so far, and relevant to the three elements in the observer module, are summed together, as in Eq. (7), and employed for the identification of the best DCGAN model. In this regard, it ought to be remembered that the stability of the GAN is evaluated based on the discriminator loss, as shown in Fig. 5b and described in Sect. "Deep convolutional GAN (DCGAN) model implementation", while the best model is selected based on the score of Eq. (7).

$$score_{observer} = MC_{score} + Feat - Corr_{score} + DTW_{score} \tag{7}$$

As a final remark, in the DCGAN setting, the observer module remains unconditioned, is detached from the training loop, and serves solely as an external evaluator, guiding

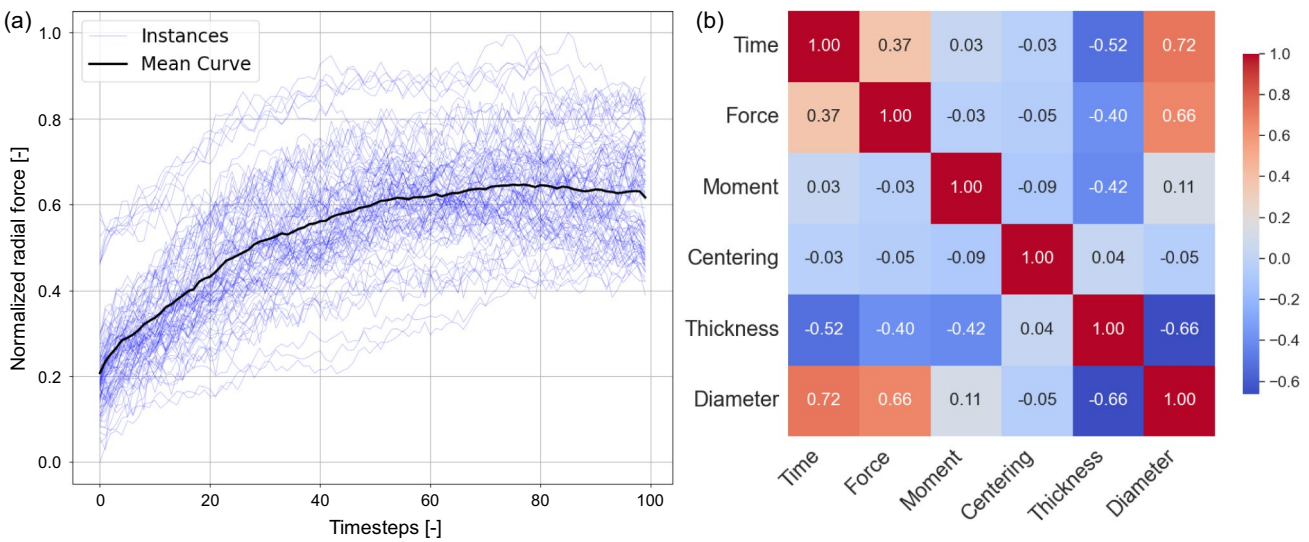


Fig. 6 **a** Mean curves for the rolling force and **b** Pearson correlation matrix, both training data

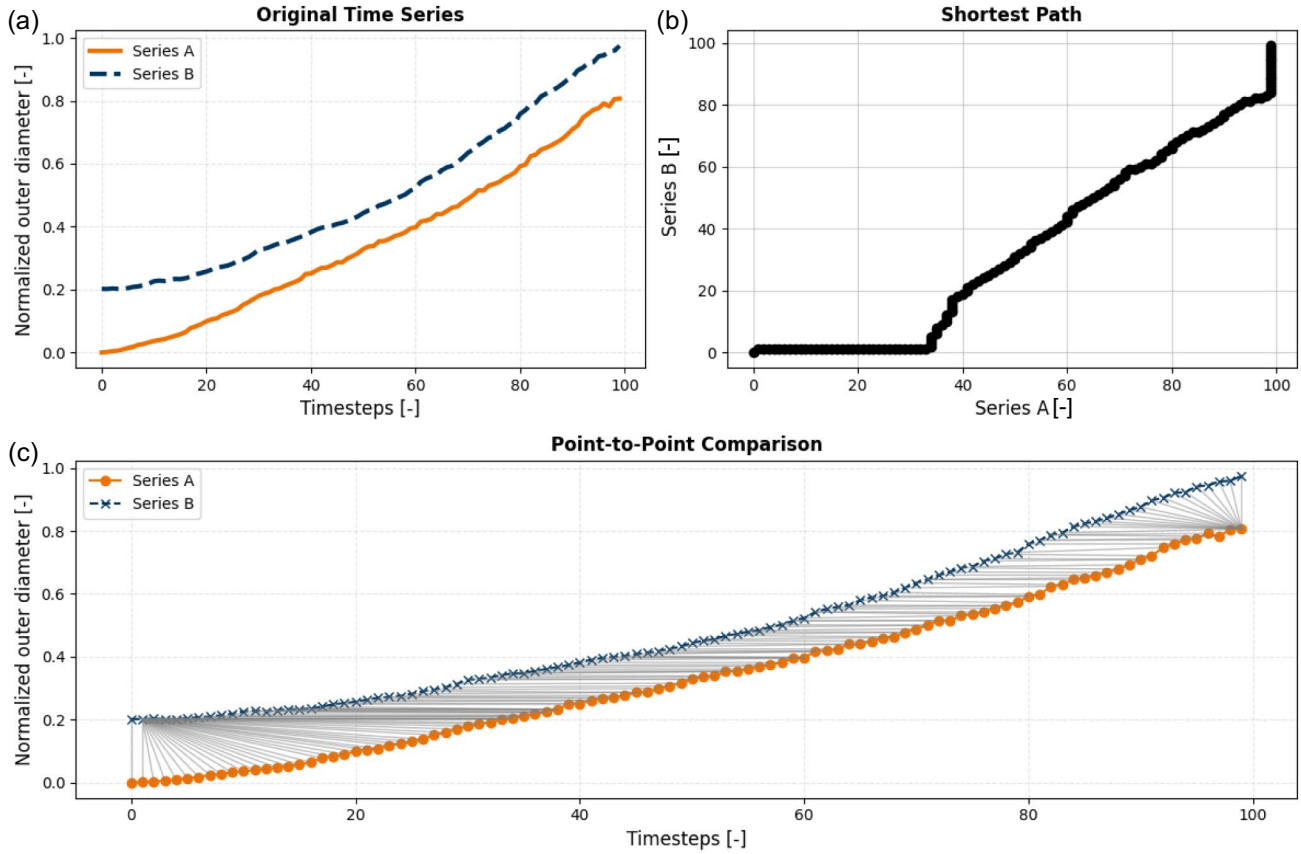


Fig. 7 DTW comparison of an instance of the outer diameter to a second instance with an offset of 0.2. Visualization framework adjusted from (Stent, 2024)

the model selection by quantifying the alignment between generated and real data distributions without contributing to gradient updates.

Physics-informed and analytical transfer learning loss modeling

To improve the physical adherence of the data generated, an extension of the GAN framework is introduced in this research, as here after described. First, the improvement of the GAN architecture aims at including physical knowledge relevant to the ring rolling process and metal forming in general in the data generation pipeline and it is based on the well-known concept of physics-informed neural networks (PINNs).

PINNs, developed by Raissi et al. in (2019), incorporate physical loss derived by partial differential equations (PDE) to either provide a data-driven solution for the PDE at hand, or to discover unknown parameters in the PDE. This setup has been applied to continuous and discrete time models, for instance for the Navier–Stokes and Schrodinger equation (Raissi et al., 2019). In this regard, and returning to the motivation of the presented study, providing a physical adhered model for TCRR, by integrating a PINN-Scheme in the GAN-framework, is one of the key novelty aspects of this research, especially when considering the nature of the process involved.

Related work on physics-guided GAN modeling

Physics-Informed GANs (PI-GANs) have recently been proposed (Yang et al., 2020) to incorporate known physical laws directly into the adversarial framework. In a PI-GAN, the generator not only seeks to produce realistic data but also outputs quantities that must satisfy a given set of partial differential equations (PDEs). To enforce these constraints, the outputs of the generator are passed through a differentiable neural network, such as the one of the PINNs architecture, and the relevant residuals are added as an additional term in the loss. Therefore, during the training, the discriminator is presented with two sets of data. The former being based on real data and aimed at mimicking their behavior, the latter is composed of physically bounded rules, defined by the constraints included in the PINN architecture. Hence, the

discriminator checks the synthetically generated data for their similarity in features distribution, like in a standard GAN, but also for their consistency with respect to the physical constraints defined for the process at hand.

In this regard, a recent study by Yonekura (2024) showed the possibility for integrating knowledge without the need for including gradients defined as physics-guided GAN, or PG-GAN. This approach enables the integration of any kind of user-defined evaluation, even through black box or surrogate models that do not necessarily need to be differentiable and can be represented by commercial simulation software tools. In this case, the compliance factor is calculated outside of the training processes for the binary labeling of the generated data to train the discriminator and therefore subsequently to the physics- or rule-guided optimization of the generator. As hypothesized in Yonekura (2024), since the quality of the synthetically generated data is based solely on the user constraints, no dataset for training is needed, since the discriminator is represented through the relevant model. Consequently, a datapoint $\hat{x}$ from the generator distribution p_g is evaluated by any chosen physical model and classified as real or fake depending on a threshold error ε. Therefore, the classification in Real (R) and False (F) is defined as in Eq. (8). It should be noted that since R_ε for early epochs is equal to zero, pre-training based on a conventional GAN scheme is required.

$$R_\varepsilon = \{\hat{x}|P(\hat{x}|\theta) \leq \varepsilon\}, F_\varepsilon = \{\hat{x}|P(\hat{x}|\theta) > \varepsilon\}. \tag{8}$$

In addition to the PG-GAN, Yonekura (2024) also suggested the possibility of combining the PG-GAN and the PI-GAN to a PG-PI-GAN with an additional loss for the generator as in (9), defined in terms of the constant λ and the residual r.

$$\underset{G}{^{Min}} E_{G(z)\in F_\varepsilon}\left[log(1 - D(G(z)))\right] + \lambda r, \tag{9}$$

The fully PINN-driven networks, as suggested in (Yang et al., 2020), though advantageous from the data gathering standpoint, shift the focus towards analytical-like models that already showed several limitations in metal forming scenarios when compared to either finite element or machine learning solutions (Cao et al., 2024). To this end, the following sub-section addresses this gap by detailing the PG-A-GAN model, adapted from Yonekura (2024), proposed in this research.

Physics-guided auxiliary GAN (PG-A-GAN)

Considering the background summarized so far, the methodology proposed in this research incorporates, besides conventional generator loss, a second regularizing loss term defined as an auxiliary physical loss. Accordingly, the proposed model is defined as physics-guided auxiliary GAN (PG-A-GAN). The analytical loss is included in the conventional generator loss, previously shown in Eq. (1), as a summation term multiplied by the factor λ, as shown in Eq. (10).

$$L_{Gen} = \mathbb{E}_{z\sim p_z(z)}\left[\log\left(1 - D(G(z))\right)\right] + \lambda p_r \tag{10}$$

In comparison to Yonekura (2024), the discriminator of the proposed PG-A-GAN remains data-driven and is optimized by the dataset $p_{data}(x)$, as in Eq. (1) while the physical models are accounted for at residual and loss levels and thus must be differentiable. Since the physical models are only evaluating the generated data samples $\hat{x}$, the models do not necessarily have to be a partial differential equation of x, but can be also represented by analytical equations or machine learning surrogate models, as also considered in this research.

In this regard, and as known from a prior study on PINNs modeling in GANs, such as Yonekura (2024), early epochs show high volatility of the loss due to the high residuals associated with the physical constraints. To overcome this challenge, the annealing factor λ was added and optimized during training, enabling a smoother transition from early epochs to a more stable convergency of the GAN model. The overall implementation of the PG-A-GAN is depicted in Fig. 8, the different components of the auxiliary loss are summarized in Table 2 while Appendix A details the pseudo-code to illustrate the relationships among the components of the PG-A-GAN.

To enforce physical consistency throughout the generated trajectories, the physics-based constraints are evaluated directly on the generated sequence at each normalized time index t. Operatively, this means that, for every generated sample, the corresponding residuals are computed along the full length of the normalized time and then aggregated across time, and the considered batch, using a MSE formulation. This per-timestep evaluation ensures that the generator is penalized for local violations occurring at any point in the rolling trajectory, rather than only matching global trends, thereby promoting temporally consistent and physically plausible sequences.

The physical auxiliary loss combines a physics-informed component based on established analytical relationships and an analytical transfer learning (ATL) component derived from surrogate modelling. The main difference between these terms lies in their physical representation. While the physics-informed loss enforces relationships that can be expressed directly in analytical form, ATL introduces learned adjustments modeled by surrogate models to reduce systematic mismatch with the target domain. The individual constraint terms are detailed in Sect. "Physics-informed modeling" (physics-informed) and "Analytical transfer learning" (ATL).

Physics-informed modeling

As introduced in the previous sub-section, the synthetic data generation in the proposed PG-A-GAN is controlled by both physics-informed and analytical transfer learning concepts. In this sense, the physics-informed part utilizes established ring rolling equations but does not implement them as differential equations as is done in conventional PINNs (Raissi et al., 2019). The PI-part in this work uses the following equations (Eqs. 11–15) to directly calculate the physical generator loss (Eq. 10).

The volume conservation principle is employed as it is regarded as a basic, yet informative metric for evaluating the quality of predictions in several metal forming processes. In this regard, the volume conservation takes into account the geometrical expansion of the ring calculating for each step its volume V based on the rectangular ring cross section with its width w:

$$V = \left(\pi \cdot R^2 - \pi \cdot r^2\right) \cdot w \tag{11}$$

to address the volume conservation:

$$\Delta V = 0. \tag{12}$$

Since small deviations of Eq. (12) do exist in the experimental dataset, in terms of magnitude of a few mm^3, the loss backpropagated for the volume conservation is the averaged MSE of the subtraction of the average volume of the dataset to the averaged generators instances.

