Published in: Key Engineering Materials
ISSN: 1662-9795, Vol. 1049, pp 13-28
doi:10.4028/p-B4tPZr

Submitted: 2025-12-18
Revised: 2026-02-12
Accepted: 2026-03-02
Online: 2026-04-15

Proxy-Physics-Informed and Transfer Learning Networks for Radial-Axial Ring Rolling (RARR) under Varying Data Availability

Luca Quagliato[1,a*], Johannes Seitz[2,b], Mattia Perin[3,c], Dominik Malejka[2,d], Tobias Moser[2,e], Caterina Zanella[1,f], and Bernd Kuhlenkötter[2,g]

[1]Department of Industrial Engineering, University of Trento, 38123, Trento, Italy

[2]Chair of Production Systems, Ruhr-University Bochum, 44780, Bochum, Germany

[3]Department of Management and Engineering, University of Padova, 36100, Vicenza, Italy

[a]luca.quagliato@unitn.it, [b]seitz@lps.ruhr-uni-bochum.de, [c]mattia.perin@phd.unipd.it, [d]malejka@lps.ruhr-uni-bochum.de, [e]moser@lps.rub.de, [f]caterina.zanella@unitn.it, [g]kuhlenkoetter@lps.rub.de

Keywords: Radial-Axial Ring Rolling (RARR), Proxy-Physics Informed Neural Networks (P-PINNs), Transfer Learning, Gematrical Modeling, Forces Modeling.

Abstract. This contribution presents a proof-of-concept and benchmark analysis between two different approaches for multiple domain modeling, employing numerical and experimental results relevant to the Radial-Axial Ring Rolling (RARR). First, a finite element method (FEM) simulation data-based Deep Neural Network (DNN) was employed as base model for a subsequent transfer learning (TL), carried out by employing 40 experimental (EXP) data belonging to a different RARR domain. The DNN-TL model is benchmarked against the proposed solution, in terms of a Proxy-Physics Informed Neural Network (P-PINN), defined using proxy equations for the outer diameter (OD) expansion, the radial force (RF), and the axial force (AF) and trained on 30 FEM and 30 EXP data. The proxy equations for the P-PINN are based on known knowledge and analytical formulations developed for the RARR. The results show that employing the whole 218 cases strong FEM database for the training of the base DNN model results in high prediction accuracy on the FEM cases and can be adapted well through TL. When employing only 30 FEM cases for the training of the base DNN model results in slight improvements for RF and in a significant drop of performance for AF, after TL. Instead, in a limited data scenario, the P-PINN model, powered equally by data and proxy equations steering the learning process, is capable of modeling well and simultaneously both FEM and EXP cases with reasonable accuracy, averaging at 3.7% for OD and 24.0% for RF/AF.

Introduction

Radial-Axial Ring Rolling (RARR) enables the production of seamless rings for machinery, metallurgy and aerospace applications [1, 2]. Historically, RARR research progressed from analytical models to increasingly detailed finite element method (FEM) simulations [3-9]. In parallel, manufacturing research has recently shifted toward data-driven modeling, where machine learning (ML) is used to support process understanding, prediction and optimization of outputs such as geometry, loads, energy consumption and defect minimization [10, 11]. For RARR, ML has been applied to predict loads and torque [12] and energy consumption [13]. Moreover, Mirandola et al. [13] reported that the high process complexity and large number of independent variables present in RARR databases can cause relatively simple ML architectures to break down when experimental variability is introduced.

Although ML can provide quick and flexible surrogates, its accuracy in interpolation, and especially extrapolation, depends strongly on the quality, variance and coverage of the training data [10, 11]. Consequently, recent manufacturing-oriented ML studies increasingly emphasized careful data pre-processing and structuring, and the use of hybrid formulations that introduce physical or expert information during training [14-16]. For instance, Lee et al. [14] used Gaussian Process Regression (GPR) to support deep-drawing blank design, demonstrating how probabilistic models can guide design decisions while quantifying uncertainty. Similarly, Lee et al. [15] proposed an

Extreme Gradient Boosting (XGB) inspired optimization algorithm aimed at RARR applications, exemplifying current trends toward combining powerful regressors with explicit optimization loops. Complementarily, Modanloo et al. [16] introduced a Gradient-Enhanced Expert-Informed Neural Network (GE-EINN) that translates expert knowledge into differential constraints, improving convergence and extrapolation accuracy also for small-scale EXP and FEM hybrid datasets.

Within the ring rolling domain, the technical depth of data-driven workflows has similarly increased while remaining grounded in manufacturing targets. Lee et al. [15] coupled an XGB regressor with a Differential Evolution (DE) optimizer to create case-specific processing maps, focusing on minimizing mandrel work for flat rings. The XGB-DE model was trained on FEM data and validated against experimental cases. Lafarge et al. [17] explored data-based model predictive control for ring rolling, integrating measured process signals with predictive models to support closed-loop operation. Fahle et al. [18] addressed quality monitoring by modeling ring ovality from rolling time-series data through a semi-supervised scheme and further employed a Generative Adversarial Network (GAN) to augment scarce datasets with synthetic samples and reported the ability to generate distinct samples with comparable variance. Finally, Cappellini et al. [19] formulated a multi-objective optimization workflow that trades off production energy and time using surrogate response surfaces, illustrating how validated FEM models can be leveraged to generate structured databases for process analysis and optimization.

Besides the groundwork laid so far, a key issue for deployment is model longevity [11]. Trained models often degrade when applied to new mills, products or operating windows, making domain adaptation essential [10, 11]. Transfer learning (TL) addresses this challenge by reusing a model trained in a source domain and fine-tuning it using comparatively limited data from a target domain. In RARR, Seitz et al. [20] demonstrated TL across two mills with a pronounced size ratio (20:1), reducing training effort for the target configuration; the study also discussed the use of GAN-generated synthetic data to alleviate underfitting in low-data scenarios. However, TL remains fully data-driven, and its success depends on having a robust pre-trained backbone and sufficient variance in the target domain database [10, 11]. Synthetic data generation and oversampling can be practical for regularization and dataset expansion [21, 22], but generated samples typically remain close to the distribution of the data used to train the generator. Hence, synthetic data cannot reliably create new information density in poorly sampled regions. To further reduce dependence on large datasets, physics-informed neural networks (PINNs) have been proposed, adding residual terms derived from governing equations to the loss function [23]. Expert-informed variants, such as GE-EINNs [16], extend this idea by encoding heuristic trends as differentiable constraints. However, the lack of differentiable equations typical of RARR, and other manufacturing processes, limits the direct applicability of standard PINNs/EINNs as benchmarks for the multi-output prediction problem considered in this research.

To address these limitations, this contribution propose a Proxy-Physics Informed Neural Network (P-PINN) that follow the PINN training philosophy [23] but replace exact governing equations with proxy relations encoding *cause-effect* trends between features and target variables. These proxy residuals provide a tunable bias through the loss function and offer a control for model customization, with the aim of improving generalization in low-data and cross-domain settings. The proposed P-PINN was benchmarked against a supervised DNN with transfer learning (DNN-TL). The base DNN models were trained on FEM datasets of varying size and then fine-tuned using experimental data, while performances were evaluated for three key process outputs: OD, RF) and AF using mean average percentage error (MAPE), together with training time and model complexity. Results show that when a sufficiently large FEM database is available, DNN-TL delivers the lowest deviations (≈3.7% for OD, ≈12.7% for RF and ≈17.3% for AF). Conversely, when only 30 FEM cases are available, the base DNN becomes unreliable for force prediction (deviations >60%), and TL inherits this weakness, leading to noisy AF predictions (≈36% MAPE). Under the same low-data scenario, P-PINNs trained with 30 FEM and 30 EXP cases use the proxy residuals as an initial backbone and achieve more consistent cross-domain predictions (≈3.7% OD and ≈23–25% RF/AF), positioning P-PINNs as an alternative to TL in the case of data scarcity or PINN framework incompatibility.

FEM and EXP database generation

To train the models proposed in this research time-series belonging to both FEM simulations and RARR experiments were employed. The thermo-mechanical FEM simulations were implemented in the Simufact Forming 15 by employing a model developed by one of the authors and extensively validated against analytical solutions and experimental results [5, 6], as in Fig. 1(a).

