Key Engineering Materials
ISSN: 1662-9795, Vol. 1050, pp 215-225
doi:10.4028/p-j0sDDJ

Submitted: 2026-01-19
Revised: 2026-02-17
Accepted: 2026-02-18
Online: 2026-04-16

Using Kolmogorov–Arnold Networks for Discrepancy Modeling of Lateral Flow in Hot Rolling of Steel Slabs

HASHEMZADEH Amirali[1,a*], BOCK Frederic E.[2,b], COMETA Antonella[1,c], SOYARSLAN Celal[1,3,d], KLUSEMANN Benjamin[2,4,e], VAN DEN BOOGAARD Ton[1,f]

[1]Chair of Nonlinear Solid Mechanics, University of Twente, The Netherlands

[2]Institute of Materials and Process Design, Helmholtz-Zentrum Hereon, Geesthacht, Germany

[3]Fraunhofer Innovation Platform, University of Twente, The Netherlands

[4]Institute for Production Technology and Systems, Leuphana University Lüneburg, Lüneburg, Germany

[a] a.hashemzadeh@utwente.nl,[b] frederic.bock@hereon.de,[c] a.cometa@utwente.nl, [d] c.soyarslan@utwente.nl,[e] benjamin.klusemann@leuphana.de,[f] a.h.vandenboogaard@utwente.nl

Keywords: Hot Rolling, Finite Element Models, Lateral Flow, Discrepancy Modeling, KANs

Abstract. Kolmogorov-Arnold networks (KANs) have emerged as a promising counterpart to multilayer perceptrons (MLPs) which offer a more interpretable functionality for different machine learning (ML) applications. Their main difference lies in the definition of KAN layers, using learnable activation functions, which has made these networks optimal for physics-based applications. In this work, we focus on analyzing the performance of KANs in capturing the physics of the hot rolling process, which is an integral part of steel manufacturing industry. Initially, we introduce non-dimensional parameters to encapsulate geometrical factors in the process. We perform space-filling sampling in the space spanned by these parameters. The sampled points yield the necessary parameters for the finite element (FE) simulations, forming the ground truth (GT) data for the network. A closed-form analytical model for spread is considered from previous studies in the literature, and its predictive performance is assessed against the FE results. In defining the input space for the network, different alternatives are compared and it was seen that input space containing the non-dimensional features and the predictions of the analytical model reduced overfitting and better generalization. The effect of KAN hyperparameters are evaluated, and the network with tuned parameters demonstrate optimal performance on the test set. Lastly, after applying symbolification for this network, a closed-form expression is obtained that captures the discrepancy between the analytical model and the GT results, and its performance is tested against test set data.

Introduction

MLPs or multilayer feedforward networks have become the essential constituents of ML applications using deep learning. This was primarily due to the universal approximation theorem [1], which states that a feedforward network with a linear output and at least one hidden layer with any type of *squishing* activation functions can approximate any continuous and Borel-measurable nonlinear function with a mapping between finite-dimensional spaces. The error of this approximation can be reduced arbitrarily if a sufficiently large number of neurons are used in the hidden layer. Often to optimize their performance, the architecture of MLPs includes multiple hidden layers with fixed activation functions, which enables a nested formulation of the problem. However, such a formulation has also brought forward various limitations associated with these networks. For instance, catastrophic forgetting in accommodating variations in the data distribution [2], prioritizing learning data with higher frequency as opposed to ones with lower frequency, a phenomenon referred to as *spectral bias* [3, 4], which results in convergence problems for the network if the problem physics includes a high frequency

solution, such as wave propagation [5]. Moreover, especially in deep neural networks (DNNs), MLPs are prone to suffer from *exploding gradients* problem which can make the use of stochastic gradient descent (SGD) unfeasible [6]. Moreover, the blackbox nature of MLPs makes them less interpretable [7].