To model the ring expansion over time, the ring diameter can be calculated incrementally based on the number of revolutions (Berti et al., 2015). For the TCRR, the width is a

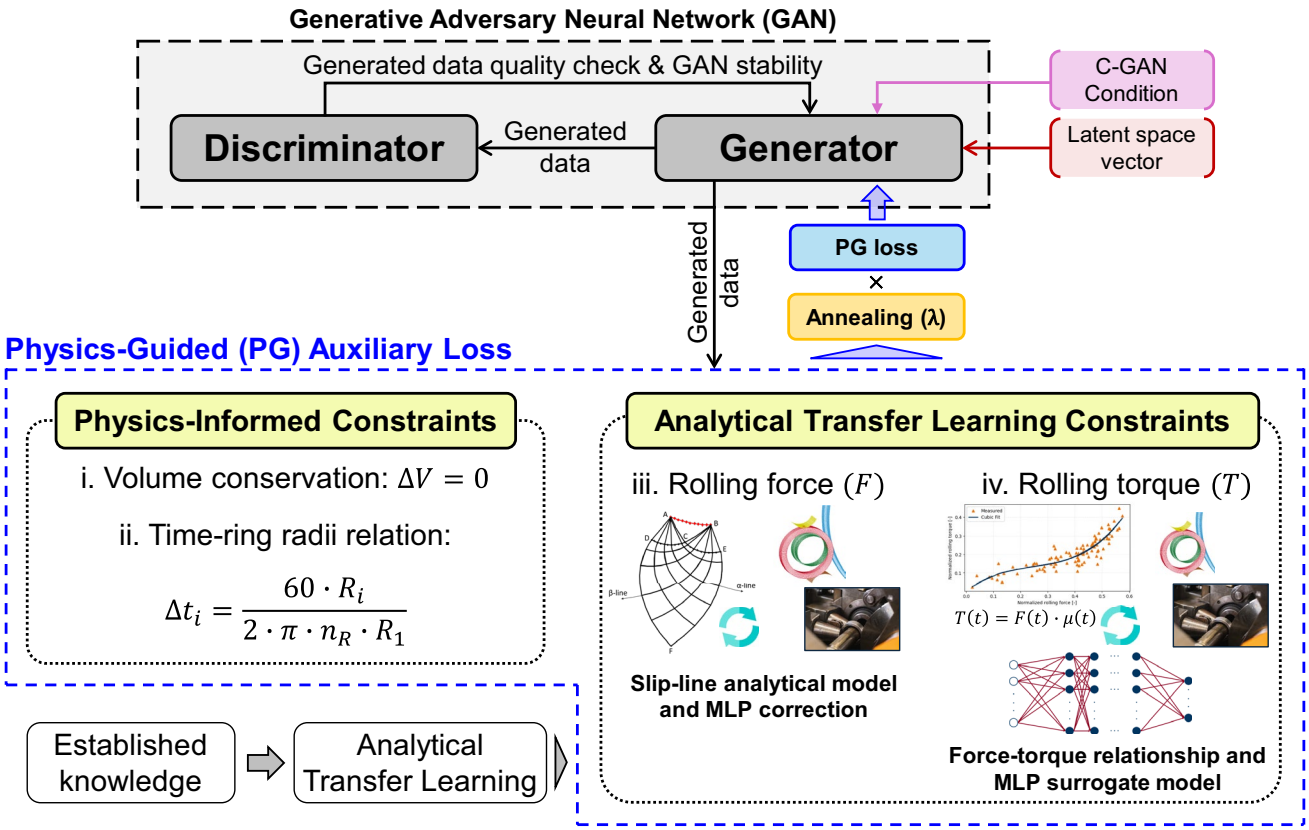


Fig. 8 Modification of a GAN formation by addition of physics-informed and analytical transfer learning based auxiliary loss, with relevant interaction in the backpropagation through annealing function

Table 2 Definition of the physical modeling and its controlled parameters

Modeling	Model type	Parameters controlled
Volume conservation	Analytical	Diameter, thickness
Ring expansion over time	Semi-analytical	Time
Slip-line model	Analytical + machine learning [MLP]	Rolling force
Surrogate	Machine learning [MLP]	Rolling torque

fixed parameter with minor deviations in the 1% magnitude, thus it can be considered as a constant. Moreover, considering that the starting geometry of the ring represented by its initial thickness, outer radius, and inner radius s_0, R_0, r_0, is known *a-priori*, the time required for the next ring revolution is computed using Eq. (13), where i denotes the i-th rotation of the process and R_i and r_i represent the outer and inner radii of the ring at this stage, respectively. Accordingly, the corresponding thickness in next ring rotation can be estimated from Eq. (14), allowing to update the ring geometry together with Eq. (15).

$$\Delta_{t_i} = \frac{60 \cdot R_i}{2 \cdot \pi \cdot n_R \cdot R_1} \tag{13}$$

$$s_{i+1} = s_i - v \cdot \Delta t_i = R_{i+1} - r_{i+1} \tag{14}$$

$$\left(\pi \cdot R_{i+1}^2 - \pi \cdot r_{i+1}^2\right) \cdot w = V \tag{15}$$

Equations (13)–(15) are used to calculate the time required to reach the final diameter. In other words, the summation of Eq. (13) is performed by incrementally solving the linear equation system of Eqs. (14) and (15) to calculate the required process time for forging the ring analytically. Implementing this analytical scheme and comparing the results with the specific dataset (p_{data}) yields an RMSE of 0.83 s for this analytical modelling approach. This 0.83 s error in the analytical description will be used in Chapter 4 to penalize deviations greater than the model accuracy.

Analytical transfer learning

As concerns the latter set of models for the auxiliary loss in the PG-A-GAN, the implementation of analytical transfer learning for rolling force and torque modeling is first presented in Fig. 9.

Regarding the rolling force, its estimation is based on a fully analytical model, based on the slip line field developing between the mandrel and the main roll, as proposed by (Quagliato et al., 2018). However, considering that this slip line model was developed for hot radial-axial ring rolling (RARR), where the change of both flow stress and contact arc can be regarded as negligible throughout the process, surrogate modeling based on the FE and experimental results was employed. First, the baseline rolling force slip line model (Quagliato et al., 2018), based on the original solution by Hawkyard et al. (1973), requires, as inputs, the ring wall thickness change Δs, ring height h, contact length L_C, shear stress k, and pressure factor γ, as per Eq. (16), where L_C and γ are calculated as in (17)–(18).

$$F = 2k \cdot \gamma \cdot h \cdot L_C \tag{16}$$

$$L_c = \sqrt{\frac{2 \cdot \Delta s}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R} - \frac{1}{r}}} \tag{17}$$

$$\gamma = -0.0098 \cdot \left(\frac{s}{L_C}\right)^2 + 0.2999 \cdot \left(\frac{s}{L_C}\right) + 0.690 \tag{18}$$

Considering Eq. (16) withing the frame of the PG-A-GAN modeling for the rolling force, h is a constant for the TCRR process, whileΔs, needed for both L_C and γ, can be derived from the generated instances. However, the RARR assumptions forL_C,γ, and k are not valid *a-priori* for the TCRR and, for this reason, the analytical framework of (16)–(18) cannot be applied directly in its current state. To this end, considering the different nature of the plastic flow between hot RARR and the cold forging of TCRR, the contact arc and the shear stress were adapted from the analytical solutions by employing the results of a single FE simulation, as shown in Fig. 10, as well as the experimental instances summarized in Sect. "Material characterization".

The adaptation from the analytical frame of (16)–(18) is carried out by training a multi-layer perceptron (MLP) surrogate model that scales Eq. (16) to match the experimental instances for the rolling force, as shown in Eq. (19). The MLP surrogate model training and relation with the analytical slip line force model is schematized in Fig. 9. In this regard, the MLP surrogate model for the rolling force is trained to predict the best scaling factors $\alpha, \beta, \gamma, \delta$ based on the initial and final geometries of the considered ring, the process time (t), the mandrel feed speed, and centering rolls pressure. The α factor scales the slip line model (Eq. 16), while the β and γ factors account for linear and quadratic temporal relationships, respectively. The δ factor reflects a scalar offset.

$$F_{ATL} = \alpha\left(2k \cdot \gamma \cdot h \cdot L_C\right) + \beta t^2 + \gamma t + \delta \tag{19}$$

Therefore, due to the data-based adaptation of an existing analytical model to the real-world dataset, the rolling force surrogate modeling is defined as *analytical transfer learning* (ATL). Since the MLP surrogate model strategy is the same for both rolling force and torque, the relevant details are provided in Sect. "DCGAN (SOTA) vs PG-A-GAN benchmark analysis".

Once the rolling force is corrected through ATL, as presented in the previous paragraphs, the rolling torque can be described by the relationship between the force and the moment, depending on a time dependent friction coefficient $\mu(t)$ as in Eq. (20), where R is the outer radius of the main roll.

$$T(t) = F(t) \cdot \mu(t) * R \tag{20}$$

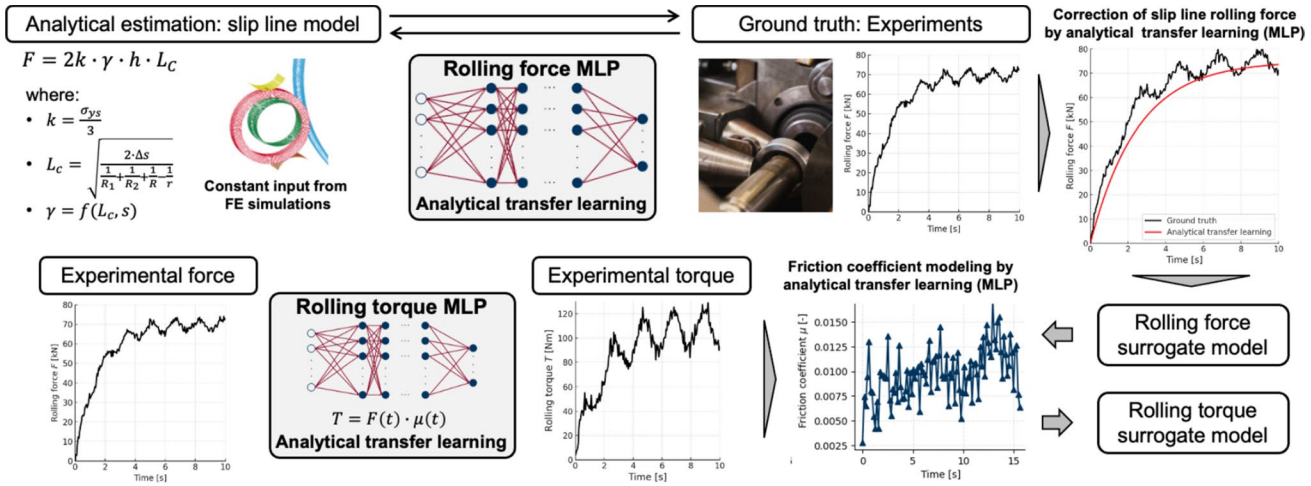


Fig. 9 Implementation of the analytical transfer learning for the correction of the analytical estimation of the rolling force by slip line model and for the time dependent friction correlation for the rolling torque modeling

The dependency between the *Min–Max* normalized rolling force and rolling torque, based on the experimental instances of Sect. "Tangential cold ring rolling experiments", is presented in Fig. 11a, and is fitted with a 3rd order polynomial function, resulting in the time-dependent friction coefficient presented in Fig. 11b. Similarly to what was carried out for the rolling force surrogate model, the analytical transfer learning model for the rolling torque, in dependency of the rolling force, is defined as in Eq. (21), based on four trainable parameters $(\epsilon, \theta, \varphi, \omega)$ defining a 3rd order polynomial function. These four coefficients are optimized during the MLP surrogate model training to minimize the deviations with respect to the experimental instances summarized in Sect. "Material characterization".

$$T_{ATL} = \epsilon F_{ATL}^3 + \theta F_{ATL}^2 + \varphi t + \omega. \tag{21}$$

Benchmark, physics-guided, and conditional implementations

The aim of this last section before the results chapter is trifold. First, in Sect. "Analytical transfer learning models training and performance", the two modeling techniques presented in Sects. "Deep convolutional GAN (DCGAN) model implementation" and "Physics-informed and analytical transfer learning loss modeling" are summarized to provide a clear background for the understanding of the results of the benchmark analysis. Second, the training and performance of the two ATL models, as presented in Sect. "Analytical transfer learning", is summarized in Sect. "DCGAN (SOTA) vs PG-A-GAN benchmark analysis". Third, insights into the conditions applied to the PG-A-GAN for the generation of synthetic data, and the relevant modeling implications, are

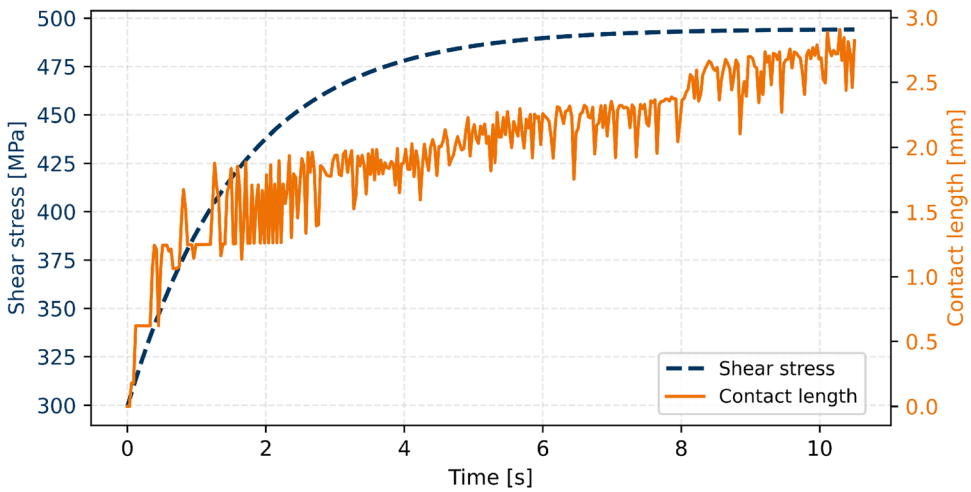


Fig. 10 FE-based shear stress and contact length curves used as a starting point for the adaptation of the slip line rolling force model during the analytical transfer learning scheme

detailed in Sect. "Conditional synthetic data generation: the AC-PG-A-GAN approach".

Analytical transfer learning models training and performance

As mentioned in Sect. "Analytical transfer learning", to correct the analytical estimation of the rolling force and define the time-dependent friction coefficient relationship between the rolling force and torque, two machine learning surrogate models were introduced. The surrogate models, one for each of the two tasks above, is based on a multi-layer perceptron (MLP) network and are trained to learn, and correct, the offset between the analytical predictions and the experimental ground truths, based on the process parameters and ring geometry. The structure of the two surrogate models is presented in Fig. 12a, for the rolling force model, and Fig. 12b, for the rolling torque model, respectively. As a reminder, the machine input shown as the starting block for both MLP networks is a six components vector containing process time [s], outer diameter (start and end) [mm], ring thickness (start and end) [mm], and the feed speed [mm/s].