In all simulations the following settings were employed: 325 mm for main roll radius, 125 mm for the mandrel radius, 596 mm for the axial rolls' length and 35° for their vertex angle, 175 mm for the radius of the centering rolls, 150°C for the tools' temperature and 50°C for the environmental temperature. As per the ring dimensions, the following ranges were considered: 429.3~990.2 mm and 211.0~646.6 mm for the outer and inner diameters, 112.2~423.0 mm for the height for the preforms and 556.7~1992.8 mm and 397.0~1812.9 mm for the outer and inner diameters, 99.8~388.0 mm for the height for the final rings, for a total of 218 FEM simulations.

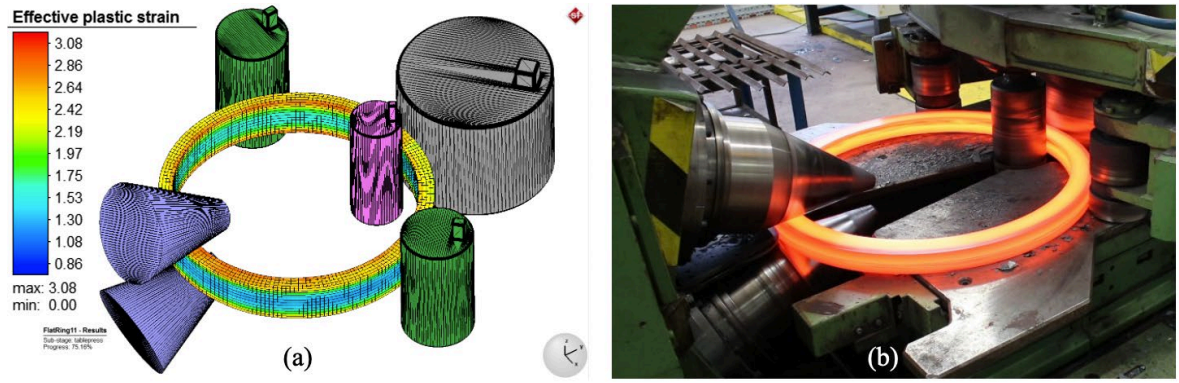

Fig. 1. (a) Example of FEM model implementation with results for the RARR process and (b) RARR process experiment at the Chair of Production Systems of Ruhr-University Bochum.

As materials, a 42CrMo4 medium-carbon low-alloy steel, an E355 carbon-steel, a S355-J2 structural grade steel, an AlMgSi1 6000 series aluminum, and a nickel-based Inconel 718 superalloy were employed. The material properties of all five materials employed in the FEM simulations were modeled by employing the Hensel-Spittel (H-S) flow stress, Eq. (1), summarized by material in Table 1. In the FEA simulations considered in this research, a shear friction law was employed by setting the friction factor to $m = 0.85$ for the contact between the ring and the mandrel and the ring and the main roll whereas to $m = 0.6$ for the contact between the ring, axial rolls and centering rolls [5, 6].

$$\sigma_f = C_1 \cdot e^{C_2 \cdot T} \cdot \varepsilon^{(n_1 \cdot T + n_2)} \cdot e^{(L_1 \cdot T + L_2)} \cdot \dot{\varepsilon}^{(m_1 \cdot T + m_2)} \; [MPa] \qquad (1)$$

Table 1. Hensel-Spittel flow stress material constants employed in the FEM simulations.

	Constants							
Materials	C_1	C_2	n_1	n_2	L_1	L_2	m_1	m_2
42CrMo4	5290.5	-3.7E-3	-3.3E-4	2.1E-1	-8.3E-5	2.9E-2	3.0E-4	-1.6E-1
E355	2549.5	-3.4E-3	-3.2E-4	1.9E-1	-4.9E-5	1.1E-2	1.2E-4	-4.3E-1
S355-J2	2478.7	-2.9E-3	-4.2E-4	3.7E-01	-7.5E-5	3.2E-2	2.6E-4	-1.4E-1
AlMgSi1	378.5	-4.9E-3	-1.1E-4	-2.6E-2	6.0E-5	-2.6E-2	3.5E-4	-3.2E-2
Inconel 718	10501.1	-3.1E-3	-1.8E-4	5.4E-1	-2.2E-5	2.4E-2	-2.7E-6	9.8E-2

The 218 FEM cases were discretized with 3D 8-node brick hexahedra (MARC® element type 7) using a 16/12/16 mm (axial/radial/tangential) mesh with elements' number scaling with the initial ring size and no remeshing. Simulations were solved with MUMPS using a Lagrangian formulation

for large deformations. Contact employed a 1/20 element-size criterion, and relative velocities < 0.001 mm/s were treated as sticking conditions. Each run took ~10 h and the cases originate from a repository at the University of Padova [13]. Full settings are available from the corresponding author.

The experimental database comprises 40 rolling series, 17 from the Chair of Production Systems (Ruhr-University Bochum) and 23 from an industrial collaboration at the University of Padova (mill details confidential). The former tests were run on a ThyssenKrupp Wagner mill (20/16-1000/160; max forces 200 kN radial and 160 kN axial; capacity ≈ 1 m diameter and 160 mm height). Data were acquired via the CARWIN control system: RF/AF through analog-to-digital converters, ring thickness/diameter via an ultrasonic displacement transducer and an elastically supported metering roll, and process time from the PLC sampling rate. Each case provides force/geometry time series plus static initial/final geometries, material properties and process parameters. All experiments used S235 steel with H-S model constants (Table 1): 1967.97, −3.0E−8, −6.9E−4, 0.58, −9.9E−5, 0.049, 4.2E−4, −0.27. Preforms covered OD 262~2148 mm, inner diameter 158~1357 mm, height 71~515 mm; final rings OD 387~4192 mm, ID 289~3741 mm, height 51~450 mm.

For each FEM and EXP case, the same feature set composed of initial and final ring geometry (outer diameter, wall thickness and height), billet temperature, main-roll rotational speed during rolling, and material encoded by H-S coefficients was collected. The targets in terms of outer diameter (OD) evolution and the radial (RF) and axial (AF) force time series were collected as well. Figure 2 reports the corresponding FEM and EXP distributions as mean curves with *Min–Max* envelopes.

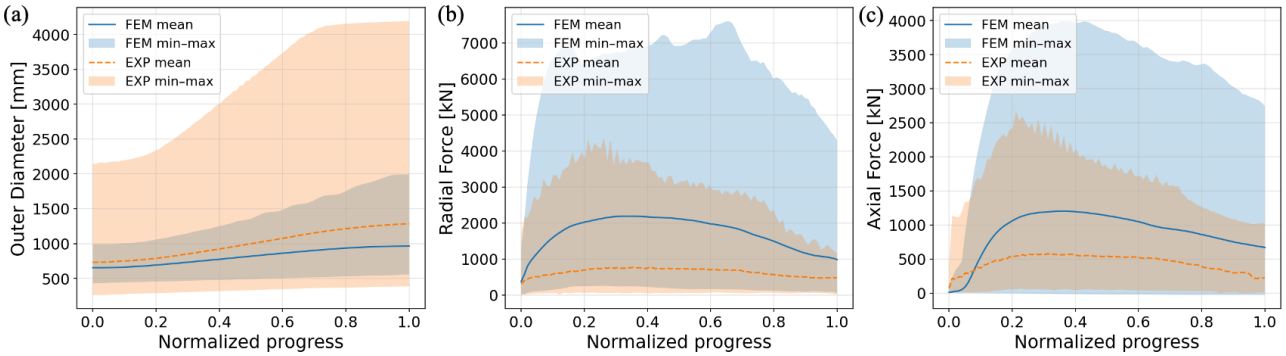

Fig. 2. Mean curves and *Min-Max* ranges for FEM and EXP time-series for three target variables employed in this study, namely (a) OD, (b) RF, and (c) AF.

Base DNN and transfer learning models

Preprocessing. Since ring-rolling FEM time series differ in duration, sampling rate, and noise, each raw simulation was converted into a uniform fixed-size sample. Process inputs (main-roll rotational speed, mandrel feed, upper axial-roll feed, and initial ring temperature) were time-normalized and sampled at 32 evenly spaced instants, while targets (OD, RF, AF) resampled at 100 instants. Static descriptors, initial/final ring geometry (OD, thickness, height) and material encoding via the flow-stress law of Eq. (1), were appended, while all signals smoothed with a Savitzky–Golay filter.