In search of circumventing or improving existing problems with MLPs, KANs have been introduced as a promising alternative [8]. Development of KANs has been based on the Kolmogorov-Arnold's Representation Theorem (KART) [9], which as opposed to MLPs, translated into having *learnable activation functions* on the *edges* rather than *nodes*. One of the main advantages of KANs, that made them attractive alternatives to MLPs, is the fact that KANs result in interpretable models with much less parameters required for the same task [10]. Consequently, KANs have been applied to benchmark problems, commonly tested with MLPs, to compare their performance. For instance, in the study of [11], it was found that in physics-informed KAN using Jacobi orthogonal polynomials enhanced the limitations of traditional physics-informed neural networks (PINNs), together with less ill-conditioning for flow-field predictions in a fluid dynamical application. Moreover, in [12], a physics-informed KAN was proposed to predict the temperature and velocity fields in a turbulent flow using 3D experimental ground-truth (GT) data. To achieve this, the loss function included terms penalizing the residuals of the governing equations for velocity, as well as boundary conditions.

The initial version of KANs allows for great flexibility, which has been explored in different studies. For instance, the main development of KANs introduced B-spline functions together with a smoothing term as bases to expand the edge activation functions. However, other orthogonal basis functions have been implemented and compared, and each has shown to contain advantages for different applications. Chebyshev polynomials [13], ReLU [14], Jacobi polynomials [15], Gaussian radial basis function (RBF) [16], and Fourier bases [17] include some of the alternatives that have been introduced as possible replacements of B-spline functions. These alternatives have proved to reduce the computational time and the required parameters associated with B-splines while maintaining the overall network accuracy and precision for different applications. Also, in [18], a method was introduced for better tackling the model complexity in KANs, based on the differential evolution (DE) algorithm for selecting the optimal KAN structure. This method showed enhancements in convergence, prediction performance, and interpretability, compared with traditional KANs.

KANs have attracted attention for their potential benefits across different research fields, from material modeling to industrial applications. For instance, Zhang et al. in [19] have shown that integrating KANs into PINNs result in more accurate predictions of nonlinear deflections of ionic polymer-metal composites (IPMC) and an improved convergency compared with MLP-based PINNs. Also in [20], KANs were used to construct physically admissible and polyconvex free-energy functions that were later used for modeling compressible hyperelastic materials. With the aim of reducing the training time for KANs, Howard et al. [21] introduced an architecture for KANs, based on domain-decomposition of the analysis region, allowing for multiple KANs to be trained in tandem in smaller domains, yielding more accurate solutions. As another application, in [22], the graph convolutional networks (GCNs) were replaced by spline-based KAN layers, resulting in graph Kolmogorov-Arnold networks (GKANs), forming a new way of inter-layer information processing, which proved to have a superior accuracy compared to GCNs when comparable number of parameters were considered.

MLPs have been used for modeling different aspects of hot rolling process, including *lateral flow* or *spread*, which refers to the transverse deformation that occurs during thickness reduction. A hybrid CNN–long short-term memory (LSTM) model was proposed to predict lateral flow in hot strip finishing mills by integrating spatial and temporal features from rolling data [23]. Zhong et al. [24] enhanced the Shibahara spread model by incorporating equipment wear and interference factors, optimizing its parameters with Bayesian-optimized differential evolution and adaptive gradient descent, achieving a 9.77% improvement in width prediction.

In this work, we explore the potential of KANs to render interpretable models for spread in hot rolling of steel slabs. For reaching this purpose, initially, the GT data is generated via FE simulations. Next, an analytical model resulting in a closed-form solution for the spread is compared to the simulations results. Lastly, KANs are utilized for modeling the discrepancy between the analytical models and GT data, yielding interpretable models for width prediction that can serve as online monitoring tools.