During the training of the MLP surrogate models for rolling force and torque, it was observed that limited generalization was the main bottleneck when using shallow network architectures. To address the generalization, a manual hyperparameter grid search was performed, focusing on combinations of dropout rates and L_2-regularization strengths, alongside variations in the number of hidden layers and units per layer. The search space and tested parameter ranges are summarized in Fig. 12c while the best-performing configuration is shown in Fig. 12d. This optimal setup consisted of four hidden layers with progressively decreasing units per layer [32, 16, 8, 3], small dropout rates of [0.01, 0.01, 0.01], and L_2-regularization strengths of 0.1 for the force model and 0.01 for the torque model, respectively. As shown in Fig. 12a-b, the shear stress and contact length trends estimated by FE simulations for the force model, and friction coefficient for the torque model are fed into the analytical equations during training, enabling backpropagation of the residual error between the corrected analytical prediction and the experimental ground-truth data. This coupling ensures that the MLP correction factors, $(\alpha, \beta, \gamma, \delta)$ for the rolling force model and $(\epsilon, \theta, \varphi, \omega)$ for the rolling torque model, are optimized to bridge the gap between the analytical model and experimental or FE-derived observations. The ATL models achieved root mean square errors (RMSE) of 4.825 kN for the force model and 10.144 Nm for the torque model, evaluated on the test set comprising 15% of the total dataset. These results demonstrate the capability of the model to capture the overall evolution of force and torque over time, despite the complexity introduced by multiple parameters and strong time dependence. In this regard, Fig. 13 shows the performance of two test cases with different magnitudes and process durations.

In both cases, the ATL models closely follow the average trend of the experimental measurements, successfully reproducing the general course of the signals. However, short-term fluctuations and high-frequency noise present in the sensor data are only partially captured, indicating that the models prioritize robust trend prediction over replicating instantaneous volatility.

DCGAN (SOTA) vs PG-A-GAN benchmark analysis

The first analysis presented in the results section (Sect. "Results on SOTA and physics-guided GANs") is relevant to the comparison between the conventional DCGAN with the custom-made observer, as detailed in Sects. "Deep convolutional GAN (DCGAN) model implementation" and "Observer for the DCGAN model", and the PG-A-GAN, with the physics-guided auxiliary loss defined on physics-informed and analytical transfer learning terms, as per Sects. "Physics-informed modeling" and "Analytical transfer learning". The results include first the model structure optimization, hyperparameters optimization, and performance

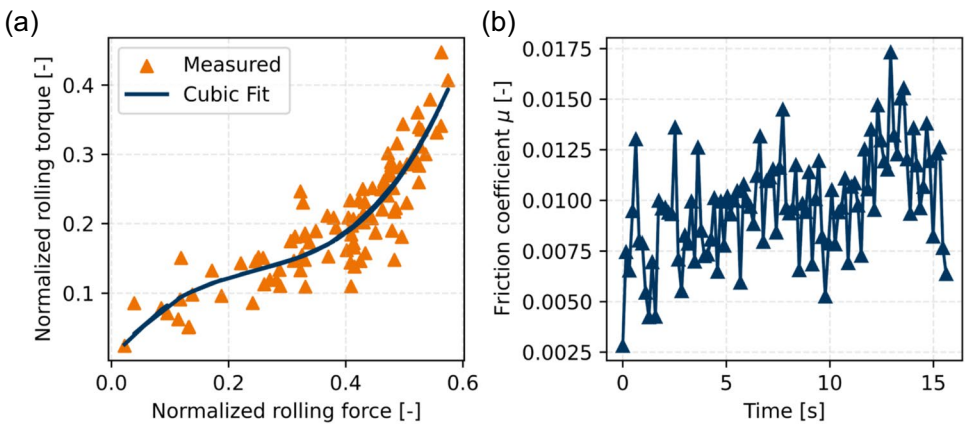


Fig. 11 Rolling torque over the rolling force for one ring rolling instance. The torque in dependency of the force for this example rolling instance is defined to $T = 5.2633x^3 + -3.8657x^2 + 1.1655x + 0.0017$

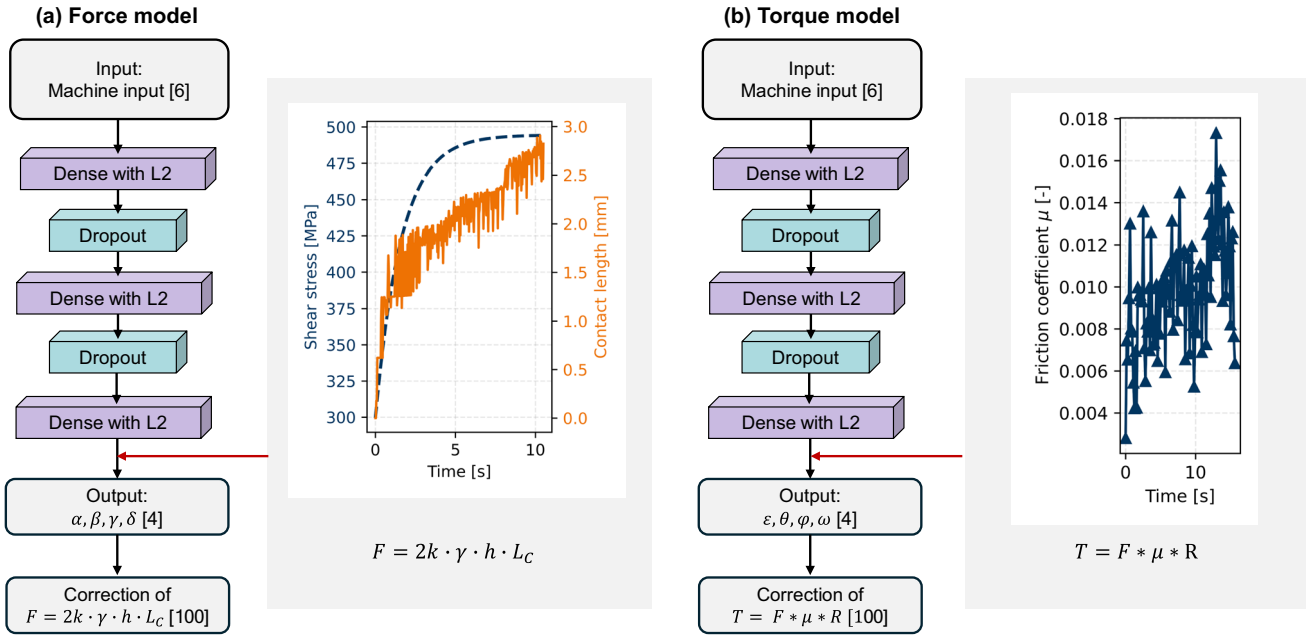
in generating synthetic instances without any conditions applied.

Conditional synthetic data generation: the AC-PG-A-GAN approach

An extension of the classic GAN is the ACGAN (Odena et al., 2017), which introduces one or more class conditions to guide the generation of specific outputs. This is achieved by adding an auxiliary decoder network to the discriminator, enabling it to distinguish real from generated data while also to predict the corresponding class label. For instance, the TCRR process considered in this study, such conditions can be represented by the final ring geometries to be generated. In the ACGAN framework, the generator receives two inputs: the class label and the noise vector. The discriminator, equipped with the auxiliary decoder, outputs both the probability of data authenticity and a prediction of the class label. Consequently, the objective function of Eq. (1) is extended by an additional auxiliary term, namely the logarithmic class label loss.

In terms of the model proposed in this research, the PG-A-GAN is extended by introducing the condition of a desired final ring geometry, leading to the definition of the AC-PG-A-GAN, where AC stands for the auxiliary condition. This allows the generation of specific rolling instances aligned with target geometries. The conditioning follows the backpropagation principle of ACGAN (Odena et al., 2017), but within the physics-guided framework of the PG-A-GAN, thereby embedding physical constraints while generating condition-specific outcomes. The effectiveness of the proposed AC-PG-A-GAN is evaluated with respect to three main aspects:

1) **Variance retention**: Does the model preserve stochasticity, *id est*, does the same condition input yield different outputs, thereby reflecting machine-induced variance?
2) **Conditional sensitivity**: Does the model respond appropriately to different target ring geometries across all generated parameters?
3) **Deviation from reference data**: How great is the deviation of the average curves for different conditions of the AC-PG-A-GAN in comparison to the TCRR-dataset?



(c) Hyperparameters grid search ranges (MLP)

Hyperparameter	Range
Layers	[1, 2, 3, 4, 5, 6]
Units	[3, 8, 16, 32, 64]
Dropout	[0, 0.1, 0.2, 0.3, 0.4]
L_2-Regularization	[0, 0.1, 0.01]

(d) Best hyperparameters configuration (MLP)

Hyperparameter	Values
Layers	4
Units	[32, 16, 8, 3]
Dropout	[0.01, 0.01, 0.01]
L_2-Regularization	0.1 (Force), 0.01 (Torque)

Fig. 12 Model structures for analytical transfer learning (ATL) based on MLPs for **a** rolling force and **b** rolling torque. **c** Ranges considered during the grid search analysis and (d) best hyperparameters for the MLP surrogate models

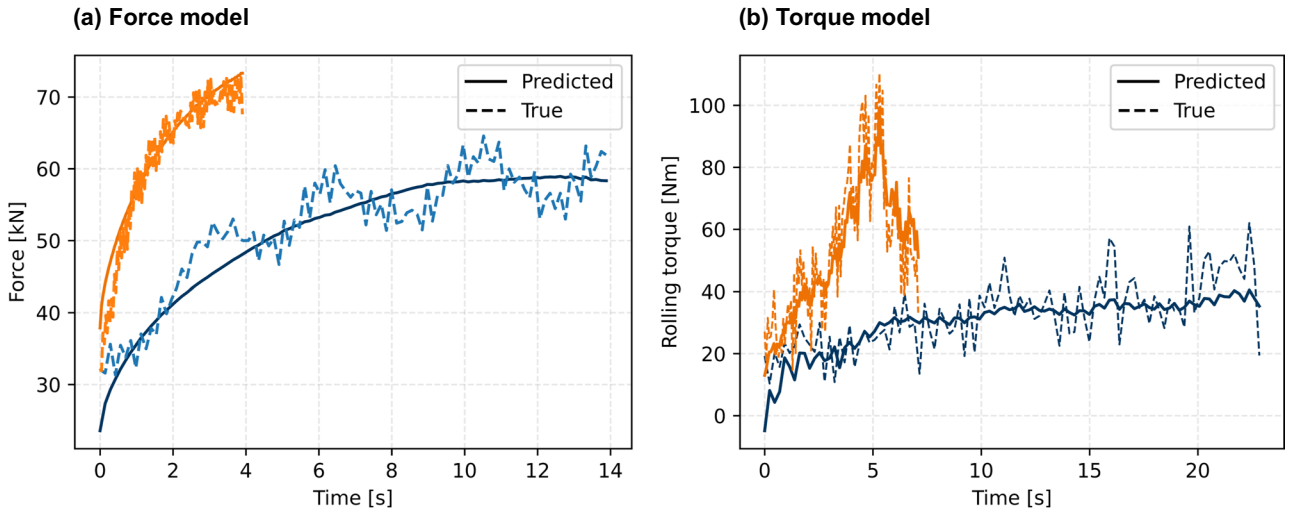


Fig. 13 Prediction instances for the force modeling (**a**) and the torque modeling (**b**). The ground truth (True) is plotted with a dotted line, whilst the predictions by the analytical transfer learned model (Predicted) have a solid line

In summary, the AC-PG-A-GAN leverages a physics-driven modeling strategy that not only respects physical boundary conditions but also enables the controlled generation of rolling outcomes defined by target ring geometries or, in principle, other user-defined class labels.

Results on SOTA and physics-guided GANs

This chapter is organized into three sections. The first presents the performance of the state-of-the-art (SOTA) DCGAN, integrated with the custom-developed observer, as described in Sects. "Deep convolutional GAN (DCGAN) model implementation" and "Observer for the DCGAN model". The second section reports the results of the proposed PG-A-GAN, introduced in Sect. "Physics-informed and analytical transfer learning loss modeling", which is further extended under specific conditions to form the AC-PG-A-GAN, as addressed in Sect. "Conditional synthetic data generation: the AC-PG-A-GAN approach". The results are presented independently, while their interpretation and comparative discussion are deferred to the next chapter.

DCGAN with custom-observer synthetic data generation (benchmark model)

The depth of the generator and discriminator in the DCGAN must be designed correctly to enable stable training. Therefore, at first, a manual hyperparameter tuning was conducted to derive stable configurations, followed by a fine-tuning of the hyperparameters using Bayesian optimization. In this regard, stable training is characterized by the generator and discriminator converging to a similar value for the logarithmic loss of Eq. (1), as shown in Fig. 5b where both modules converge to a value of 0.693, proving the success of the adversarial game. During the initial manual hyperparameter tuning, it was found that the generator required greater depth, *id est* more blocks; see Fig. 5, to ensure stable training. The additional convolutional blocks increased the representational capacity of the generator, enabling it to progressively refine the synthetic signals and better approximate the underlying data distribution. Additional configurations with more blocks were also tested but yielded no further improvements to the training process.