DNN for Multi-Output Prediction (base DNN). A feed forward DNN, as the one employed in [16] is employed in this research. For numeric stability, RF and AF were mapped through an *asinh* transform, as in Eq. (2), where y is the physical force value in kN, y' is the transformed value, and $s \geq 1$ is the scaling coefficient.

$$y' = asinh\left(\frac{y}{s}\right) \rightarrow y = s \cdot sinh\left(\frac{y}{s}\right) \quad (2)$$

Differently from the DNN in [16], the loss function employed in this research utilizes some specific features to help steering the learning process, as hereafter detailed. First, considering the substantial difference between OD and RF/AF behaviors, while all losses are based on the mean

square error (MSE), their formulations differ. As concerns OD, the loss is defined as in Eq. (3) while the single components are detailed in Eq. (4).

$$Loss_{(OD)} = L_{base} + L_{BC} + L_{span} + L_{area} + L_{mono} + L_{curv} \tag{3}$$

$$L_{base} = \frac{1}{N}\sum_k (\hat{y}_k - y_k)^2 \quad ; \quad L_{BC} = (\hat{y}_0 - y_0)^2 + (\hat{y}_{N-1} - y_{N-1})^2$$

$$L_{span} = \left(\hat{y}_{N-1} - \hat{y}_0 - (y_{N-1} - y_0)\right)^2 \quad ; \quad L_{area} = \left(\sum_k \hat{y}_k - \sum_k y_k\right)^2 \tag{4}$$

$$L_{mono} = \sum_k max(0, y_k - y_{k+1})^2 \quad ; \quad L_{curv} = \sum_k (\hat{y}_{k+1} - 2\hat{y}_k + \hat{y}_{k-1})^2$$

For OD, the loss for the predicted curve $\hat{y}$ combines six terms to enforce accuracy and physical plausibility, namely: a pointwise MSE term L_{base} against the target y, boundary constraints L_{BC} to match known initial/final OD, and L_{span} to prevent cumulative drift. Global behavior is additionally controlled through an area-matching term L_{area}, a monotonicity penalty L_{mono} that discourages negative OD increments, not physically possible in RARR, and a curvature term L_{curv} that suppresses higher-order oscillations, consistent with the expected pseudo-linear OD expansion trend. For RF and AF, the loss is tailored to emphasize the mid-cycle *active* region where forces matter most. Thus, points between 25% and 75% of normalized time are weighted more, Eq. (5), sudden peaks are smoothed/stabilized, Eq. (6), and L_{area} is also included. Hence, performance is then evaluated as *MAPE@active* over that *active* region avoiding excessively effort on fitting the near-zero regions.

$$Lw_{(RF/AF)} = \left(\frac{\sum_k w_k(\hat{y}_k - y_k)^2}{\sum_k w_k}\right) \; with \; 0.25 \leq w_k \leq 0.75 \tag{5}$$

$$Lp_{(RF/AF)} = [max_k(\hat{y}_k) - max_k(y_k)]^2 \tag{6}$$

The three encoded output sequences, OD residual, *asinh*-transformed RF, and *asinh*-transformed AF, are concatenated into a single multi-output prediction vector, employed during the learning process of the base DNN model, and subsequently also in the TL stage.

Transfer learning (TL) module. Considering the feed forward DNN model, as presented in the previous section, to adapt it to the EXP domain, a transfer learning module was developed. The TL workflow predicts first a *zero-shot prediction* aimed at understanding the differences between the origin (FEM) and target (EXP) domains, and then progressively updates the DNN weights and biases to adapt them to the target domain. To ensure compatibility between FEM-trained models and EXP training data, all EXP targets are transformed into the same encoded space used during FEM training. The OD curve is first residualized as in Eq. (7) while RF and AF are transformed by employing the same *asinh* scaling coefficients, as in Eq. (8), respectively.

$$OD_{res}^{EXP}(k) = OD^{EXP}(k) - OD^{FEM}(k) \tag{7}$$

$$RF_{tr}^{EXP}(k) = asinh\left(\frac{RF^{EXP}(k)}{s_{RF}}\right) \quad ; \quad AF_{tr}^{EXP}(k) = asinh\left(\frac{AF^{EXP}(k)}{s_{AF}}\right) \tag{8}$$

The OD baseline $OD^{FEM}(k)$ is defined as the linear interpolation between the known initial and final ring diameters and serves as an initial guess of the trajectory for this target variable. By considering Eq. (7) and (8), the TL training proceeds in the same latent space previously defined during the base DNN training, meaning that the domain is adapted rather than retrained. After

transforming each experimental target curve into the same encoded representation used during FEM training, the three sequences are concatenated into a single target vector, as in Eq. (9), standardized by using the FEM latent space statics in terms of mean ($\boldsymbol{\mu}_y^{FEM}$) and variance ($\boldsymbol{\sigma}_y^{FEM}$), as in Eq. (10).

$$y_{tr}^{EXP} = \left[OD_{res,tr}^{EXP},\ RF_{tr}^{EXP},\ AF_{tr}^{EXP}\right] \in \mathbb{R}^{3,N} \tag{9}$$

$$y_{std}^{EXP} = \frac{y_{tr}^{EXP} - \boldsymbol{\mu}_y^{FEM}}{\boldsymbol{\sigma}_y^{FEM}} \tag{10}$$

At this point, fine-tuning is applied by constructing TL heads for each of the target variables, namely OD, RF, and AF, separately. It should be noted that even though separate sections of the network are devoted to different target variables, the model is still a multivariate regression algorithm. Given a standardized EXP input ($\boldsymbol{X}_{std}^{EXP}$), the TL phase aims to adjust the heads, defined by the parameter θ_g, so that the predictions match the encoded experimental targets y_{std}^{EXP}, as per Eq. (11).

$$\hat{z}_g^{EXP} = f_g\left(\boldsymbol{X}_{std}^{EXP}, \theta_g\right)\ with\ g \in \{OD, RF, AF\} \tag{11}$$

At this point, the losses are constructed with the same strategy employed for the base DNN model and reported in Eq. (3) and (4), yet the main aim changes. For OD, their purpose shifts from enforcing global consistency to applying localized corrections that compensate for small experimental deviations while for RF and AF, since EXP forces contain longer low-force phases and sharper active windows, the weighting isolates the physically meaningful segment more aggressively than in the FEM stage, preventing the model from being dominated by negligible force regions.

Proxy-Physics Informed Neural Networks (P-PINNs) model

P-PINN governing equations. This section presents all the details relevant to the P-PINN model, developed for this research due to the lack of RARR-related physical equations translatable into learnable constraints. Moreover, as previously mentioned, EINN formulations [16] were also not suitable for the task at hand considering the high dimensionality of features and target variable that are described in $\mathbb{R}^{3,N}$ and not in $\mathbb{R}^{2,N}$, as in [16]. In this regard, the P-PINN model is grounded on two separate input groups, employed for the prediction of OD, RF, and AF. The former, defined as in Eq. (12), comprises the so-called *deformation sequence*, composed of the resampled rotational speed $rpm(k)$, mandrel feeding distance $\Delta T(k)$, upper axial roll feeding distance $\Delta H(k)$, and resampled axial height (H(k)) calculated from the feeding speed. The latter, as presented in Eq. (13), is a *scalar descriptor* including the initial/final geometry of the ring ($D_0, T_0, H_0, D_f, T_f, H_f$), a material strength proxy (S_{proxy}), and the 8 constants for the Hensel-Spittel flow stress model ($HS_{1,}, \dots, HS_{8,}$), previously defined in Eq. (1). $\boldsymbol{X}_{seq}$ and $\boldsymbol{x}_{scal}$ of Eq. (12) and (13) are then stacked in a unified latent vector, as in Eq. (14), together with the labels identifying the material ($\boldsymbol{e}_{mat}$) and the domain ($\boldsymbol{e}_{dom}$), either FEM or EXP, for data management and plotting purposes.

$$\boldsymbol{X}_{seq} = [rpm(k), \Delta T(k), \Delta H(k), \mathrm{H}(k)]^T \in \mathbb{R}^{4,N} \tag{12}$$

$$\boldsymbol{x}_{scal} = \left[D_0, T_0, H_0, D_f, T_f, H_f, S_{proxy}, HS_{1,}, \dots, HS_{8,}\right] \tag{13}$$

$$\boldsymbol{z} = \left[\boldsymbol{X}_{seq}, \boldsymbol{x}_{scal}, \boldsymbol{e}_{mat}, \boldsymbol{e}_{dom}\right] \tag{14}$$