Methodology

KART and KANs: The main development of KANs has been established based on KART. According to this theorem [9], every continuous multivariate function $f : [0,1]^n \to \mathbb{R}$ for $n \in \mathbb{N}$ on a close and bounded interval can be expressed as a composition of finite number of continuous functions $\phi_{i,j}$ and addition operation:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{2n+1} \Phi_i \left(\sum_{j=1}^{n} \phi_{i,j}(x_j) \right), \qquad (x_1, \ldots, x_n) \in [0,1]^n. \tag{1}$$

In this equation, $\phi_{i,j} : [0,1]^n \to \mathbb{R}$ and $\Phi_i : \mathbb{R} \to \mathbb{R}$ correspond to the continuous inner and outer map functions, respectively. Also, $\phi_{i,j}$ are global, hence they are independent of f, and Φ_i depend on the mapping f that is to be approximated. Theoretically, it is possible to express any nonlinear function f with only 2 layers, if those layers are wide enough; however, this might make the $\phi_{i,j}$ functions non-smooth and hence pathological [25]. One of the main innovations that was proposed in [8] was how to make a KAN layer and deeper KAN networks. Using the global nature of the inner functions $\phi_{i,j}$, B-spline functions were used as bases to expand them. The number of the chosen splines (N) and their order (k) are considered as tunable hyperparameters:

$$\phi(x) = \sum_{n=0}^{N-1} c_n B_n^k(x), \tag{2}$$

where $B_n^k(x)$ represents the n^{th} B-spline of order k and c_n's are parameters that are optimized in the back propagation. At one KAN layer, the ranges of considered grid points are updated according to the outputs of the previous layer. Moreover, to gain more control over the magnitudes of the resulting activation functions, a residual term was introduced in the expansion of the inner functions:

$$\tilde{\phi}(x) = \sum_{n=0}^{N-1} c_n B_n^k(x) + w_b \frac{x}{1+e^{-x}}, \tag{3}$$

where w_b is another trainable parameter. Between the nodes of KAN layers l and $l+1$, the inputs are transformed via the inner functions to form the outputs, as it can be seen in Fig. 1.

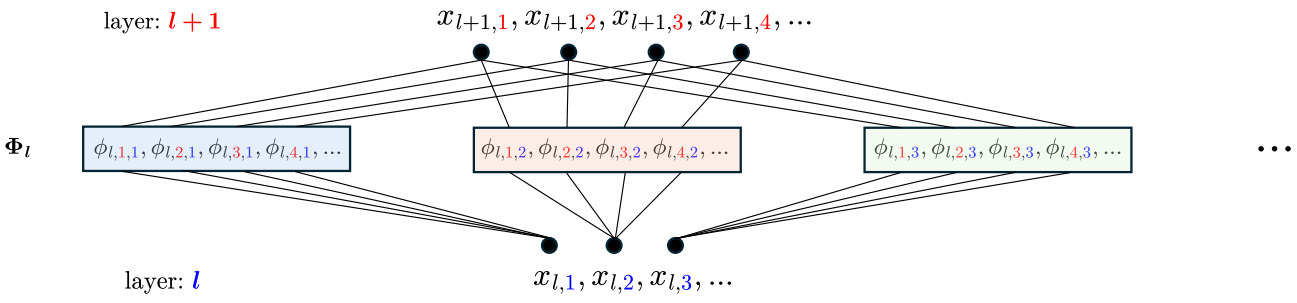

Fig. 1: Structure of a KAN layer.

If the nodes on layer l and $l+1$ are represented by $\boldsymbol{x}_l$ and $\boldsymbol{x}_{l+1}$, respectively, where:

$$\boldsymbol{x}_l := \left[x_{l,1}, x_{l,2}, x_{l,3}, \dots\right]^T, \quad \boldsymbol{x}_{l+1} := \left[x_{l+1,1}, x_{l+1,2}, x_{l+1,3}, x_{l+1,4}, \dots\right]^T.$$

then, the transformation equation between these two layers can be written as:

$$\boldsymbol{x}_{l+1} = \underbrace{\begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix}}_{:=\boldsymbol{\Phi}_l} \boldsymbol{x}_l \Rightarrow x_{l+1,j} = \sum_{i=1}^{n_l} \phi_{l,j,i}\left(x_{l,i}\right). \tag{4}$$

where for layer l, n_l represents the number of inputs and $\boldsymbol{\Phi}_l$ defines the collective activation functions. As a result, the overall mapping for the KAN network containing a total number of L layers performing on the input vector $\mathbf{x}_{\text{inp}}$ can be expressed as:

$$\text{KAN}\left(\boldsymbol{x}_{\text{inp}}\right) = \left(\boldsymbol{\Phi}_L \circ \boldsymbol{\Phi}_{L-1} \circ \cdots \circ \boldsymbol{\Phi}_2 \circ \boldsymbol{\Phi}_1\right)\left(\boldsymbol{x}_{\text{inp}}\right). \tag{5}$$