Based on these initial tuning results, Bayesian optimization through the *Optuna* framework (Akiba et al., 2019) was applied by employing the search space reported in Table 3, where also the identified best combination of hyperparameters is reported. In this regard, a total of 100 runs were carried out and the observer score of Eq. (7) was employed as the evaluation criteria to identify the best combination. To ensure comparability, this configuration

Table 3 Search space for hyperparameter tuning and the best configuration. * Applies to all layers

Parameter	Search space	Best configuration
Latent dimension	[8, 16, 32]	16
Batch size	[8, 16, 32]	32
Kernels generator*	(2, 11)	[8, 11, 11, 5]
Kernels discriminator*	(2,11)	[8, 2, 8]
Filters generator*	(0, 512)	[128, 67, 45, 33]
Filters discriminator*	(8, 256)	[16, 75, 95]

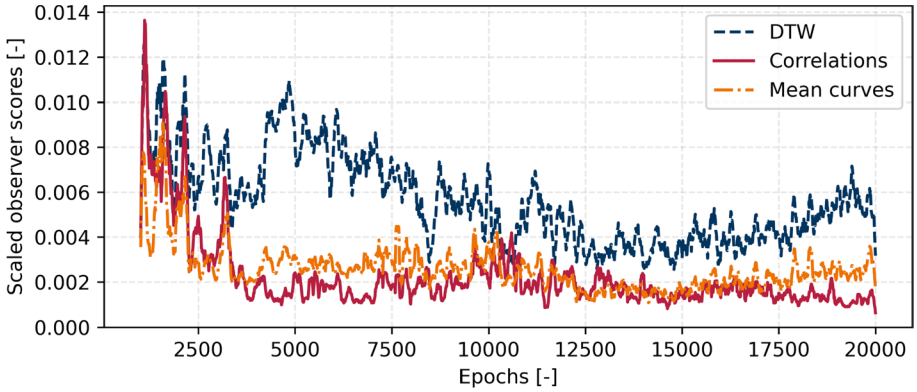


Fig. 14 Scaled observer score curves with respect to the mean curves over the training epochs. The curves are smoothed with a moving average filter with a window size of 20

has been applied to all modelling approaches (DCGAN, PG-A-GAN and AC-PG-A-GAN) throughout this research. In this regard, an example of the evolution of the scaled observer scores for the totality of 20,000 epochs of the training, and for all the three components of the observer score of Eq. (7), is reported in Fig. 14.

Referring to Fig. 14, the overall trend in the three scores is similar, with convergence occurring at similar epochs. However, a more detailed perspective reveals differences in the scores' evolutions throughout the epochs, thus the best model should be chosen based on the combined score, rather than weighting the three components separately. Considering the best configuration for the DCGAN, selected according to the observer score and reported in Table 3, an example of 10 synthetic instances generated with the optimized model is reported in Fig. 15 alongside the average curves for both the real and synthetic data are presented.

As presented in this section, the DCGAN offers a valuable and reliable framework for the generation of synthetic data samples. Moreover, the addition of the developed observer unit, through its evaluation score, allowed for the identification of a robust model, matching well with the experimental instances, as shown in Fig. 15, where the average curves for synthetic and real (experimental) data are well aligned for all six features. However, though effective, the SOTA DCGAN configuration does not allow for the monitoring of the physical adherence of the generated instances to the real dataset, nor does it allow to account for this variation during the generation process. To address this limitation, the proposed PG-A-GAN model was defined, as presented in Sect. "Physics-informed and analytical transfer learning loss modeling", and aimed

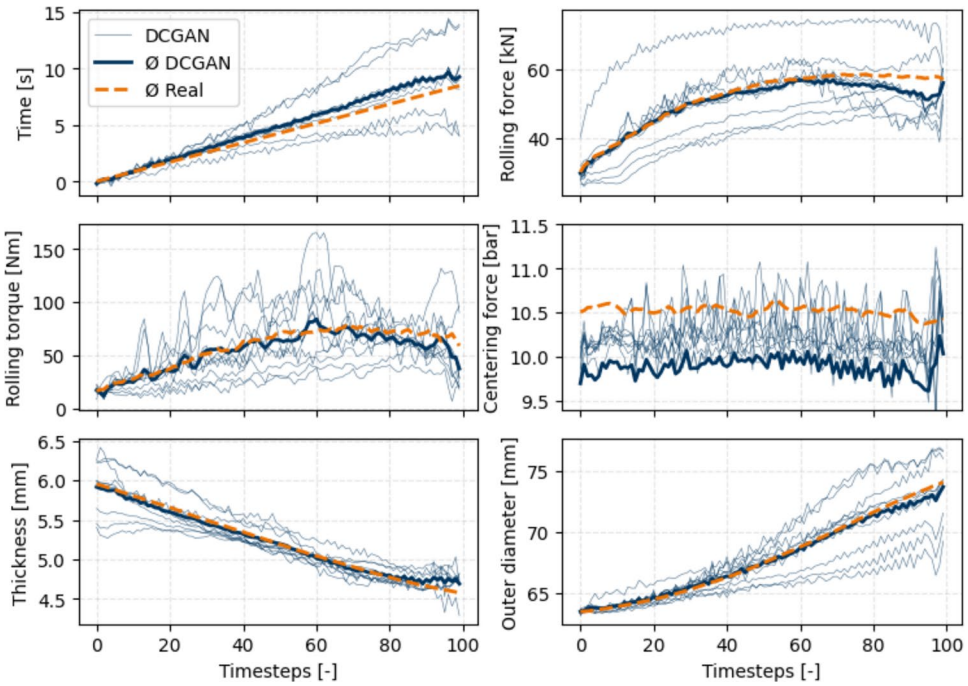


Fig. 15 Ten instance time series for all features of the baseline DCGAN with respect to the experimental average curve

to be employed for the generation of synthetic instances under the same conditions, to be subsequently benchmarked against the DCGAN model, to prove its superiority. The results of this comparison analysis are reported in the following section and commented upon in detail.

PG-A-GAN: synthetic data generation

The physical equations derived from the PG-A-GAN scheme are implemented based on their level of accuracy. This involves the subtracting of the predicted value of the physical equation from the overall model accuracy (RMSE) and backpropagating the square of this difference. Backpropagation is designed with annealing so that maximum amplification is reached after 5000 epochs. In other words, up to 5000 epochs, higher priority is given to the standard generator-discriminator adversarial game while after that the priority shits towards the generation of physics-compliant instances.

To examine the effect of backpropagation, the losses for four different model training are analyzed. The losses cover the three observer scores discussed earlier, as well as alignment with the four introduced physical models. To exclude outliers, only the 75th percentile was considered, and the curves are smoothed with a moving average filter with a window size of 10. Based on these curves, the average for each loss is calculated to compare the baseline DCGAN approach with the PG-A-GAN approach. To enable a direct comparison, the relative percentage improvement or worsening is calculated and presented in Fig. 16. The relative comparison reveals an improvement in all observed scores, except for the torque, which worsened by 3%. In particular, the rolling force and volume conservation show significant improvement of over 100%. As an additional metric of comparison, Fig. 17a shows the propagation of the loss for the observer scores, Sect. "Observer for the DCGAN model", while Fig. 17b

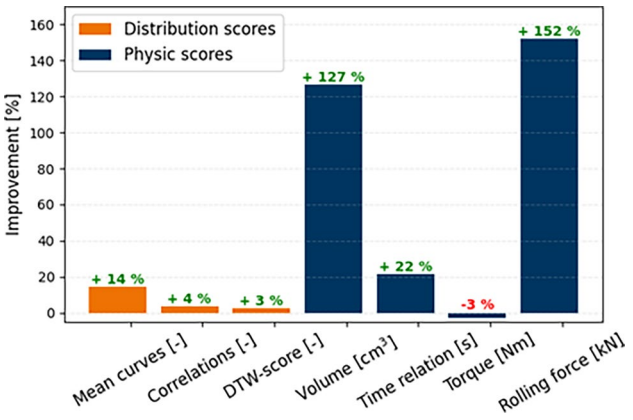


Fig. 16 Comparison of the relative percentage improvement of the PG-A-GAN to the DCGAN for 20,000 epochs

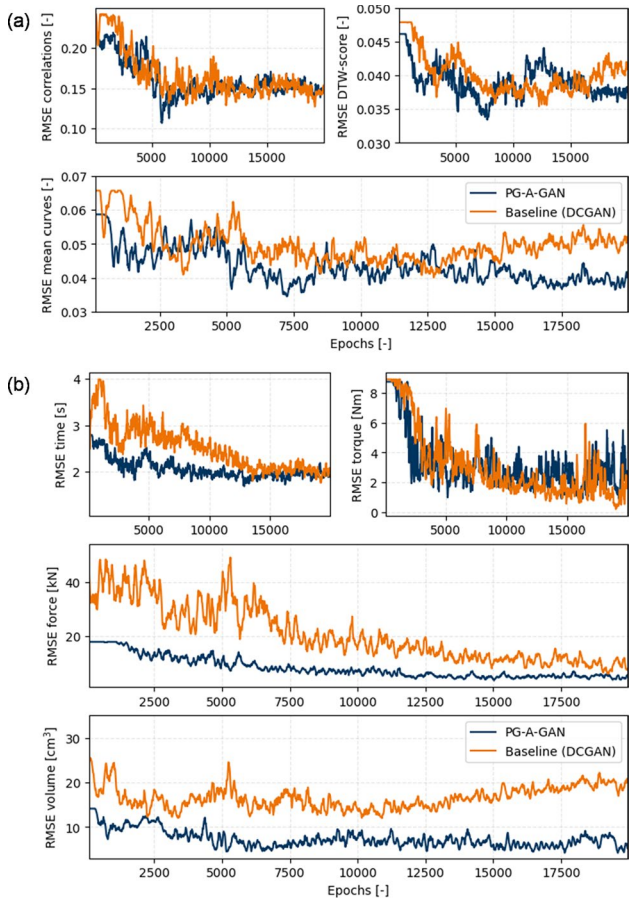


Fig. 17 **a** Loss curves of the observer scores, which reflect the distribution properties. **b** Loss curves of physics-guided alignment. All curves are smoothed with a moving average filter with a window size of 10

the physical losses, as presented in Sects. "Physics-informed modeling" and "Analytical transfer learning", for a selected instance where both DCGAN and PG-A-GAN were setup and employed under the same synthetic data generation conditions.

Considering the results relevant to the observer scores, Fig. 17a, the baseline DCGAN seems to show a generalized better convergence for higher number of epochs, though this cannot be generalized for all scores. In addition to that, considering that this convergence of the observer loss arises after the generator and discriminator losses have already converged, as per Fig. 5b, it can be concluded that the observer loss is diagnostic, not prescriptive. Since the observer is not backpropagated in the baseline DCGAN model, its delayed convergence does not influence the adversarial training dynamics but rather provides an external indication that higher-level temporal features become stable only after the adversarial game has already reached equilibrium. Accordingly, to fairly assess the differences in observer scores and physics-compliance between the

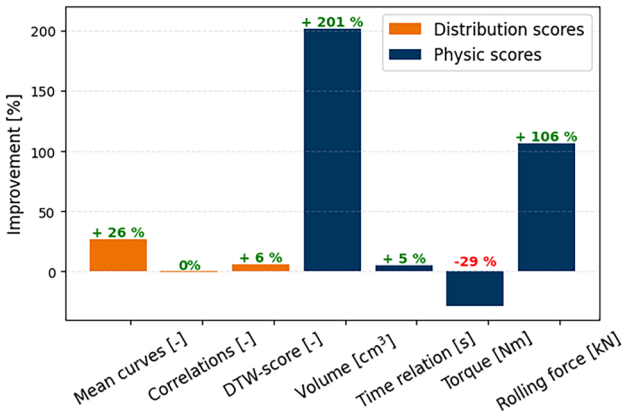


Fig. 18 Comparison of the relative percentage improvement of the PG-A-GAN to the DCGAN for the last 5000 epochs of the training loop where both models have achieved convergence

baseline DCGAN and PG-A-GAN approaches, Fig. 18 shows the same results of previous Fig. 16 but relevant only to the last 5,000 epochs of the training process, thus the last 25% of the whole training loop.

Even when considering only the last 25% of the training epochs, the improvement introduced by the PG-A-GAN persists and is amplified for some scores, such as the volume and mean curves, while the torque worsens by 29%, and the time improvement decreases from 22 to 5%. It should be noted that, since the torque model examines the torque of the generated data relative to the generated force, only a relative examination is possible. In other words, the model checks the generated torque in relation to the force generated by the dependency of the temporal friction coefficient, as previously shown in Fig. 11a. Therefore, potential errors in the force (i.e. not perfectly predicted force samples) can influence the evaluation of the torque surrogate model. This will be further elaborated in the discussion. Summarizing, when employing the PG-A-GAN with its physics-guided backpropagation, an improvement was observed for six out of the seven tracked scores, showing the superiority of the proposed approach. Besides that, further insights into the rationale behind the strong improvements provided by the proposed PG-A-GAN, such as the one shown for the volume consistency and rolling force, as well as the limitation showed for the rolling torque modeling, as reported in Fig. 18, will be provided in the discussion chapter.

AC-PG-A-GAN: conditional synthetic data generation

Considering the improvements provided by the proposed PG-A-GAN architecture, granted by the inclusion of physics-guided rules as an auxiliary loss, the aim of this section is to provide insights into the performance of this model when a condition is added for the generation of synthetic data. Therefore, the results presented in this section are relevant to the Auxiliary Classifier Physics-Guided Auxiliary GAN, subsequently referred to as AC-PG-A-GAN, as previously introduced in Sect. "Conditional synthetic data generation: the AC-PG-A-GAN approach".

Accordingly, to generate specific rolling instances, the final ring geometry in terms of the outer diameter, is added as an additional condition. Considering that in TCRR the ring width and height are fixed, the condition on the outer diameter and the concurrent consideration of the volume conservation allows for linking the initial and final geometries of the ring also during the generation of synthetic instances. From an operational perspective, both the generator and the discriminator no longer receive only the latent space vector but are also provided with scalar condition values representing the final ring diameter and the ring volume. Within this conditional generation framework, the generator produces rolling time series instances that correspond to a specific, user-defined end geometry and volume. In parallel, the discriminator is not limited to the adversarial classification of real versus generated sequences but is also equipped with a regression branch designed to predict the continuous condition values, specifically the final outer diameter and the ring volume, corresponding to the input time series. This auxiliary task enforces consistency between the generated signals and the user-defined end geometry, thus guiding the generator towards physically consistent and condition-compliant outputs.

The losses from these tasks are backpropagated through both networks to improve their ability to generate conditional time series. As in the case of the physics-guided constraints, the conditional loss is weighted by a factor λ to balance its influence. For stable training, it is crucial to equalize the weighting of the physical $\lambda_{1,p}$ and the conditional $\lambda_{2,c}$. To enable convergence, however, the physical loss is reduced by one third compared to the value used in Sect. "Deep convolutional GAN (DCGAN) model implementation", thereby leaving sufficient margin for the conditional loss to take effect.