P-PINN OD modeling. After constructing the training dataset, the learning process is different for OD and for RF/AF. For the former, rather than predicting $OD(k)$ directly as a single sequence, the P-PINN model predicts incremental radial growth and then constructs OD from these increments. First, through a small MLP (multi-layer perceptron), the OD raw head is constructed to roughly

estimate how much OD should grow at each step based on the information in $\boldsymbol{z}$, as in Eq. (15). For all means and purposes, Eq. (15) represents a standard data-driven learning process, as the one shown for the base DNN in the previous section.

$$\Delta\mathrm{OD}_i^{raw} = head_{OD}(\boldsymbol{z})_i \;\; with \;\; i = 1, \dots, N \tag{15}$$

However, the estimation of Eq. (15) is completely unconstrained, meaning that $\Delta\mathrm{OD}_i^{raw}$ could be negative, not following the true $D_0 - D_f$ behavior or inconsistent OD domain. To avoid this, physical-proxy constraints are applied to $\Delta\mathrm{OD}_i^{raw}$ to both make it physically meaningful and learn the patterns of the EXP domain. The constraints of Eq. (16), enforced sequentially, ensures that the predictions are positive and follow a positive growth for OD, as per Eq. (17). For OD, the P-PINN predicts incremental radial growth from a preliminary shape evolution ($\Delta\mathrm{OD}_i^{raw}$) by enforcing physical-proxy constraints, as per Eq. (16) at model level in the forward pass of the network rather than through PDE residuals or soft physics penalties typical of standard PINNs.

$$\Delta\mathrm{OD}_i = softplus(\Delta\mathrm{OD}_i^{raw}) > 0 \quad ; \quad \Delta\mathrm{OD}_i{}^{*} = \Delta\mathrm{OD}_i \frac{D_f - D_0}{\sum_{j=1}^{N} \Delta\mathrm{OD}_i} \tag{16}$$

$$\mathrm{OD}(k) = D_0 + \sum_{i=1}^{k} \Delta\mathrm{OD}_i{}^{*} \tag{17}$$

P-PINN RF/AF modeling. For RF and AF, proxy modeling is formulated more explicitly than for OD, reflecting their different feature dependence and relying on prior analytical RARR force models developed by the authors [5, 6]. In P-PINNs, physics is not imposed through PDE residuals or loss penalties, instead, the governing structure is embedded as *forward-pass proxies*, so that the proxies capture the dominant geometric and material drivers while the data-driven residual component mainly provides fine tuning**.** Accordingly, the RF proxy in Eq. (18) is expressed as a weighted combination of the principal contributors, namely the thickness variation $\Delta T(k)$, diameter growth $\Delta OD(k)$, and main-roll speed $p_m(k)$, scaled by an effective flow stress σ_{eff} computed from the H-S law. The weights $\omega_1, \omega_2, \omega_3$ and a small scaling factor s_{soft}^{RF} are learned to calibrate the RF magnitude. As concerns AF, the modeling is similar is shape but different in meaning than that of RF. As shown in Eq. (19), the *ReLU* function is not included in $\mathrm{AF}_{proxy}(k)$ to allow for negative values in case of specific machine or simulation settings.

$$RF_{proxy}(k) = \sigma_{eff}[\omega_1 ReLU\{\Delta T(k)\} + \omega_2 ReLU\{\Delta OD(k)\} + \omega_3 ReLU\{\mathrm{rpm}(k)\}] \cdot s_{soft}^{RF} \tag{18}$$

$$AF_{proxy}(k) = 0.75 \cdot \sigma_{eff}[\omega_4 c(k) + \omega_5 \Delta OD(k) + \omega_6 \mathrm{rpm}(k)] \cdot s_{soft}^{AF} \tag{19}$$

Moreover, $AF_{proxy}(k)$ is also based on the outer diameter $\Delta OD(k)$ and main roll rotational speed $\mathrm{rpm}(k)$, as $RF_{proxy}(k)$, but includes $c(k) = max(H_0 - H_k, 0)$, defined as the *compression gap*. In this regard, $c(k)$ quantifies the instantaneous axial displacement between the ring and the axial roll, estimated through the upper axial roll feeding speed, and serves as the primary driver of axial force in the Proxy-PINN by activating AF only when true geometric compression, and not noise, occurs. It ought to be pointed out that $\Delta OD(k)$ in Eq. (18) and (19) is estimated through Eq. (17), thus it is an internal model estimation and is not based on the ground truth OD. Accordingly, RF and AF are predicted, at each epoch of the calculation, by a mix of data- and physics-driven equations (20).

$$\mathrm{RF}(k) = \mathrm{RF}_{proxy}(k) + g_{RF} \cdot \Delta RF(k) \;\; ; \;\; \mathrm{AF}(k) = \mathrm{AF}_{proxy}(k) + g_{AF} \cdot \Delta \mathrm{AF}(k) \tag{20}$$

In Eq. (20), $RF_{proxy}(k)$ and $AF_{proxy}(k)$, are fully proxy-driven and serve as the backbone of the learning process, while the latter parts of both equations are DNN residual-based. The data-driven contribution is learned, through the scalars g_{RF} and g_{AF}, during the training of the model.

In terms of losses, since most of the constraints are applied at model level in the forward section of the network, the loss for OD is only based on the batch size (B) and the number of discretized points (N), as in Eq. (21), which is based on an MSE.

$$Loss_{(OD)}^{data} = \frac{1}{BN}\sum_{j=1}^{B}\sum_{k=1}^{N}\left(\widehat{OD}_s^{(j)}(k) - OD_s^{(j)}(k)\right)^2 \tag{21}$$

As concerns RF and AF, they use the same data loss structure on scaled outputs, as in Eq. (22), thus only the former is hereafter summarized. First, the data-driven part of the loss is constructed considering discrepancy between predicted and measured RF and AF curves using an *asinh*-scaled MSE with FEM/EXP domain-dependent weights as $\omega_{FEM}^{(j)} = 0.5$ and $\omega_{EXP}^{(j)} = 2$ and where a_{rf} is the *asinh* scaling constants used to normalize the force magnitudes before applying MSE. The *asinh* scaling reduces the dynamic range of RF/AF and improves numerical conditioning for large and noisy force peaks. The second part of the loss is represented by the P-PINN consistency term, where the predictions are steered to be closer to the surrogate proxies. The utilization of this additional control over the learning process applied at loss level required the utilization of the PINN module over the standard DNN module. For RF, and for AF, the proxy part of the loss is defined as in Eq. (23). The last part of the loss is introduced to stabilize and capture physical smoothness of force evolution, thus the main behavior over noise, as per Eq. (24).

$$\begin{gathered}\widehat{RF}_s^{(j)}(k) = asinh\left(\frac{RF^{(j)}(k)}{a_{rf}}\right) \\ so\ that:\ Loss_{(RF)}^{data} = \frac{1}{B}\sum_{j=1}^{B}\omega_{dom}^{(j)}\left[\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{RF}_s^{(j)}(k) - RF_s^{(j)}(k)\right)^2\right]\end{gathered} \tag{22}$$

$$Loss_{(RF)}^{proxy} = \frac{1}{BN}\sum_{j=1}^{B}\sum_{k=1}^{N}\left(\widehat{RF}_s - \widehat{RF}_{s,proxy}\right)^2 \tag{23}$$

$$Loss_{(RF,AF)}^{smooth} = \left\|\Delta^2\widehat{RF}_s\right\|_2^2 + \left\|\Delta^2\widehat{AF}_s\right\|_2^2 \tag{24}$$

In conclusion, the total loss for the P-PINN model is the sum of $Loss_{(OD)}^{data}$ of Eq. (21), the data-driven loss for RF and AF as shown for the former in Eq. (22) in terms of $Loss_{(RF)}^{data}$, the proxy PINN part of RF $Loss_{(RF)}^{proxy}$ as in Eq. (23), and the corresponding $Loss_{(AF)}^{proxy}$ part constructed in the same way, and the $Loss_{(RF,AF)}^{smooth}$ part of Eq. (24) helping to promote a behavior towards the main trend.