During the training of the KAN networks, for having sparse and simpler models, the concepts of *sparsification* and *pruning* were introduced to the network. In the sparsification of KAN networks, the L_1 norm was initially defined for the individual and collected inner activation functions over the edges. Within a KAN layer l with n_{inp} and n_{out} number of inputs and outputs, respectively:

$$\|\phi\|_1 := \frac{1}{n_{\text{inp}}} \sum_{m=1}^{n_{\text{inp}}} \left|\phi(x_m)\right|, \quad \|\boldsymbol{\Phi}_l\|_1 := \sum_{i=1}^{n_{\text{inp}}} \sum_{j=1}^{n_{\text{out}}} \left\|\phi_{l,i,j}\right\|_1. \tag{6}$$

Also, an entropy regularization term for gaining a sufficient effect on sparsity was introduced:

$$S(\boldsymbol{\Phi}_l) := -\sum_{i=1}^{n_{\text{inp}}} \sum_{j=1}^{n_{\text{out}}} \frac{\left\|\phi_{l,i,j}\right\|_1}{\|\boldsymbol{\Phi}_l\|_1} \ln\left(\frac{\left\|\phi_{l,i,j}\right\|_1}{\|\boldsymbol{\Phi}_l\|_1}\right). \tag{7}$$

As a result, for a KAN with a total of L layers, the total loss objective $\mathcal{L}$ is the sum of the network prediction loss $\mathcal{L}_{\text{pred}}$ and the regularization loss.

$$\mathcal{L} = \mathcal{L}_{\text{pred}} + \lambda \left[\mu_1 \sum_{l=1}^{L} \|\boldsymbol{\Phi}_l\|_1 + \mu_2 \sum_{l=1}^{L} S(\boldsymbol{\Phi}_l)\right]. \tag{8}$$

For a regression task with a total number of N samples, if we denote the GT target with y and if $(\bullet)^{(i)}$ represents the i^{th} sample, $\mathcal{L}_{\text{pred}}$ reads:

$$\mathcal{L}_{\text{pred}} := \frac{1}{N} \sum_{i=1}^{N} \left(\text{KAN}\left(\boldsymbol{x}_{\text{inp}}^{(i)}\right) - y^{(i)}\right)^2. \tag{9}$$

The overall model sparsity is dependent on the parameter λ in Eq. 8. In addition to regularization, pruning is performed on the node level as a strategy to detect the nodes that do not contribute to the overall network performance. For doing so, for each node i at layer l, the incoming and outgoing inner activation functions are evaluated, and the maximum value is taken for all the input and output connections (represented by $I_{l,i}$ and $O_{l,i}$ in Fig. 2). The node is deactivated if the difference between these scores are less than a threshold value, resulting in smaller overall networks. The value of the threshold is considered a hyperparameter.

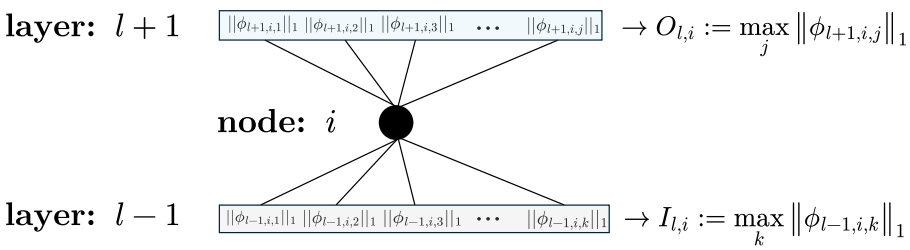

Fig. 2: Pruning of KANs at node level.