Also, for the case of applying a condition to the synthetic data generation, the final generator model is selected by minimizing a cost function combining the physical loss with the observer loss reflecting the data distribution. Since the generated time series exhibited volatility, a Savitzky–Golay filter (SciPy, Virtanen et al., 2020) was applied as a post-processing step. The rationale behind this increased variability will be discussed in the following chapter. Figure 19 illustrates force–time series generated under the condition of a final outer diameter of 72 mm. Notably, the same model input for the AC-PG-A-GAN yields different output time series, a sign that the model has learned to reproduce the natural variability observed

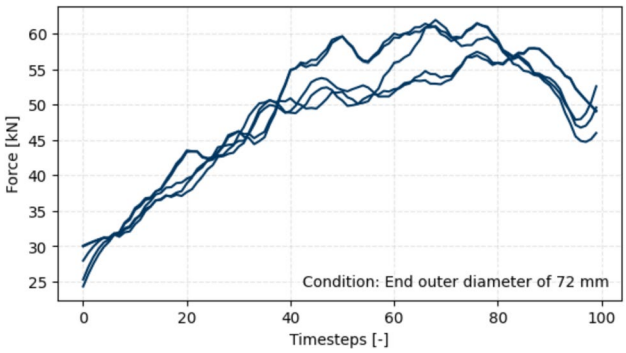


Fig. 19 Conditional time series generation for an outer diameter of 72 mm. The presented force curves offer data variability with the same model input, aiming to model natural process variance

in the experimental dataset. This variability is crucial for synthetic data generation, as it ensures sufficient diversity even under identical conditional inputs. Such diversity enhances the quality of the generated samples and, when these are used to train machine learning models, reduces the risk of overfitting.

To evaluate the performance of the AC-PG-A-GAN under different conditions, a broader range of end geometries was tested. To this end, Fig. 20 illustrates all experimental rolling instances corresponding to final diameters between 68.2 and 70.5 mm, together with the synthetic sequences generated by the AC-PG-A-GAN model. A comparison of the distributions reveals that the experimental data exhibits slightly higher variability, whereas the synthetic sequences remain concentrated within the same region without producing additional outliers. Importantly, the generated data reproduces the maximum and minimum ranges of most parameters, except for the time feature, where the synthetic instances tend to align more closely with the average rolling duration.

While the results presented in Fig. 20 demonstrate the comparability between real and synthetic data under a single condition, a more critical evaluation is to test the AC-PG-A-GAN model's sensitivity across different conditions. This type of analysis is essential, as it allows a direct comparison with alternative modeling approaches such as FE simulations or analytical formulations, and will be further elaborated in the discussion chapter. In this regard, Fig. 21 presents the condition-dependent evolution of thickness and outer diameter for two representative cases. The results show a clear separation between the two conditions, confirming that the model correctly adapts its outputs to the specified geometry. Furthermore, no additional outliers are observed compared to the experimental dataset, indicating stable and condition-consistent generation.

To provide a quantitative evaluation of the AC-PG-A-GAN performance as a synthetic data generator and process modeling approach, the RMSE between the average synthetic and experimental time series was calculated for each differing condition shown in Fig. 22 while the

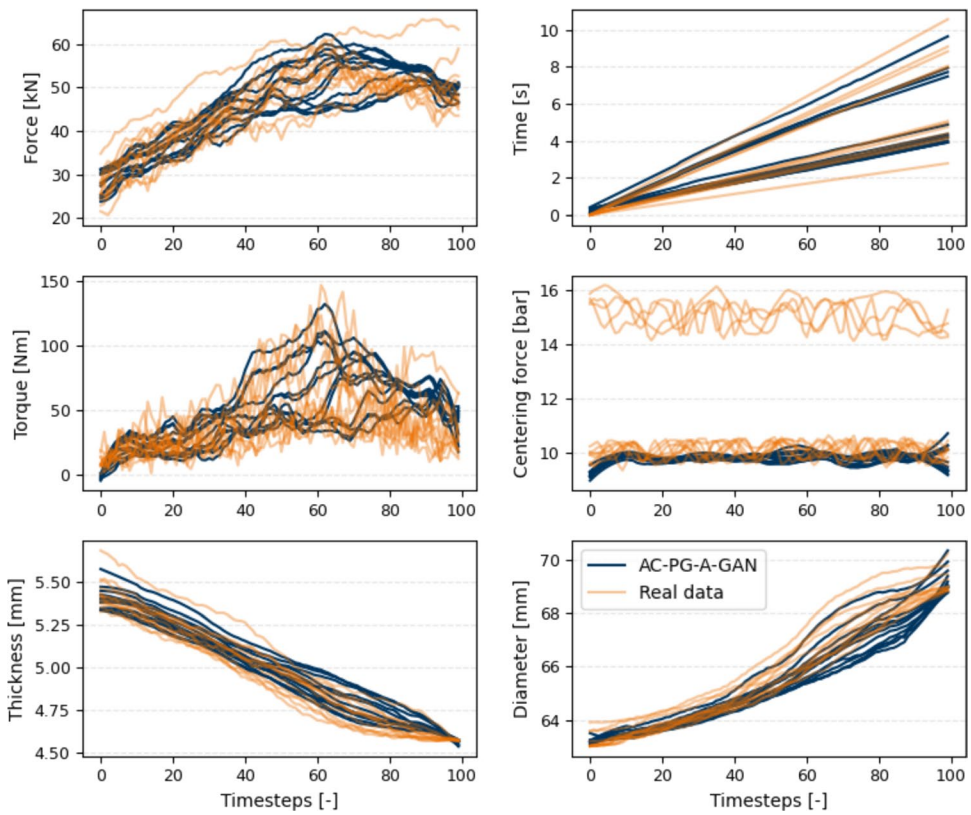


Fig. 20 Conditional time series for an outer diameter between 68.2 and 70.5 mm compared to the real measured time series

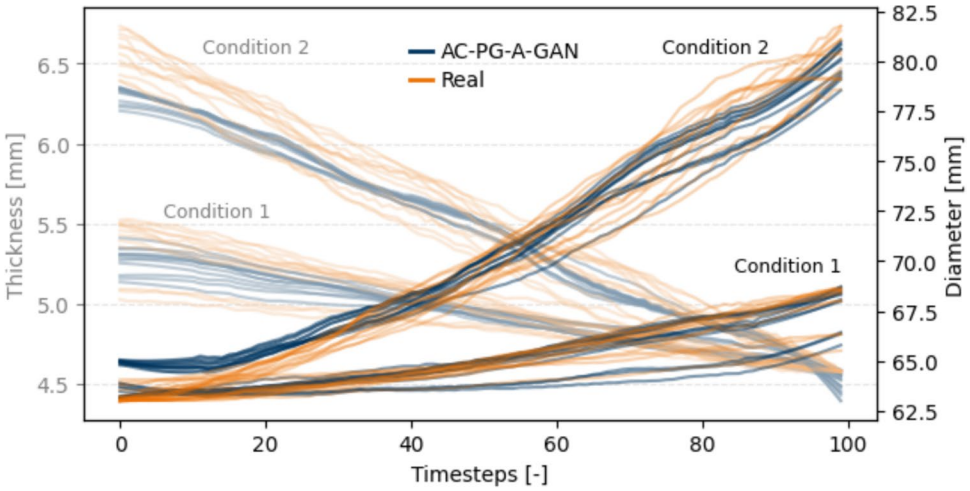


Fig. 21 Conditional time series for the outer diameter and the thickness for two sets of conditions

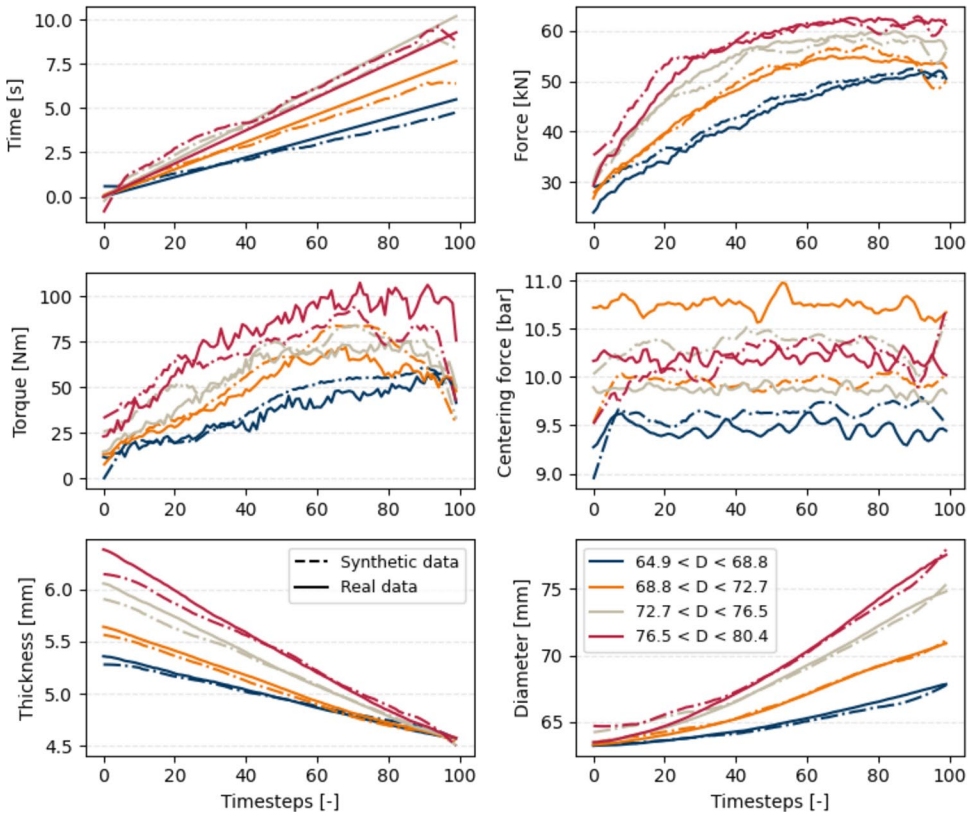


Fig. 22 Conditional average time series of all process parameters (time, force, torque, centering force, thickness, and outer diameter) for four sets of final diameter conditions. Solid lines correspond to experimental data and dashed lines to AC-PG-A-GAN generated sequences. The figure illustrates that the model adapts its outputs to different conditions, reproducing the expected trends across parameters while preserving sensitivity to the imposed final geometries

resulting deviations are summarized in Table 4. Among the parameters, the torque displays the largest error, as also shown for the unconditioned PG-A-GAN, although its sensitivity to the imposed conditions remains clearly observable in Fig. 22. Overall, the deviations remain within narrow bounds, on the order of a few percent of each parameter's range, corresponding, for the geometric features, to sub-millimeter accuracy (compare Table 4). These results confirm that the AC-PG-A-GAN can generate condition-dependent sequences that reproduce the experimental trends with high fidelity across multiple process variables while also achieving high variance still withing the physically consistent target variables' boundaries.

Table 4 Accuracy of AC-PG-A-GAN generated data for four conditions based on the outer diameter. The metric is calculated by comparing the average curves of real and AC-PG-A-GAN data

Time series type	RMSE Real to AC-PG-A-GAN
Time	0.49 s
Force	1.90 kN
Torque	10.20 Nm
Centering force	0.43 bar
Thickness	0.06 mm
Outer diameter	0.36 mm

Discussion

Considering the different modelling methodologies, the discussion follows the structure of the results chapter and is organized into three parts:

(I) Baseline DCGAN results,
(II) PG-A-GAN performance in data generation and comparison with DCGAN,
(III) Ring geometry-based conditional data generation by AC-PG-A-GAN.

Starting with the baseline DCGAN, the generated samples resemble real-world samples, indicating the model's capability to generate realistic ring-rolling sequences in TCRR. Selecting the best generator model presented an initial challenge, since the adversarial loss does not provide any direct information about the quality of the data produced. Therefore, in this research, the custom-made observer was designed to support the selection of a suitable model showing the greatest adherence to real data distribution. In this regard, the observer was designed to include three metrics, namely the DTW, the average time series, and the feature correlations score, and proved suitable for the task. The different behavior of the observer components over time (Fig. 14) supports selecting a model with low mismatch across all metrics, thereby reducing the risk that model selection is dominated by only one characteristic of the data distribution. Furthermore, the minimum observer score is achieved long after the conventional GAN loss has converged, making it worthwhile to train the model for a higher number of epochs.

The physics-guided modeling and implementation in the GAN scheme added to the observer four more metrics: force and torque alignment, diameter to time relation, and volume consistency. These additional metrics enable an explicit physical evaluation of the generated data. Similar to the observer metrics, these quantities were evaluated at each epoch during training; unlike the observer, however, they were also backpropagated as auxiliary losses to construct the physics-guided auxiliary GAN (PG-A-GAN). The observer is not backpropagated to maintain an unbiased evaluation metric. To enable such model training, correct implementation of the models in ML training is necessary. In the present *TensorFlow* implementation, graph execution was required to ensure reasonable computational time; accordingly, the model components must be differentiable, which should be considered when adapting PG-A-GAN to other domains. The main challenge in this regard was associated with training stability, thus annealing plans had to be included to enable learning. In the early stages of training, the generator distribution deviates strongly from the imposed physical rules, resulting in very high auxiliary losses and unstable gradients. On the other hand, once training has stabilized, the physical contribution to the loss can be fully included to guide the generator toward physically consistent solutions, resulting in better convergence behavior during training as well as improved final observer and physics scores. Although the improvements in observer scores are modest, they are still meaningful because the observer metrics were not backpropagated and therefore provide an external evaluation of data quality. Nevertheless, when comparing the benchmark DCGAN with the proposed PG-A-GAN, physical adherence improves significantly for rolling force and volume consistency, with percentage improvements of 201% (in relation to cm^3) for volume and 106% (in relation to kN) for rolling force, as summarized in Fig. 18. Also, the time-diameter relationship improves slightly, while the rolling torque worsens by 3% (with respect to Nm) before baseline convergence (see Fig. 16) and by 29% (see Fig. 18) after baseline convergence. Since six of the seven tracked parameters improved, the PG-A-GAN can be regarded as the better-performing approach overall.

As also mentioned in the comments to the results in Sect. "PG-A-GAN: synthetic data generation", the modeling of the rolling torque still retains certain limitations when dealing with the PG-A-GAN model. In this study, torque was modelled in relation to the predicted force, achieving reasonable accuracy of 10.144 Nm (Fig. 13, RMSE). Therefore, the accuracy of the rolling torque modeling depends directly on the accuracy of the rolling force prediction, since the torque model is formulated as a function of the generated force. In turn, the quality of this force prediction depends on the physical plausibility of the GAN-generated force data (p_g). As reported in Fig. 13 and Sect. "PG-A-GAN: synthetic data generation", the force model itself has an RMSE of 4.825 kN. Consequently, generated force samples with errors below this threshold are not meaningfully penalized by the backpropagated force loss. The dependency between force and torque losses is illustrated in Fig. 23, which is a scatter plot displaying torque loss over force loss for each training epoch. This dependency is also reflected by a *Pearson* correlation of 0.78, indicating that higher force losses are associated with higher torque losses.