Models' Pre-Processing and Tuning

This last section before the results chapter summarizes all the technical information related to the base DNN, the TL, and the P-PINN models defined in the previous sections. All computations were performed in Python on a MacBook Pro (Apple M4), without external GPU acceleration.

DNN for Multi-Output Prediction (base DNN). The base DNN operates on the FEM-only matrix considering a combination of scalars, namely the initial/final geometry of the ring, the initial ring temperature, and time-sequences describing the main roll rotational speed, the thickness reduction rate and height compression rate, namely the mandrel and upper axial roll feeding speeds. The network outputs the full trajectories of OD, RF and AF on a fixed grid of N = 100 points per target,

i.e. 300 output channels and inputs are standardized with a *StandardScaler*. OD is handled in physical units by subtracting a linear baseline (added back after inference). RF/AF are first scaled by a per-group factor, then mapped with an *asinh* transform to compress noisy peaks and finally standardized. Training is performed in this transformed space, and predictions are post-processed by de-standardization, inverse-*asinh*, and reconstruction in physical units. Hyperparameters are tuned separately for each target, consistently with the different pre-processing approaches, using a grid search of 60 trials combined with $k = 3$ cross-validation and a 90%/10% train/validation split. Given the sparsity of EXP data, no independent test set is held out during tuning. In contrast, the selected model is evaluated across 5 random seeds to assess generalization. The same tuning protocol is used for DNN-TL and P-PINN to ensure a fair comparison. Here follows the search ranges, while the full grid available on request. OD: hidden units 256–512, dropout 0.00–0.35, learning rate 3×10^{-5}–5×10^{-3}, weight decay 10^{-7}–10^{-3}, batch size 48–256, epochs 450–1500. RF: hidden units 384–1024, dropout 0.00–0.40, learning rate 3×10^{-5}–10^{-2}, weight decay 10^{-8}–10^{-3}, batch size 64–384, epochs 450–1500. AF: hidden units 384–512, dropout 0.05–0.25, learning rate 10^{-4}–3×10^{-3}, weight decay 10^{-7}–10^{-4}, batch size 64–192, epochs 600–1200. All runs used the *AdamW* optimizer coupled with an early stopping to limit overfitting and select the best model based on the validation performance.

Transfer learning (TL) module. The transfer learning (TL) stage adapts the pre-trained base DNN to the experimental (EXP) domain while preserving the structure learned from the FEM simulations. TL is performed separately for OD, RF, and AF output heads. Inputs are normalized using the original FEM-trained *StandardScaler*, and the TL targets are standardized using the stored FEM statistics for each group. OD is fine-tuned in residual form, exactly as in pre-training, while RF and AF remain in the *asinh*-scaled space. Each TL head is trained on an 80%/20% split of the filtered EXP dataset using *AdamW* with learning rate 3×10-4, weight decay 10-5, batch size 128, and gradient-norm clipping at 1.0. Training proceeds for a maximum of 1,000 epochs with early stopping based on the validation loss and a patience of 35~60 epochs. After TL, predictions are de-standardized, inverse-transformed, and reconstructed in physical units. For the TL model, a manual hyperparameters tuning was carried out on a subset of those employed for the base DNN model.

P-PINN model. The P-PINN is trained on 30 and 30 randomly sampled FEM and EXP cases. Inputs consist of a *deformation sequence* and *scalar descriptors*. The sequence block is encoded by a 1D CNN with three Conv1d layers (4→32→64→128 channels) and *ReLU* activations. Material and domain labels are appended to tag each case during learning. As for the base DNN, OD is kept in physical units, while RF/AF are mapped to *asinh* space. In the total loss, AF is prioritized by using weights of 8:1 vs OD and 4:1 vs RF. Optimization uses *AdamW* for stability under mixed data-driven and proxy-physics terms and consistency with DNN-TL, with gradient-norm clipping = 5 to avoid instabilities. The best model selection retains the parameters at the minimum validation data-loss**.** Hyperparameters were tuned with *Optuna* under 3-fold cross-validation, rather than grid search, because P-PINN hyperparameters are more strongly coupled due to loss-weight scheduling and proxy-term interactions. Each trial samples: learning rate **[**5×10^{-4}, 3×10^{-3}] (log-uniform), weight decay [10^{-6}, 10^{-3}] (log-uniform), batch size {4, 8, 12}**,** and epochs {4000, 12000} in 1000-step increments. The intervention/decay schedule for $Loss_{RF}^{proxy}$ and $Loss_{RF,AF}^{smooth}$ is also optimized via $\lambda_{start}^{proxy} \in [0.1,0.6]$and $\lambda_{end}^{proxy} \in [0.02,0.3]$, with the schedule switch epoch selected from {500, 750, 1000, 1250, 1500, 1750, 2000} to control when training transitions from capturing global behavior to correcting smaller deviations. The *Optuna* objective is the fold-average of *MAPE@active* summed over OD, RF, AF**,** evaluated on the full FEM and EXP sets. The best setting yields lr ≈ 6.35×10^{-4}**,** wd ≈ 6.4×10^{-6}**,** B = 4**,** 12,000 epochs, with $\lambda_{start}^{proxy} \approx 0.34$, $\lambda_{end}^{proxy} \approx 0.03$, and a switch at epoch 2000. The model is then retrained once on the full FEM+EXP training set with this configuration, and the best checkpoint is used for benchmarking against DNN-TL.

Results

Given the large result set, each model is presented through representative OD/RF/AF charts and a numerical comparison based on *MAPE@active* averaged over five runs and five random seeds. The chapter compares the base DNN trained on 218 vs 30 FEM cases, the corresponding TL models adapted with 40 EXP cases, and a P-PINN trained on a balanced 30 FEM + 30 EXP subset, to contrast data-rich and data-limited regimes and benchmark P-PINN against DNN/TL under both.

Base DNN performance. Two base-DNN configurations were evaluated: training on the full FEM set (218 simulations) and on a reduced subset (30 randomly selected cases). Both tests were repeated five times with different random seeds and train/validation splits to assess repeatability. Fig. 3 shows representative validation cases, while performance is reported as *MAPE@active*.