FE simulations:

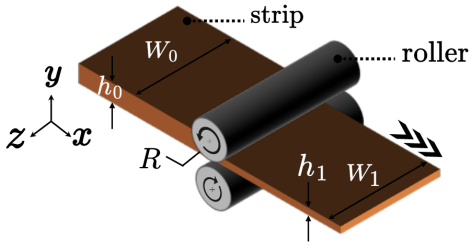

Fig. 3: The geometric parameters of the rolling process.

For generating the GT data, initially, the process parameters, including the initial width (W_0), initial and final thicknesses (h_0 and h_1, respectively), and rolling radius (R) were transformed into non-dimensional parameters using Buckingham-Π theorem [26]. The resulting three dimensionless groups $x_1 := W_0/h_0$, $x_2 := h_0/R$, and $x_3 := h_1/h_0$ also define the input space for the KANs, with bounds $x_1 \in [1, 5]$, $x_2 \in [0.1, 0.25]$, and $x_3 \in [0.5, 1)$. In the second step, to capture the full problem non-linearity resulting from the FE simulations, a space-filling sampling was performed in the (x_1, x_2, x_3) space within the considered intervals. The space-filling quality of a particular sampling is quantified via its *central discrepancy* (CD) value [27] as the lower values correspond to better space-filling quality. We considered Latin Hypercube Sampling (LHS), Sobol, Halton, and random sampling methods, and it was observed that for lower sample sizes, Sobol sampling resulted in a lower CD value, which made it the optimal choice for the subsequent analysis (see Fig.4). Also, we considered a total of 128 sample points for the subsequent analysis.

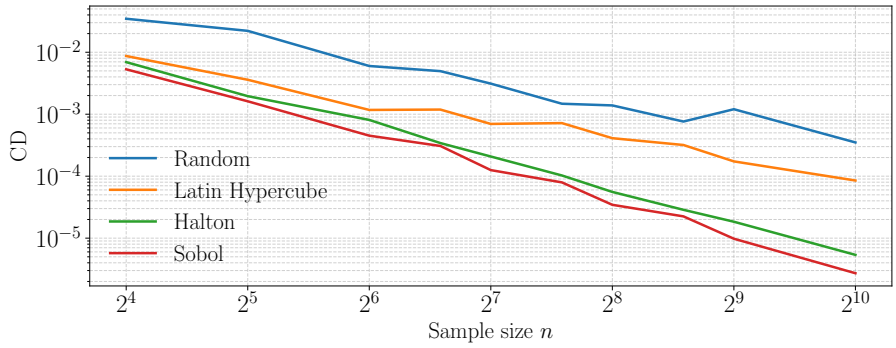

Fig. 4: CD vs. n for different sampling methods.

For obtaining the value of the geometrical parameters, required in the FE simulations, the value of the rolling radius was set to a value of 200 mm. The FE simulations were performed using the explicit solver of *Abaqus* with the assumption of an isothermal process at a temperature of 1100 °C, typical for industrial hot rolling applications. For the considered geometries, a global mesh size of 3 mm and

mass scaling was applied, while keeping slab kinetic energy within 5% of its internal energy. The angular velocity (ω) of the rollers were set to a constant value of 2.7227 rad/s. Also, the rollers were assumed to be analytical rigid surfaces. For modeling the material behavior in the slab, the values of Poisson's ratio (ν) and coefficient of friction (μ) were set to 0.28 and 0.2, respectively. Although the elastic modulus of steel decreases significantly at elevated temperatures, a constant value of $E = 200$ GPa was adopted, as elastic effects are negligible compared to plastic deformation under hot rolling conditions. Also, the Johnson-Cook (JC) model was selected for modeling the hardening behavior, accounting for plastic strain ($\bar{\varepsilon}_{\text{pl}}$), plastic strain-rate ($\dot{\bar{\varepsilon}}_{\text{pl}}$), and temperature (non-dimensionalized as $\hat{T} := [T - T_{\text{ref}}]/[T_{\text{melt}} - T_{\text{ref}}]$):

$$\bar{\sigma}(\bar{\varepsilon}_{\text{pl}}, \dot{\bar{\varepsilon}}_{\text{pl}}, \hat{T}) = [A + B\,\bar{\varepsilon}_{\text{pl}}^{n}][1 + C \ln\left(\dot{\bar{\varepsilon}}_{\text{pl}}/\dot{\varepsilon}_0\right)][1 - \hat{T}^{m}]\,. \tag{10}$$

The required parameter set ($A, B, C, m, n, T_{\text{melt}}, T_{\text{ref}}, \dot{\varepsilon}_0$) was adopted from [28] (corresponding to steel grade 2), as shown in Table 1.