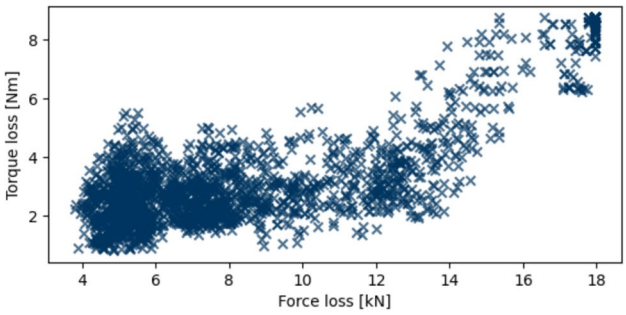


Fig. 23 Scatter plot showing the relationship between force and torque loss over training epochs for the PG-A-GAN

These deviations directly affect the accuracy of the torque modeling and therefore the effectiveness of the PG-A-GAN scheme. The PG-A-GAN shows a slight worsening in torque alignment compared with the baseline DCGAN, amounting to approximately 1 Nm (Fig. 17). However, this difference is small relative to the torque model RMSE of 10.144 Nm. This indicates that the choice and accuracy of the surrogate models are decisive factors in determining the effectiveness of the PG-A-GAN. Accordingly, the torque model represents a limiting case of ATL-based backpropagation in PG-A-GAN, where the accuracy of the underlying surrogate model is insufficient to yield a clear training benefit. This is reflected in the torque loss, which remains comparable to that of the baseline model and therefore does not indicate a clear benefit from ATL-based backpropagation in this case. By contrast, the rolling-force results show that ATL can enhance analytical model accuracy within the latent space and thereby provide a more effective process model for PG-A-GAN training. Future work could therefore integrate more detailed analytical torque models, such as the slab-analysis approach of Parvizi et al. (2011), and adapt them within the ATL framework for physics-guided training. In conclusion, the introduction of physics into the GAN framework improved training performance; accordingly, the PG-A-GAN was selected as the basis for the conditioned generation of ring-rolling data for varying end geometries.

A further point affecting both the DCGAN and the PG-A-GAN concerns their multivariate data-generation strategy, in which the generator predicts all feature time series simultaneously from each latent-space vector. This approach enables direct feature relations to be constructed, as seen in real-world data where every process parameter depends on various other parameters in an interconnected and cross-influencing scenario. Despite the above-mentioned benefits of multivariate generation, the increased scatter in the generated time series, which resembles a vibration-like behavior, could be a downside. This behavior is less pronounced in univariate modeling, where the restriction to a single target variable enables better local time-series resolution. This can

Table 5 Standard deviations of outer diameter and rolling force with respect to a fitted signal function

Feature time series	Real data	Univariate DCGAN	Multivariate PG-A-GAN
Outer diameter [mm]	0.24	0.27	0.37
Force [kN]	2.19	2.10	2.28

be assessed by considering the variance with respect to a fitted function that approximates the signal without scatter or noise. To do this, a second-order polynomial function is fitted to each time series, and the standard deviation is calculated. Table 5 shows the standard deviation for the outer diameter and rolling force. As can be seen in Table 5, the residual standard deviations obtained from univariate modelling are similar to or smaller than those observed in the real data. The PG-A-GAN data shows higher values for both feature time series. This is particularly evident for the outer diameter, where noise and scatter are less prevalent in real data than in the force data. This highlights the trade-off of multivariate modeling: capturing cross-feature temporal dependencies comes at the cost of increased local noise in the generated signals.

For this reason, a multivariate approach was chosen to reflect the coupled nature of the process as closely as possible. However, if local resolution or focus is required, univariate generation should be considered. In this case, a multi-head approach could be used to mimic multivariate generation by training on the same samples with the same latent space vector copied for each feature, as applied by Fahle et al. (2022). Future research could also enable extensions for multivariate generation, with respect to local variance in time series. For instance, the frequency spectrum could be modeled by a second head with the same latent vector and applied to the generated time series to transform local patterns, in respect to the dataset distribution. In addition to that, latent representations can be introduced in training, such as proposed in Yoon et al. (2019), for the TimeGAN (Time-series Generative Adversarial Networks). Since the primary goal of this contribution is to reproduce the overall trends and physical consistency of the generated data, a Savitzky–Golay filter (SciPy, Virtanen et al., 2020) was applied as post-processing. After smoothing, no major differences were observed between synthetic and real time series at the level of overall trends and major signal variations.

Turning to the conditional synthetic data generation results, the AC-PG-A-GAN uses the end geometry as conditioning input within an auxiliary classifier GAN (ACGAN) framework. Conditioning is applied only on the final geometry, since the starting geometry varies minimally with the outer diameter (approximately 3 mm, Table 1), whereas the

final outer diameter spans a broader range between 64 and 82 mm. To accommodate this conditioning, the physics-guided loss was reduced by 30% to integrate the class loss during training. The conditional generation established through this approach enables two main applications: (i) process modeling, as a fast and flexible alternative to numerical simulation or analytical modeling, and (ii) data augmentation for machine learning tasks. A key limitation, however, lies in the latent space: since GANs are inherently restricted to the distribution of the training dataset, the model's action space is confined to end geometries of cold ring-rolled 100Cr6 forged rings with outer diameters between 64 and 82 mm. Predictions remain possible and condition-sensitive within the minimum and maximum geometry range, despite gaps in the original latent space, as shown for the thickness predictions over normalized timesteps reported in Fig. 24. In this case, only filtered training data was used to train the AC-PG-A-GAN; instances with a thickness between 5.3 mm and 5.7 mm were excluded and were not part of the training conditioning latent space. Nevertheless, the predictions for these geometries are reasonable, filling the space between the conditions in a sensitive way. This demonstrates the feasibility of generating predictions in excluded regions within the covered parameter range, that is, in an interpolative rather than extrapolative manner. However, due to its modelling nature, making predictions in an extrapolative manner remains a limitation, as is common for fully data-driven machine-learning models. Expanding the parameter space of ring rolling instances in the dataset would inherently broaden the range of time series generation without the need for extrapolative predictions. However, TCRR data acquisition using a single machine, as presented in this research, is limited by the machine's geometry and possible range.

Therefore, future work could involve combining several ring rolling machines with different dimensions to expand the latent space. To illustrate this effect, a preliminary yet informative analysis is provided here through a limited application of the benchmark DCGAN to the radial-axial ring rolling (RARR), a process offering a broader geometric operating window than that of the TCRR setup considered in the main study presented in this research. In the radial-axial ring rolling process (RARR), which is used to produce large ring geometries, a single machine and toolset can produce a greater variety of ring geometries than the TCRR. Therefore, this research briefly demonstrates the data generation of the RARR process to show the effect of a wider parameter space. The DCGAN baseline of this study was applied to a FEA dataset comprising 220 flat ring rolling instances (Mirandola et al., 2021) to visualize the effect on the range of generated time series. As Fig. 24b shows, the final geometries range from 0.9 to 1.8 m, demonstrating the greater coverage of DCGAN predictions and the potential for wide-ranging process predictions with more flexible forming machines.

Despite the limitation of the given latent space, the proposed method offers two major advantages. First, it was benchmarked against 3D finite-element simulations as a stronger baseline reference and showed comparable, and for several variables better, predictive accuracy. For this purpose, the same evaluation procedure used to assess the AC-PG-A-GAN results reported in Table 4 was also applied to 50 FE simulations generated under the same process settings (see Sect. "Tangential cold ring rolling experiments"). In this comparison, the AC-PG-A-GAN achieved RMSE values of 1.9 kN for the rolling force, 10.20 Nm for the torque, 0.06 mm for the thickness, and 0.36 mm for the outer diameter. Relative to the FEA-based reference, this corresponds to lower errors of 5.9 kN for force, 6.37 Nm for torque, and 0.09 mm for both outer diameter and thickness.

Besides its competitive accuracy relative to the FEA baseline, the proposed AC-PG-A-GAN drastically reduces

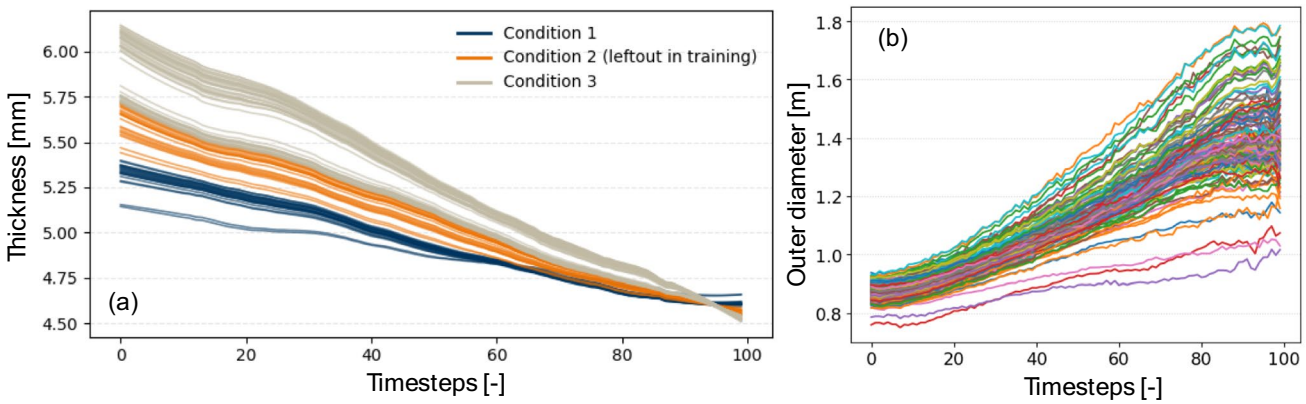


Fig. 24 **a** Predictions outside the training latent space (interpolative). The displayed orange time series have not been part of training or validation data. **b** 150 random outer diameter predictions with a DCGAN for radial–axial ring rolling data with a greater range of product geometries in the dataset

the computation time required for one rolling instance from 230 CPU-hours on average for one simulation to sub-second generation (0.2 ms) of large datasets suitable for both process modeling and data-driven applications. The simulations were calculated on a high-performance cluster of the *Technical University of Dresden*, using up to 32 cores and parallel computation. The GAN model requires 111.3 ± 17 min of training, averaged over three runs. Once trained, only model inference for different conditions is necessary. The model inference and training were performed on an Intel(R) Xeon(R) W-2255 CPU. This speed makes the method particularly attractive for surrogate-model-based optimization, where calculation time often represents a bottleneck. It should be noted that the FEA comparison proposed here serves as a baseline evaluation of predictive accuracy and computational efficiency, whereas the following analyses address diversity, robustness, and interpolation capability within the latent space.

Taken together, these results show that the proposed framework predicts average process time series with accuracy comparable to FEA simulations while also capturing the inherent process variance across different input conditions, as illustrated in Figs. 20 and 21. Beyond the visually clear variance in the generated instances reported in Figs. 20 and 21, quantitative metrics are required to assess the diversity of the generated samples more rigorously. This is important to identify possible mode collapse, where the generator produces only limited variations around mean trajectories instead of representing the full process distribution. In such a case, different random latent seeds under the same conditioning input would lead to nearly identical outputs and, consequently, to reduced sample-to-sample variability. Accordingly, the *Wasserstein distance* was therefore calculated to compare the distributions of generated and experimental trajectories. By measuring the minimal transport cost between both distributions, it quantifies discrepancies in distributional support and data space coverage (Peyré & Cuturi, 2018). For instance, for the average force trajectory, the Wasserstein distance amounts to 1.8 kN (computed using SciPy; Virtanen et al., 2020). Compared to the overall standard deviation of 8 kN in the experimental dataset, this indicates close distributional alignment relative to inherent process variability. In combination with the interpolation performance reported in Fig. 24, these results indicate that the AC-PG-A-GAN model does not only reproduce mean trajectories but also captures a meaningful portion of the distributional variance observed in the experimental data.

In addition to that, given the limited dataset size, the risk of overfitting must be addressed explicitly. In addition to standard regularization strategies such as dropout, cross-validation-based hyperparameter tuning, and early stopping, the integration of physical knowledge plays a central role. In this regard, the physics-informed and ATL-based loss components introduce structural constraints that are independent of the specific generator and discriminator architectures. Consequently, these physically motivated regularizers help mitigate overfitting and penalize physically implausible trajectories even under limited data availability. As demonstrated in recent literature, such a strategy is increasingly relevant in modern machine-learning model development (Modanloo et al., 2025).

Beyond its role as a regularized generative model, the short inference time of the trained AC-PG-A-GAN also opens up several potential downstream applications. In particular, the prediction of expected mean trajectories together with their variance could support probabilistic process assessment and, potentially, early detection of instabilities and abnormal operating regimes. By extending the generator conditioning to include current process states, such as time step and geometry, the AC-PG-A-GAN framework could in principle support fast forward rollouts of the ring evolution. Given the current inference time of about 0.2 ms, this may provide a basis for real-time closed-loop digital twins in process control and monitoring.

Beyond process modeling, a second important application of the proposed architecture is data augmentation for machine-learning tasks. In this context, the generated process variance (see Fig. 19) can be utilized to augment existing datasets and provide valuable complementary information. Potential use cases in ring rolling include quality prediction tasks, such as fishtail defects and circularity, as well as the optimization of process settings and parameters, typical of the TCRR and other rolling processes employed in the manufacturing industry. In this regard, the torque prediction model presented in Sect. "Analytical transfer learning models training and performance" achieved a best RMSE of 10.144 Nm and was therefore selected as the best model. However, because of the limited database size for the torque surrogate model definition within the ATL module, the training process was sensitive to model initialization, initial weights, and batch composition, resulting in considerable run-to-run variability and making robust model selection challenging. Across five runs, the initial model achieved an average RMSE of 16.26 ± 4.46 Nm. After augmenting the training database with AC-PG-A-GAN-generated synthetic data, this average RMSE was reduced to 12.11 ± 1.36 Nm. Although the best model's peak performance in a single run was not outperformed by this data augmentation, the training stability was improved significantly, thereby enhancing the models' generalization capabilities. This observation is consistent with recent work showing that, when real measurements are scarce or expensive to obtain, augmenting the training set with additional, distribution-consistent synthetic samples can increase the effective coverage of the training distribution and help the model learn more robust features, thereby improving generalization and reducing run-to-run

sensitivity (Sun et al., 2025a, 2025b; Yu & Liu, 2024). This result further bolsters the usefulness of the proposed approach for ML data augmentation, as shown in similar application reported in recent literature contributions. At the same time, these application-oriented results should be interpreted within the limitations imposed by the training distribution and the AC-PG-A-GAN model architecture.