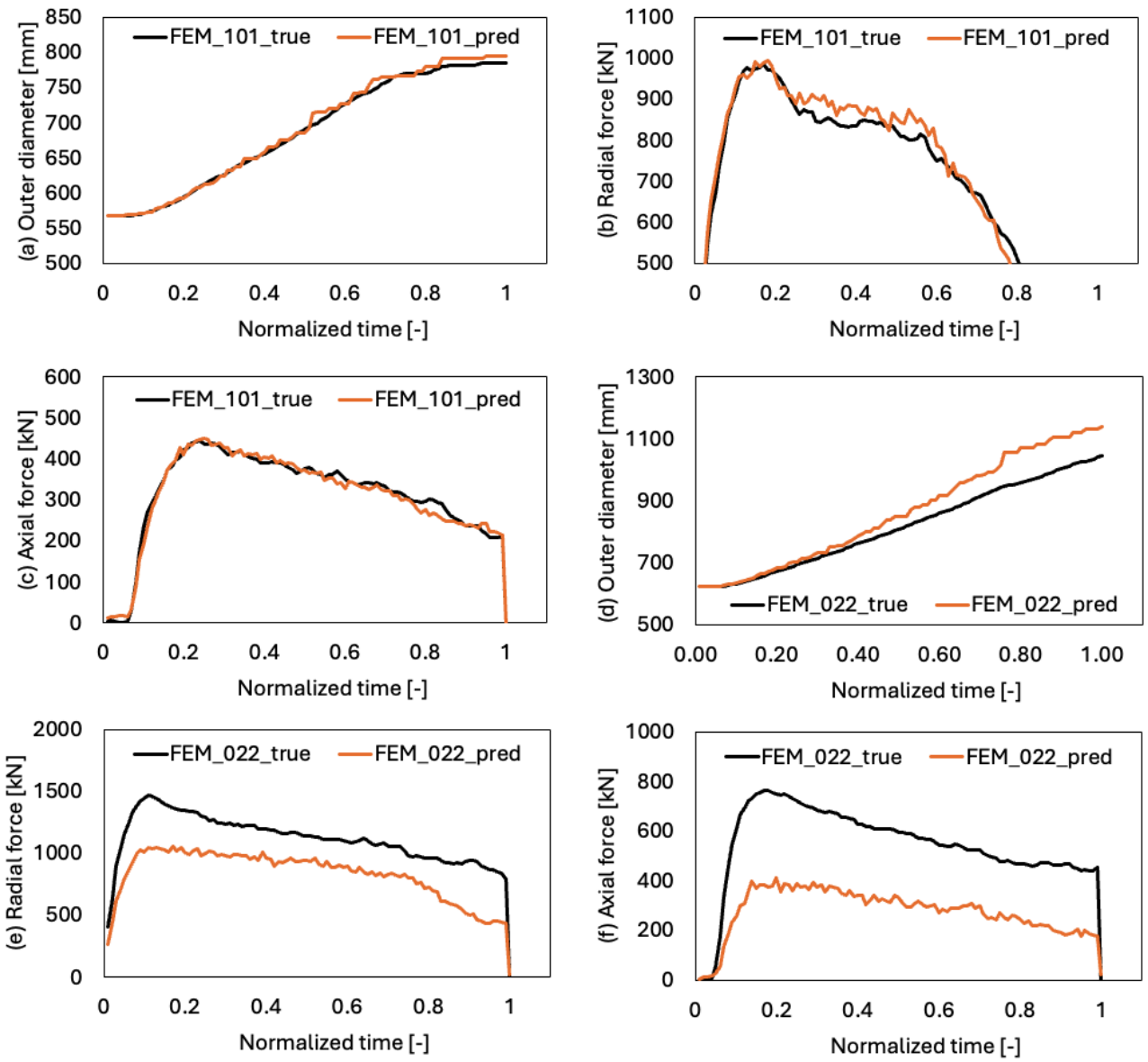

Fig. 3. Examples of true vs predicted curves by the base DNN model trained with 218 cases for (a) the outer diameter OD, (b) the radial force RF, and (c) the axial force AF considering 218 input cases. (d), (e), and (f) show the same comparison but considering 30 training cases for the base DNN model.

Prior to consolidation, five FEM files were discarded because some features could not be reliably extracted/encoded, mainly due to unresolved material-name mapping, leaving 213 FEM cases for the "full-data" training. Across the five seeded trials, the DNN trained on 213 FEM cases predicts all three targets well, achieving a *MAPE@active* of 1.98% (OD), 15.5% (RF), and 17.4% (AF). Reducing

training data to 30 FEM cases degrades performance considerably, Fig. 3(d)–(f). OD remains reasonable but is slightly overestimated (6.2%), whereas RF and AF are strongly underestimated, with 62.8% and 63.9% *MAPE@active*, respectively. Overall, the results confirm that purely data-driven accuracy depends critically on database size/coverage, motivating hybrid formulations, *id est* P-PINN, for low-data or boundary scenarios.

DNN-TL performance. As concerns the TL application, the two base DNN models reported in the previous section were employed by applying the same 40 cases strong EXP database. In this regard, the first attempt is related to the utilization of the base DNN model trained with 218 FEM cases, which showed the best performance in its own latent space, as reported in Fig. 4.

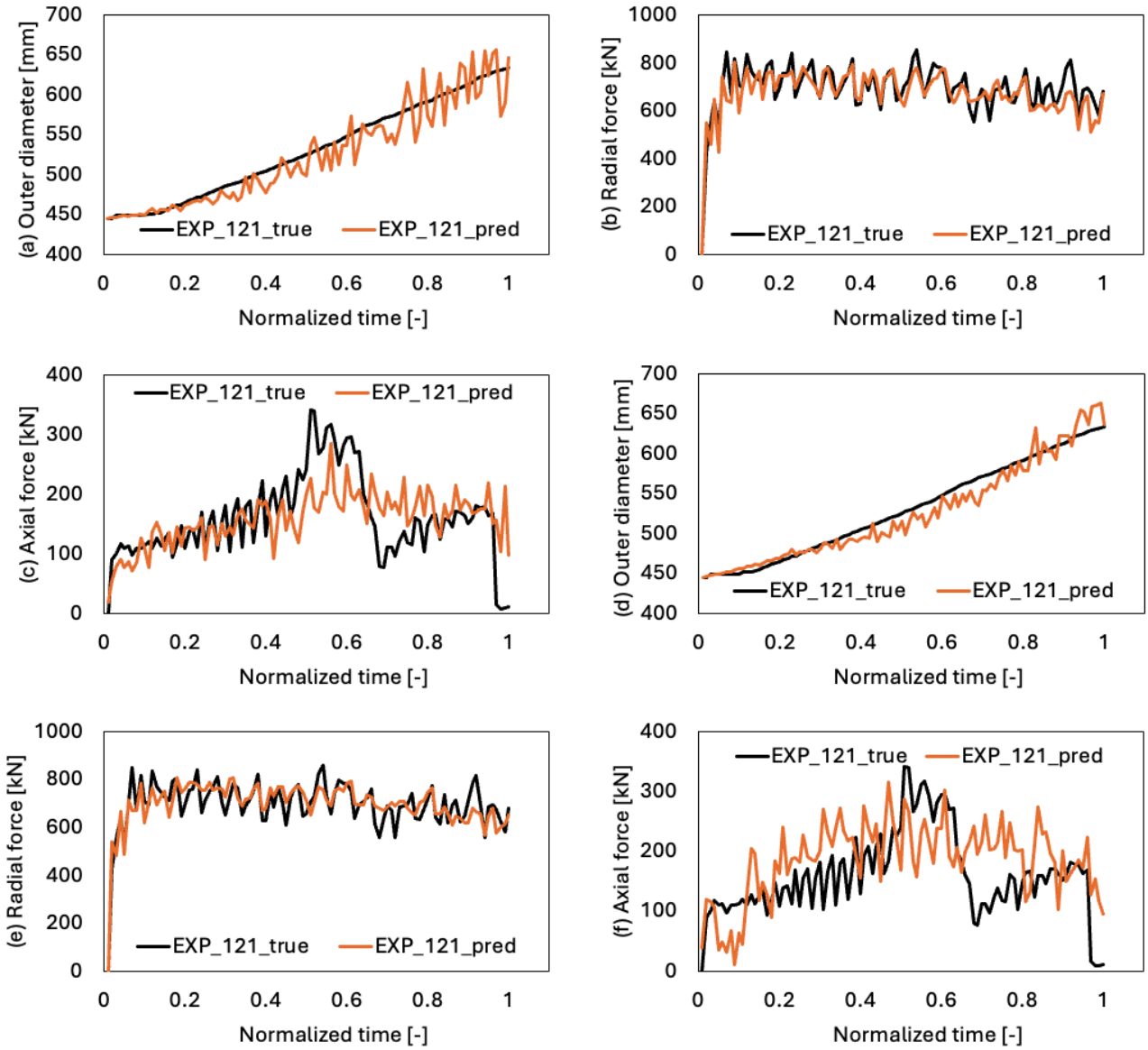

Fig. 4. True vs predicted curves by the DNN-TL model for an example EXP case (#121) for (a) the outer diameter OD, (b) the radial force RF, and (c) the axial force AF considering 218 input cases. (d), (e), and (f) show the same comparison but considering 30 cases for the training of the base DNN.

The above-mentioned model, hereafter referred to as DNN(218)-TL, showed comparable performance in comparison to the base DNN model from which it is transfer learned, being the *MAPE@active*, for the three target variables, equal to 3.7% OD, 12.69% for RF, and 17.3% for AF, respectively. The results for an example case are reported in Fig. 4(a), (b), and (c). However, when the base DNN model is switched from DNN(218)-TL, to the one trained with only 30 FEM cases, defined hereafter as DNN(30)-TL, an interesting scenario arises. First, thanks to its monotonic

behavior and the similarity between the FEM and EXP domains for this target variable, OD seems to be insensitive to the change in the base DNN model with the *MAPE@active* changing to 3.2. On the other hand, while RF predictions improve to 10.6%, those for AF degrade to 36.3%. Since all the results of Fig. 4 refer to the same case, by comparing Fig. 4(c) with Fig. 4(f) is it possible to grasp the increase noise in the predicted curve, a symptom of the lack of stability in the base DNN model.

Large AF errors are not purely computational but reflect the intrinsic complexity of axial loading in RARR. AF arises from contact between curved, out-of-plane ring and axial-roll surfaces, where force transmission depends on rapidly changing contact states and secondary deformation. These mechanisms are weakly constrained when the base DNN is trained on only a few FEM cases, and the mismatch is amplified in experiments, making AF particularly sensitive to the representational capacity inherited by TL. Consistently, applying the 30-FEM base DNN directly to EXP data yields *MAPE@active* = 20.3% (OD) but extreme errors for forces (>800% RF, ~300% AF), highlighting a strong FEM–EXP domain gap not evident from Fig. 2 alone.

Overall, the 218-FEM base DNN is a robust FEM-domain regressor and its DNN(218)-TL remains reliable for OD/RF/AF. With only 30 FEM cases, the base model captures mainly the global trends, making TL more flexible and even improve RF adaptation, but it does not provide a sufficiently structured backbone to transfer-learn AF, whose performance degrades relative to DNN(218)-TL. Hence, while TL can still work under limited FEM data, the weak FEM representation particularly limits AF transfer. Hence, the proposed P-PINN is introduced next to jointly model FEM and EXP within a single, proxy-constrained framework.

P-PINNs performance. Since the P-PINN is trained on a randomly selected 30 FEM + 30 EXP mix, the results in this section refer to both numerical and experimental cases, which are learned and predicted simultaneously by a single model. Representative validation curves from the five repetitions (different random seeds) are shown in Fig. 5, while the mean performance across seeds is MAPE@active = 3.66% (OD), 22.9% (RF), and 25.1% (AF). Notably, RF and AF errors are comparatively balanced, addressing the DNN(30)-TL behavior where RF was adapted much better than AF. However, deviations remain higher than DNN(218)-TL, indicating that proxy relations alone cannot fully compensate for the information density provided by the full FEM database used to build the TL backbone.

Comparing the FEM example in Fig. 5(a)–(c) with the EXP example in Fig. 5(d)–(f) shows that the proxy equations can represent the FEM domain, but joint FEM+EXP training prevents the *build in source domain → adapt to target* advantage of DNN-TL, which mainly affects the EXP accuracy. For OD, enforcing known initial/final geometry, as in analytical approaches [5, 6] translated in here into proxy forms, stabilizes the trajectory but may overconstrain the mid-expansion evolution, producing larger mid-phase errors for smoother expansions, Fig. 5(d). For RF/AF, predictions are substantially less noisy than DNN(30)-TL, yet they can still exhibit sharp peaks and misestimate rises/drops, suggesting that limited data coverage cannot be fully replaced by proxies. Nonetheless, the model captures the overall trends and peak magnitudes of both forces with reasonable accuracy across both FEM and EXP, using a single formulation.

Although the P-PINN model is more accurate in the presence of small-scale databases, it is also more computationally demanding, requiring a proxy equations' definition phase, similarly to EINN [16], a more complex tuning, which was carried out by the *Optuna* hyperparameters optimization scheme, and a running time of approximately 30 minutes, in contrast to the 10 minutes of the base DNN trained on 218 cases and TL which is almost real time.

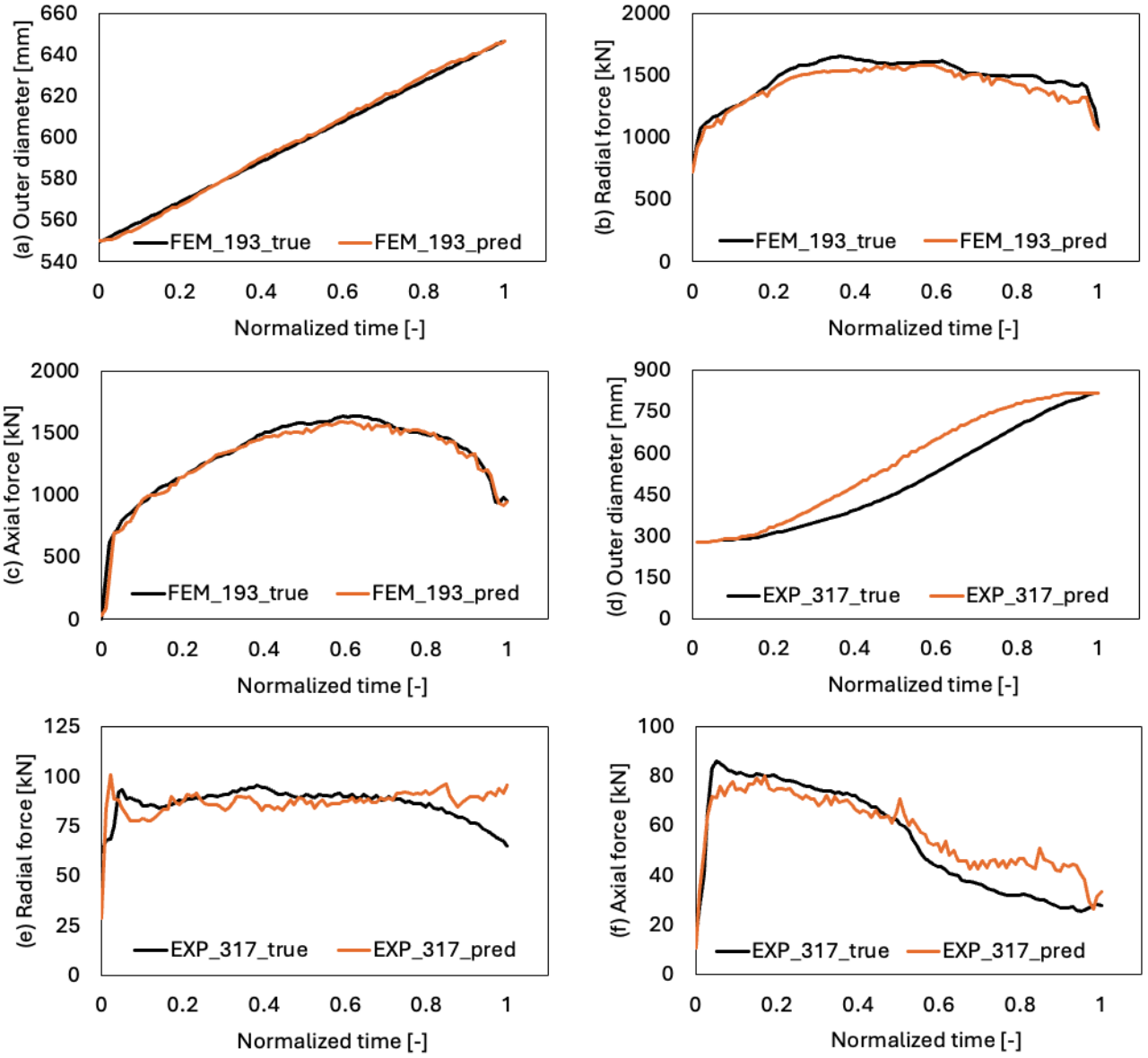

Fig. 5. True vs predicted curves by the P-PINN model for an example FEM (#193) and one EXP case (#317) for (a)-(d) the outer diameter OD, (b)-(e) the radial force RF, and (c)-(f) the axial force AF considering 30 FEM and 30 EXP cases for the training.

Discussion

This section expands on the previous results chapter by discussing the advantages, limitations, and improvement paths of the proposed P-PINN formulation, and by providing practical guidance on when to prefer P-PINNs *versus* transfer learning.

Consistent with prior TL studies in RARR [20], DNN(218)-TL achieves higher accuracy with lower modeling complexity than P-PINN. This outcome is expected since proxy equations capture dominant trends, but cannot reproduce the full variability embedded in a large well-contructed database. The broad geometric/material coverage of the 218-case FEM set is therefore central to the robustness of TL across source and target domains. Conversely, when data are limited and the feature-target variables relationships are highly nonlinear, typical for RARR [13], purely data-driven models become difficult to train reliably, especially for the most complex targets, such as AF. In this low-data regime, hybrid approaches can provide useful robustness by combining prior knowledge with data, at the cost of higher model complexity and computational effort. Since proxies are not exact closed-form laws, they also require a design phase to avoid overwhelming the data-driven component.