A further limitation concerns generalization beyond the training distribution, since GANs learn representations of the latent space defined by the available dataset. In this regard, it should be noted that changing fundamental process regimes such as introducing a new material with substantially different constitutive response, switching from TCRR to RARR, or applying strongly different geometries/contact conditions, would typically require updating the physical constraint formulations and full retraining of the PG-A-GAN model. In contrast, changes that primarily affect the calibration of the analytical relations embedded in the auxiliary losses, such as moderate shifts in friction behavior, flow-stress scaling, or sensor/measurement scaling within the same TCRR setup could be addressed by tuning only the ATL surrogate components, while keeping the GAN architecture unchanged.

As a final remark, broadening the training dataset and diversifying the types of samples used for training would improve generalization. For instance, including samples of profiled and hot ring rolling and assigning different types with class labels would expand the model's ability to generalize in this new latent space, though retraining would be necessary. In this regard, combined and heterogeneous ring rolling datasets offer significant potential for ML in data-centric approaches and could allow significant generalization and enable cross-use of research results and models. However, data aggregation, data quality and data copyright remain challenges that need to be addressed in future research to enhance existing modelling, which is limited by the dataset used for training. Another approach to enable cross modeling and expanding trained models could be represented by transfer learning, where a base model is applied to a similar, but different setting, reducing the number of experiments needed for training (Seitz et al., 2023).

At this stage, it is also important to highlight the limitations, such as the simplifications applied to physical modelling and latent space in model training. Furthermore, the time series were smoothed using Savitzky–Golay filters, a fact which may reduce the model capability in representing correctly local patterns if a multivariate modeling approach is chosen. This post-processing step should be regarded as a current limitation of the proposed model related to the generator's inability to reproduce high-frequency components and sharp transients reliably. Hence, at the present stage and with the current model architecture, it is not possible to quantify the accuracy in modeling local high-frequency events, such as abrupt changes in force/torque or rapid geometry/contact transitions. This limitation is particularly relevant for applications such as anomaly detection, where an extensions of the proposed framework under a univariate modelling conditions or alternative approaches should be considered. However, a promising direction to address this limitation is to incorporate frequency–domain information into the learning representation, for instance by comparing spectra via FFT-based losses or by modeling the signals in a compact spectral representation and reconstructing the time series from it, thereby learning both low- and high-frequency content without relying on post-hoc filtering. Nevertheless, the results shown in Figs. 20, 21, and 24 indicate that the AC-PG-A-GAN model still captures the overall temporal evolution of the process, even though local high-frequency components and sharp transients are not represented reliably with the current multivariate architecture. Future work could explore other architectures that use different temporal latent representations to determine whether they can also model the aforementioned high-frequency components and sharp transients without the need for explicit information, such as that provided by FFT losses.

An important aspect related to both surrogate modeling and synthetic data generation concerns the selection of multivariate and univariate modeling strategies. Univariate modeling can be chosen instead of multivariate modeling to generate frequency sensitive outputs, due to the greater reliability of the signal-to-noise ratio with respect to real data (see Table 5). However, univariate data generation ignores cross-feature dependencies, treating each feature separately by definition and thus it is only suitable when features can be treated independently. One possible way to combine the current PG-A-GAN framework with univariate modeling would be a multi-head architecture similar to that proposed by Fahle et al. (2022) for RARR, in which separate univariate time series are generated for each feature, then combined and evaluated jointly by the discriminator, while also incorporating the physics-guided auxiliary loss proposed here. Although interesting, and potentially the solution to achieve cross-variables influence while allowing for high frequency data modeling, such as in defect detection applications, this aspect was beyond the scope of this research and therefore was not considered in the proposed architecture.

Overall, the proposed approaches, both the unconditioned PG-A-GAN and the conditioned AC-PG-A-GAN, demonstrate the feasibility and practicality of incorporating physics into GAN-based time series generation for TCRR. Within the learned latent space, the proposed model can generate ring-rolling time series for six features, namely time [s], rolling force [kN], rolling moment [Nm], centering force [bar], outer diameter [mm], and the ring thickness [mm], with strong sensitivity to the geometrical condition (Fig. 22). Nevertheless, future work should explore

additional downstream applications beyond the presented use cases of process modeling and data augmentation to further validate PG-A-GAN in industrial settings. In addition to that, other recently developed generative ML algorithms, such as diffusion models (Ho et al., 2006) and transformer GANs (Srinivasan & Knottenbelt, 2022) could be tested to assess the possibility of integrating ring rolling knowledge in modeling as alternatives to the GANs architecture. Although not directly tested in this research, this would enable an overall performance comparison across different ML approaches.

Conclusions

A generative ML approach based on GANs for the TCRR is developed and presented in this contribution. The focus is on including physical knowledge in a physics-guided training framework, where analytical and surrogate process models contribute auxiliary losses that guide the training toward physically consistent solutions. Accordingly, the proposed framework was applied as a process-data generator for the time-series generation of both unconditioned and conditioned ring geometries. The four key results can be summarized as follows:

(I) The baseline DCGAN is a suitable algorithm to generate multivariate TCRR time series for the time [s], rolling force [kN], rolling moment [Nm], centering force [bar], outer diameter [mm], and the ring thickness [mm]. A custom-developed observer integrated into the training process supported the selection of the best generator model based on the adherence of the generated data distribution to the experimental data distribution (p_g).

(II) The introduction of physics-guided constraints in the PG-A-GAN, implemented through auxiliary losses based on physics-informed relations and analytical transfer learning (ATL), enabled guided training alongside known relationships of the ring rolling process. Six out of seven quantified parameters improved in adherence to real world data in comparison to the baseline GAN modeling (DCGAN), such as the ring volume, whose error was reduced from ~ 20 to ~ 7 cm^3. Rolling torque was the only physical loss that worsened slightly, by about 1 Nm, indicating that the effectiveness of the proposed guidance depends strongly on the quality of the underlying analytical or surrogate process model.

(III) Conditional data generation with the AC-PG-A-GAN enabled effective process modeling for customizable end-ring geometries. Importantly, the generated predictions exhibit natural process variance, *id est*, the same condition does not always produce an identical time series, but rather a realistic range of outcomes that remain sensitive to the specified condition. The prediction errors remain within a few percent, with an RMSE of 0.36 mm for the outer diameter and 1.9 kN for the rolling force.

(IV) The AC-PG-A-GAN was benchmarked against 3D finite-element simulations and showed competitive predictive accuracy together with a drastic reduction in computation time, from several CPU-hours per simulation to sub-millisecond inference once trained. However, as with all GAN-based approaches, predictions remain limited to the latent space defined by the training data, so that interpolation within the covered range is feasible, while extrapolation remains a limitation.

The derived generative models can be considered as a fast-calculating extension to numerical and analytical modeling in process modeling, for instance for process design, optimization, and synthetic data generation. In addition, the generated data also showed potential for data augmentation in ML applications, improving the average performance and stability of the torque surrogate model across repeated runs. Future work should extent and benchmark the proposed architectures on a wider range of applications, such as profiled ring rolling as well other manufacturing process where ML modeling requires great training datasets. In parallel, further improvements in the physical and surrogate modeling, particularly for rolling torque, as well as investigations of alternative generative backbones, could further enhance the quality and generalizability of the proposed PG-A-GAN framework.

Appendix A: Pseudo-code for PG-A-GAN model training.

Notation

- Dataset of real multivariate time series: $\mathcal{D} = \{x\}$
- Generator G_θ, discriminator D_ϕ
- Latent noise $z \sim p(z)$
- (Optional) condition c (only if you present AC-PG-A-GAN)
- Auxiliary physics models grouped as $\mathcal{P}(\cdot)$
- Annealing weight $\alpha(t) \in [0,1]$, ramps up over epochs
- Physics weight λ_{phys} (annealed over epochs)
- Standard GAN losses: L_D, L_G(e.g., logistic/BCE)

PG-A-GAN training

Procedure TRAIN_PG_A_GAN ($\mathcal{D}$, G_θ, D_ϕ, T, n_D, λ_{phys})
initialize θ, ϕ (network parameters)
for epoch $t = 1$ **to** T **do**
$\alpha \leftarrow \text{ANNEAL}(t)$
(A) Update discriminator n_D times
for $k = 1$ **to** n_D **do**
sample minibatch $x \sim \mathcal{D}$
sample $z \sim p(z)$
if conditional training **then** sample condition c and set $\hat{x} \leftarrow G_\theta(z, c)$
else $\hat{x} \leftarrow G_\theta(z)$
$L_D \leftarrow$ DISCRIMINATOR_LOSS ($D_\phi, x, \hat{x}$)
$\phi \leftarrow$ OPTIM_STEP ($\phi, \nabla_\phi L_D$)
end for
(B) Update generator once (adversarial + auxiliary physics)
sample $z \sim p(z)$
if conditional training **then** sample condition c and set $\hat{x} \leftarrow G_\theta(z, c)$
else $\hat{x} \leftarrow G_\theta(z)$
(C) Adversarial objective (fool discriminator)
$L_{\text{adv}} \leftarrow$ GENERATOR_ADV_LOSS ($D_\phi, \hat{x}$)
(D) Auxiliary physics objective (differentiable constraints / surrogate models)
$L_{\text{phys}} \leftarrow$ PHYSICS_AUX_LOSS ($\hat{x}; \mathcal{P}$)
(E) Total generator loss with annealing
$L_G \leftarrow L_{\text{adv}} + \alpha \cdot \lambda_{\text{phys}} \cdot L_{\text{phys}}$
$\theta \leftarrow$ OPTIM_STEP ($\theta, \nabla_\theta L_G$)
end for
return trained G_θ
end procedure

Physics-guided auxiliary loss

Procedure PHYSICS_AUX_LOSS$(\hat{x}; \mathcal{P})$

$\hat{x}$ contains generated time series (e.g., time, diameter, thickness, force, torque)

extract generated channels: $\hat{t}, \hat{d}, \hat{h}, \hat{F}, \hat{M}$

(A) Volume consistency

$\hat{V}(\tau) \leftarrow$ RING_VOLUME $(\hat{d}(\tau), \hat{h}(\tau), w)$

$L_{\text{vol}} \leftarrow \text{MSE}(\hat{V}(\tau), V_{\text{ref}})$ compare to dataset mean or reference volume

(B) Time–diameter expansion consistency (semi-analytical ring growth)

$\tilde{t} \leftarrow$ DIAMETER_TIME_MODEL$(\hat{d}, \hat{h}$,process params)

$L_{\text{time}} \leftarrow \text{MSE}(\tilde{t} - \hat{t})$

(C) Analytical transfer learning (ATL) force consistency

$F_{\text{slip}} \leftarrow$ SLIP_LINE_FORCE$(\hat{d}, \hat{h}$,material/contact terms)

$\gamma_F \leftarrow$ MLP_FORCE_SCALE(geometry,process params, $\hat{t}$)

$\tilde{F} \leftarrow \gamma_F \cdot F_{\text{slip}}$

$L_{\text{force}} \leftarrow \text{MSE}(\hat{F}, \tilde{F})$

(D) Torque consistency via force–torque coupling (surrogate friction)

$\mu(\tau) \leftarrow$ POLY_FRICTION$(\hat{F}(\tau))$ ▷ 3rd-order polynomial in normalized force

$\tilde{M}(\tau) \leftarrow \mu(\tau) \cdot \hat{F}(\tau) \cdot R_{\text{roll}}$

$L_{\text{torque}} \leftarrow \text{MSE}(\hat{M}, \tilde{M})$

Combine physics losses

$L_{\text{phys}} \leftarrow L_{\text{vol}} + L_{\text{time}} + L_{\text{force}} + L_{\text{torque}}$

return L_{phys}

end procedure

Acknowledgements Prof. Luca Quagliato was supported by the RP-Grant 2024 of Ewha Womans University. The results were partly developed in the Project 499350001, funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation).

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability The dataset used for the modeling presented in this study is publicly available at the following persistent GitLab link: https://gitlab.ruhr-uni-bochum.de/lps-projects/kodaml/genai_tcrr/-/tree/039eec7e4b8aeba6bb8e95bab38f3c35e51dde3a/. The Python script is made available on request to the corresponding author.

Declarations

Conflict of interest The authors declare no competing interest.

References

Abadi M. et al. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. www.tensorflow.org

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp 2623–2631. https://doi.org/10.1145/3292500.3330701

Allwood, J. M., Tekkaya, A. E., & Stanistreet, T. F. (2005). The development of ring rolling technology. *Steel Research International, 76*(2–3), 111–120. https://doi.org/10.1002/srin.200505981

Arjovsky, M., & Bottou, L. (2017). Towards principled methods for training generative adversarial networks. In: *5th International Conference on Learning Representations, ICLR 2017—Conference Track Proceedings*, pp 1–17.

Arthington, M. R., Havinga, J., & Duncan, S. R. (2020). Control of ring rolling with variable thickness and curvature. *International Journal of Material Forming, 13*(2), 161–175. https://doi.org/10.1007/s12289-019-01475-6

Bazarbaev, M., Chuluunsaikhan, T., Oh, H., Ryu, G. A., Nasridinov, A., & Yoo, K. H. (2022). Generation of time-series working patterns for manufacturing high-quality products through auxiliary classifier generative adversarial network. *Sensors, 22*(1). https://doi.org/10.3390/s22010029

Berti, G. A., Quagliato, L., & Monti, M. (2015). Set-up of radial-axial ring-rolling process: Process worksheet and ring geometry expansion prediction. *International Journal of Mechanical Sciences, 99*, 58–71. https://doi.org/10.1016/j.ijmecsci.2015.05.004

Cao, J., Bambach, M., Merklein, M., Mozaffar, M., & Xue, T. (2024). Artificial intelligence in metal forming. *CIRP Annals, 73*(2), 561–587. https://doi.org/10.1016/j.cirp.2024.04.102

Deng, J., & Mao, H. (2015). A blank optimization design method for three-roll cross rolling of complex-groove and small-hole ring. *International Journal of Mechanical Sciences, 93*, 218–228. https://doi.org/10.1016/j.ijmecsci.2014.10.024

Faegh, M., Ghungrada, S., Oliveira, J. P., Rao, P., & Haghighi, A. (2025). A review on physics-informed machine learning for process-structure-property modeling in additive manufacturing. *Journal of Manufacturing Processes, 133*, 524–555. https://doi.org/10.1016/j.jmapro.2024.11.066

Fahle, S., Glaser, T., Kneißler, A., & Kuhlenkötter, B. (2022). Improving quality prediction in radial-axial ring rolling using a semi-supervised approach and generative adversarial networks for synthetic data generation. *Production Engineering, 16*(1), 175–185. https://doi.org/10.1007/s11740-021-01075-x

Farrag, A., Yang, Y., Cao, N., Won, D., & Jin, Y. (2025). Physics-informed machine learning for metal additive manufacturing. *Progress in Additive Manufacturing, 10*, 171–185. https://doi.org/10.1007/s40964-024-00612-1

Fernandez, F., Saravanan, S., Omongos, R. Lou., Troncoso, J. F., Galvez-aranda, D. E., & Franco, A. A. (2025). Transfer learning assessment of small datasets relating manufacturing parameters with electrochemical energy cell component properties. *NPJ Advanced Manufacturing,*, pp. 1–14. https://doi.org/10.1038/s44334-025-00024-1

Ferré, F., Allassonnière, S., Chadebec, C., & Minville, V. (2025). Generating artificial patients with reliable clinical characteristics using a geometry-based variational autoencoder: Proof-of-concept feasibility study. *Journal of Medical Internet Research,*, 27. https://doi.org/10.2196/63130

Ge, S., Wu, Y., Qin, Z., & Hao, T. (2025). Deformation behavior of profiled ring rolling and process optimization for uniformity. *International Journal of Advanced Manufacturing Technology*, pp 871–889. https://doi.org/10.1007/s00170-025-15206-6

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Nets. Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'14), Vol. 2. https://arxiv.org/abs/1406.2661

Guo, W., Wang, Y., Chen, X., & Jiang, P. (2024). Federated transfer learning for auxiliary classifier generative adversarial networks: Framework and industrial application. *Journal of Intelligent Manufacturing, 35*(4), 1439–1454. https://doi.org/10.1007/s10845-023-02126-z

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., Fernández Del Río, J., Wiebe, M., Peterson, P., Oliphant, T. E. (2020). Array programming with NumPy. *Nature, 585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hawkyard, J. B., Johnson, W., Kirkland, J., & Appleton, E. (1973). Analyses for roll force and torque in ring rolling, with some supporting experiments. *International Journal of Mechanical Sciences, 15*, 873–893. https://doi.org/10.1016/0020-7403(73)90018-0

Ho, J., Jain A., & Abbeel P. (2006). Denoising Diffusion Probabilistic Models. In: *34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada*. arXiv:2006.11239v2.

Hua, L., Deng, J., & Qian, D. (2017). Recent development of ring rolling theory and technique. *International Journal of Materials and Product Technology, 54*(1/2/3). https://doi.org/10.1504/IJMPT.2017.080566

Hua, L., & Zhao, Z. Z. (1997). The extremum parameters in ring rolling. *Journal of Materials Processing Technology, 69*(1–3), 273–276. https://doi.org/10.1016/s0924-0136(97)00029-0

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2019). Deep learning for time series classification: A review. *Data Mining and Knowledge Discovery, 33*(4), 917–963. https://doi.org/10.1007/s10618-019-00619-1

Keogh, E. J., & Pazzani, M. J. (2000). Scaling up dynamic time warping for datamining applications. In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp 285–289. https://doi.org/10.1145/347090.347153

Kil, T. D., Lee, J. M., & Moon, Y. H. (2015). Quantitative formability estimation of ring rolling process by using deformation processing map. *Journal of Materials Processing Technology, 220*, 224–230. https://doi.org/10.1016/j.jmatprotec.2015.01.006

Kim, N., Machida, S., & Kobayashi, S. (1990). Ring rolling process simulation by the three dimensional finite element method. *International Journal of Machine Tools and Manufacture, 30*(4), 569–577. https://doi.org/10.1016/0890-6955(90)90008-7

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. In: *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp 1–15. https://doi.org/10.48550/arXiv.1412.6980

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386

Lafarge, R., Hütter, S., Tulke, M., Halle, T., & Brosius, A. (2021). Data based model predictive control for ring rolling. *Production Engineering, 15*(6), 821–831. https://doi.org/10.1007/s11740-021-01063-1

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lee, S., Park, J., Kim, N., Lee, T., & Quagliato, L. (2023). Extreme gradient boosting-inspired process optimization algorithm for manufacturing engineering applications. *Materials and Design, 226*, Article 111625. https://doi.org/10.1016/j.matdes.2023.111625

Li, S., & Zheng, P. (2020). Transfer learning for smart manufacturing: A stepwise survey. *IFAC-PapersOnLine, 53*(5), 37–42. https://doi.org/10.1016/j.ifacol.2021.04.081

Liu, Y., Fu, S., Lin, L., Zhang, S., Suo, S., & Xi, J. (2024). DECVAE: Data augmentation via conditional variational auto-encoder with distribution enhancement for few-shot fault diagnosis of mechanical system. *Measurement Science and Technology, 35*(4). https://doi.org/10.1088/1361-6501/ad197c

Mamalis, A. G., Johnson, W., & Hawkyard, J. B. (1976). Pressure distribution, roll force and torque in cold ring rolling. *Journal*

of Mechanical Engineering Science, 18(4), 196–209. https://doi.org/10.1243/JMES_JOUR_1976_018_033_02

Meert, W., Hendrickx, K., Van Craenendonck, T., Robberechts, P., Blockeel, H., & Davis, J. (2020). DTAIDistance (v2.3.10). Zenodo. https://doi.org/10.5281/zenodo.7158824

Mirandola, I., Quagliato, L., Berti, G. A., Caracciolo, R., Ryu, S. C., Perin, M., & Kim, N. (2022). Geometry evolution prediction and process settings influence in profiled ring rolling. *International Journal of Advanced Manufacturing Technology, 122*(2), 799–819. https://doi.org/10.1007/s00170-022-09928-0

Mirandola, I., Berti, G. A., Caracciolo, R., Lee, S., Kim, N., & Quagliato, L. (2021). Machine learning-based models for the estimateion of the energy consumption in metal forming processes. *Metals, 11*(5). https://doi.org/10.3390/met11050833

Modanloo, V., Jang, S., Lee, T., & Quagliato, L. (2025). Gradient Enhanced-Expert Informed Neural Network (GE-EINN) for forming depth prediction from a small-scale metal stamping dataset. *Journal of Manufacturing Processes, 140*, 224–240. https://doi.org/10.1016/j.jmapro.2025.02.052

Moser, T., Seitz, J., & Kuhlenkötter, B. (2025). Integration of machine learning methods to calculate the remaining useful life of mandrels a feasibility study. *ZWF Zeitschrift Fuer Wirtschaftlichen Fabrikbetrieb, 120*(s1), 186–191. https://doi.org/10.1515/zwf-2024-0118

Oddiraju, M., Cleeman, J., Malhotra, R., & Chowdhury, S. (2025). A differentiable physics-informed machine learning approach to model laser-based micro-manufacturing process. *Journal of Manufacturing Science and Engineering, 147*(5), Article 051002. https://doi.org/10.1115/1.4067355

Odena, A., Olah, C., & Shlens, J. (2017). Conditional image synthesis with auxiliary classifier GANs. In: *34th International Conference on Machine Learning, ICML 2017*, vol 6, pp 4043–4055. https://doi.org/10.48550/arXiv.1610.09585

Ogoke, F., Liu, Q., Ajenifujah, O., Myers, A., Quirarte, G., Malen, J., Beuth, J., & Barati, F. A. (2024). Inexpensive high fidelity melt pool models in additive manufacturing using generative deep diffusion. *Materials & Design, 245*, Article 113181. https://doi.org/10.1016/j.matdes.2024.113181

Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22(10):1345–1359. https://doi.org/10.1109/TKDE.2009.191

Parvizi, A., & Rohani Raftar, H. R. (2019). Application of artificial neural network and genetic algorithm to predict and optimize load and torque in T-section profile ring rolling. *Proceedings of the Institution of Mechanical Engineers, Part c: Journal of Mechanical Engineering Science, 233*(17), 5966–5976. https://doi.org/10.1177/0954406219861989

Parvizi, A., Abrinia, K., & Salimi, M. (2011). Slab analysis of ring rolling assuming constant shear friction. *Journal of Materials Engineering and Performance, 20*, 1505–1511. https://doi.org/10.1007/s11665-010-9824-9

Petrik, J., Kavas, B., & Bambach, M. (2023). MeltPoolGAN: Auxiliary classifier generative adversarial network for melt pool classification and generation of laser power, scan speed and scan direction in laser powder bed fusion. *Additive Manufacturing, 78*(November), Article 103868. https://doi.org/10.1016/j.addma.2023.103868

Peyré G., Cuturi M. (2018). Computational Optimal Transport. arXiv. https://doi.org/10.48550/arXiv.1803.00567

Quagliato, L., Seitz, J., & Perin, M. (2025). Machine learning modeling for material science and manufacturing: Overview and perspectives for the future. *AI Materials, 1*(1), 1–12.

Quagliato, L., & Berti, G. A. (2017). Temperature estimation and slip-line force analytical models for the estimation of the radial forming force in the RARR process of flat rings. *International Journal of Mechanical Sciences,*, 123. https://doi.org/10.1016/j.ijmecsci.2017.02.008

Quagliato, L., Berti, G. A., Kim, D., & Kim, N. (2018). Slip line model for forces estimation in the radial-axial ring rolling process. *International Journal of Mechanical Sciences, 138–139*(August 2017), 17–33. https://doi.org/10.1016/j.ijmecsci.2018.01.025

Ra, D., Lee, J., Lee, M., Kwak, S., Lee, S., & Lee, S. (2026). Physics-informed machine learning across manufacturing processes: Recent advances, challenges, and directions. *Journal of Manufacturing Systems, 85*, 72–95. https://doi.org/10.1016/j.jmsy.2026.01.002

Raftar, H. R. R., & Parvizi, A. (2018) Prediction and optimization of load and torque in ring rolling process through development of artificial neural network and evolutionary algorithms. *Journal of Computational Applied Mechanics, 49*(2), 292–303. https://doi.org/10.22059/jcamech.2018.246800.215

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics, 378*, 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

Restrepo, J. P., Rivera, J. C., Laniado, H., Osorio, P., & Becerra, O. A. (2023). Nonparametric Generation of Synthetic Data Using Copulas. *Electronics (Switzerland), 12*(7). https://doi.org/10.3390/electronics12071601

Ruediger-Flore, P., Glatt, M., Hussong, M., & Aurich, J. C. (2023). CAD-based data augmentation and transfer learning empowers part classification in manufacturing. *International Journal of Advanced Manufacturing Technology, 125*(11–12), 5605–5618. https://doi.org/10.1007/s00170-023-10973-6

Seitz, J., Moser, T., Fahle, S., Prinz, C., & Kuhlenkötter, B. (2023). Transfer learning approaches in the domain of radial-axial ring rolling for machine learning applications. In: *Proceedings of the Conference on Production Systems and Logistics*, pp 447–457. https://doi.org/10.15488/15303

Sikder, M. F., Ramachandranpillai, R., & Heintz, F. (2025). TransFusion: Generating long, high fidelity time series using diffusion models with transformers. *Machine Learning with Applications, 20*, Article 100652. https://doi.org/10.1016/j.mlwa.2025.100652

Srinivasan, P., Knottenbelt, W. J. (2022). Time-series transformer generative adversarial networks. arXiv:2205.11164v1

Stent M. (2024). Dynamic Time Warping (DTW) Tutorial. https://github.com/markstent/dynamicTimeWarping

Sun, Y., Tao, H., & Stojanovic, V. (2025a). Pseudo-label guided dual classifier domain adversarial network for unsupervised cross-domain fault diagnosis with small samples. *Advanced Engineering Informatics, 64*, Article 102986. https://doi.org/10.1016/j.aei.2024.102986

Sun, Y., Tao, H., Ni, Y., & Stojanovic, V. (2025b). A generic single-source domain generalization framework for fault diagnosis via wavelet packet augmentation and pseudo-domain generation. *IEEE Internet of Things Journal, 12*(15), 31629–31642. https://doi.org/10.1109/JIOT.2025.3573752

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., van Mulbregt, P., & SciPy 1.0 Contributors (2020). SciPy 10: fundamental algorithms for scientific computing in Python. *Nature Methods, 17*(3), 261–272. https://doi.org/10.1038/s41592-019-0686-2

Wang, Z. W., Fan, J. P., Hu, D. P., Tang, C. Y., & Tsui, C. P. (2010). Complete modeling and parameter optimization for virtual ring rolling. *International Journal of Mechanical Sciences, 52*(10), 1325–1333. https://doi.org/10.1016/j.ijmecsci.2010.06.008

Wang, X., Hua, L., Wang, X., Mei, X., Zhu, Q., & Dai, Y. (2017). Soft measurement model of ring's dimensions for vertical hot ring rolling process using neural networks optimized by genetic algorithm. *Journal of Central South University, 24*(1), 17–29. https://doi.org/10.1007/s11771-017-3404-1

Yang L, Zhang D, Karniadakis GE (2020) Physics-informed generative adversarial networks for stochastic differential equations. SIAM Journal on Scientific Computing 42(1):A292–A317. https://doi.org/10.1137/18M1225409

Yea, Y., Ko, Y., Kim, N., & Lee, J. (2003). Prediction of spread, pressure distribution and roll force in ring rolling process using rigid-plastic finite element method. *Journal of Materials Processing Technology, 0136*(03), 478–486. https://doi.org/10.1016/S0924-0136(03)00721-0

Yi, H., Hou, L., Jin, Y., Saeed, N. A., Kandil, A., & Duan, H. (2024). Time series diffusion method: A denoising diffusion probabilistic model for vibration signal generation. *Mechanical Systems and Signal Processing, 216*(March), Article 111481. https://doi.org/10.1016/j.ymssp.2024.111481

Yonekura, K. (2024). A short note on physics-guided GAN to learn physical models without gradients. *Algorithms*. https://doi.org/10.3390/a17070279

Yoon, J., Jarret, D., van der Schaar, M. (2019) Time-series generative adversarial networks. In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Article No.: 494, pp 5508–5518. https://doi.org/10.5555/3454287.3454781

Yu, Y., & Liu, Y. (2024). Physics-guided generative adversarial network for probabilistic structural system identification. *Expert Systems with Applications, 239*, Article 122339. https://doi.org/10.1016/j.eswa.2023.122339