In this regard, and from a modeling perspective, P-PINNs use the proxy terms as an initial structural backbone that guides optimization toward physically admissible solutions, while later training stages rely increasingly on data-driven residuals for fine-scale corrections. Operationally, this behaves like an integrated transfer-learning scheme where the *base model* is given by the proxies, and *domain adaptation* is performed by the residual network. In this work, this was implemented through a two-stage proxy schedule λ_{start}^{proxy} and λ_{end}^{proxy}, with λ_{start}^{proxy} about an order of magnitude larger, effectively introducing *curriculum learning*. Before the switch, a stronger proxy coupling biases training toward proxy-consistent behavior to establish the global trends. Then, after the switch (2,000 epochs in this study) the proxy influence is relaxed so the data can refine deviations rather than reconstruct the overall dynamics from scratch. Because the balance between proxy and data residuals is problem-dependent, both λvalues were tuned via *Optuna*.

Considering what stated in the last paragraph, the degradation of AF in DNN(30)-TL is not only a matter of data volume, but also of losing latent geometric structure that the base DNN learns when trained on the full FEM set. In contrast, P-PINN partially preserves this architecture through the proxy structure, yielding more balanced RF/AF errors under limited data. In addiiton to that, the slightly better RF adaptation observed in DNN(30)-TL relative to DNN(218)-TL is consistent with a known TL effect where a mild underfitting in the source model can increase flexibility in the target domain, as also reported in other TL contexts [24, 25]. However, when underfitting becomes severe, as for AF, transfer performance deteriorates, leading to high deviations in both the source and target domains.

In this regard, and conceptually speaking, P-PINNs can be seen as embedded TL where proxies act as a fixed, non-trainable prior, while the residual network performs data-driven adaptation. Accordingly, P-PINNs are not intended to replace TL when large, high-quality FEM databases are available. Rather, they provide a viable option when data scarcity**,** domain mismatch**,** or longevity requirements limit purely data-driven approaches. Overall, P-PINNs occupy a middle ground between TL and physics-informed or expert-informed modeling, offering controlled customization and robustness while still requiring data for calibration. While the present formulation remains a proof of concept, the results indicate that P-PINNs can prioritize controlled generalization over maximum achievable accuracy in practically relevant, data-limited settings.

Conclusions

This contribution detailed a proof-of-concept study for the development of customized neural networks that employs the PINNs structure while grounded on proxy equations, thus P-PINNs. The proposed P-PINN model was benchmarked against the well-established transfer learning approach. In low data scenario, for both the origin and target domains, P-PINNs allow for a better and wiser modeling, resulting in a stable training and better overall performance. On the other hand, for industrial applications, or more broadly when large datasets are available, TL is still confirmed as the tool of choice, being easier to set up and overall, better performing. To this end, some key issues raised in this contribution will require additional work in the future. First and foremost, the definition of the proxy equations would benefit from a semi-automatization, possibly based on a known set of analytical or empirical equations. Moreover, the threshold underlining the choice between P-PINN and TL should also benefit from a deeper and more systematic modeling, allowing end-users to select the best modeling approach based on model selection maps, helping in this initial yet critical choice.

References

[1] J. Seitz, Q. Wang, L. Goldbach, T. Moser, A. Brosius, B. Kuhlenkötter. A Review of Data-Centric Optimization and Machine Learning in Ring Rolling, in: Drossel, WG., Ihlenfeldt, S., Dix, M. (eds) Production at the Leading Edge of Technology. WGP 2024. Lecture Notes in Production Engineering. Springer, Cham, 2025, pp. 242-252.

[2] J.M. Allwood, A.E. Tekkaya, T.F. Stanistreet, The development of ring rolling technology, Steel Res. Int. 76 (2005) 111-120.

[3] J.B. Hawkyard, W. Johnson, J. Kirkland, E. Appleton, Analyses for roll force and torque in ring rolling, with some supporting experiments, Int. J. Mech. Sc. 15 (1973) 873–893.

[4] A.G. Mamalis, W. Johnson, J.B. Hawkyard, Pressure distribution, roll force and torque in cold ring rolling, J. Mech. Eng. Sc. 18 (1976) 196-204.

[5] L. Quagliato, G.A. Berti, D. Kim, N. Kim, Slip line model for forces estimation in the radial–axial ring rolling process, Int. J. Mech. Sc. 138–139 (2018) 17-33.

[6] G.A. Berti, L. Quagliato, M. Monti, Set-up of radial–axial ring-rolling process: Process worksheet and ring geometry expansion prediction, Int. J. Mech. Sc. 99 (2015) 58–71.

[7] L. Guo, H. Yang, Towards a steady forming condition for radial–axial ring rolling, Int. J. Mech. Sc. 53 (2011) 286–299.

[8] N. Kim, H. Kim, K. Jin, Optimal design to reduce the maximum load in ring rolling process, Int. J. Prec. Eng. Manuf. 13 (2012) 1821–1828.

[9] Z.W. Wang, J.P. Fan, D.P. Hu, C.Y. Tang, C.P. Tsui, Complete modeling and parameter optimization for virtual ring rolling, Int. J. Mech. Sc. 52 (2010) 1325-1333.

[10] J. Cao, M. Bambach, M. Merklein, M. Mozaffar, T. Xue, Artificial intelligence in metal forming, CIRP Annals – Manuf. Tech. (2024) 1–27.

[11] L. Quagliato, J. Seitz, M. Perin, Machine learning modeling for material science and manufacturing: overview and perspectives for the future, AI Mater. (2025) 0005.

[12] A. Parvizi, H.R.R. Raftar, Application of artificial neural network and genetic algorithm to predict and optimize load and torque in T-section profile ring rolling, Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sc. 233 (2019) 5966–5976.

[13] I. Mirandola, G.A. Berti, R. Caracciolo, S. Lee, N. Kim, L. Quagliato, Machine learning-based models for the estimation of the energy consumption in metal forming processes, Metals 11 (2021) 833.

[14] S. Lee, Y. Lim, L. Galdos, T. Lee, L. Quagliato, Gaussian process regression-driven deep drawing blank design method, Int. J. Mech. Sc. 265 (2024) 108898.

[15] S. Lee, J. Park, N. Kim, T. Lee, L. Quagliato, Extreme gradient boosting-inspired process optimization algorithm for manufacturing engineering applications, Mat. & Des. 226 (2023) 111625.

[16] V. Modanloo, S. Jang, T. Lee, L. Quagliato, Gradient enhanced-expert informed neural network (GE-EINN) for forming depth prediction from a small-scale metal stamping dataset, J. Manuf. Proc. 140 (2025) 224–240.

[17] R. Lafarge, S. Hütter, M. Tulke, T. Halle, A. Brosius, Data-based model predictive control for ring rolling, Prod. Eng. 15 (2021) 821–831.

[18] S. Fahle, T. Glaser, A. Kneißler, B. Kuhlenkötter, Improving quality prediction in radial-axial ring rolling using a semi-supervised approach and generative adversarial networks for synthetic data generation, Prod. Eng. 16 (2022) 175–185.

[19] C. Cappellini, C. Giardini, S. Bocchi, A multi-objective optimization workflow of ring-rolling process parameters based on production energy and time, Procedia CIRP 122 (2024) 683–688.

[20] J. Seitz, T. Moser, S. Fahle, C. Prinz, B. Kuhlenkötter, Transfer learning approaches in the domain of radial-axial ring rolling for machine learning applications, Proceedings of the 5th Conference on Production Systems and Logistics (CPSL) (2023).

[21] L. Quagliato, S. Kim, O.R. Hassan, T. Lee, Heel pad's hyperelastic properties and gait parameters reciprocal modelling by a Gaussian mixture model and extreme gradient boosting framework, Biomedical Signal Processing and Control 107 (2025) 107818.

[22] V. Modanloo, M. Elyasi, T. Lee, L. Quagliato, Modeling of tensile strength and wear resistance in friction stir processed MMCs by metaheuristic optimization and supervised learning, The International Journal of Advanced Manufacturing Technology (2025).

[23] S. Cuomo, V. Schiano Di Cola, F. Giampaolo, G. Rozza, M. Raissi, F. Piccialli, Scientific machine learning through physics-informed neural networks: Where we are and what's next, Journal of Scientific Computing 92 (2022) 88.

[24] A. Deng, X. Li, D. Hu, T. Wang, H. Xiong, C.-Z. Xu, Towards inadequately pre-trained models in transfer learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023, pp. 19397–19406.

[25] J.M. Springer, S. Goyal, K. Wen, T. Kumar, X. Yue, S. Malladi, G. Neubig, A. Raghunathan, Overtrained language models are harder to fine-tune, arXiv preprint arXiv:2503.19206, 2025.