Table 1: Selected JC parameters for steel grade 2 in [28].

A [MPa]	B [MPa]	n [−]	C [−]	m [−]	$\dot{\varepsilon}_0$ [s^{-1}]	T_{ref} [°C]	T_{melt} [°C]
33.901	100.18	0.4951	0.2471	0.6444	0.0037	900	1500

Moreover, since it was necessary to evaluate the spread value for each simulation only under steady-state (SS) condition, the SS detection utility of *Abaqus*, available for the explicit solver, was utilized [29]. For this purpose, among the different possible *norms*, equivalent plastic strain (SSPEEQ) was chosen with the default tolerance of 0.001. Also to form the SS detection control volume, *exit plane* and *cutting plane* were located at the roller center, and at $x = 60$ mm from the roller center, respectively.

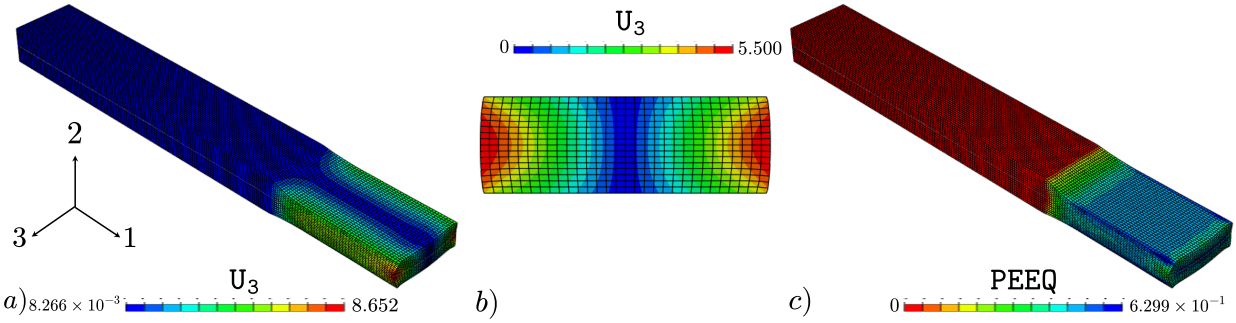

Fig. 5: FE simulation of the rolled slab with depicted results at the last time-step for $a)$ transverse displacement distribution (U3), $b)$ deformed cross-section at SS, and $c)$ distribution of the equivalent plastic strain (PEEQ). In $a)$ and $b)$, absolute values of U3 are represented with units of mm.

Results and Discussion

Traditionally, there have been different closed-form solutions for predicting spread in hot rolling applications either grounded on assumptions to simplify the multi-physics problem or on empirical correlations. Accordingly, they introduce partial physics to the considered problem with reasonable generalization capability. One of such models is the proposed model by Shibahara [30]:

$$\ln\left(\frac{W_{1,\text{Shibahara}}}{W_0}\right) = \ln\left(\frac{h_0}{h_1}\right) \exp\left(-1.64\left(\frac{W_0}{h_0}\right)^{0.376}\left(\frac{W_0}{\sqrt{R\delta}}\right)^{0.016\left[\frac{W_0}{h_0}\right]}\left(\frac{h_0}{R}\right)^{0.015\left[\frac{W_0}{h_0}\right]}\right). \tag{11}$$

Considering the FE solutions obtained for the sampled configurations, Shibahara's model indicates a drastic deviation from the GT with $R^2 \approx -0.159$, MSE $\approx 18.3058\ \text{mm}^2$, as shown in Fig. 6. The pronounced observable discrepancy between Shibahara's model and the GT is the objective to be defined as the output for KANs.

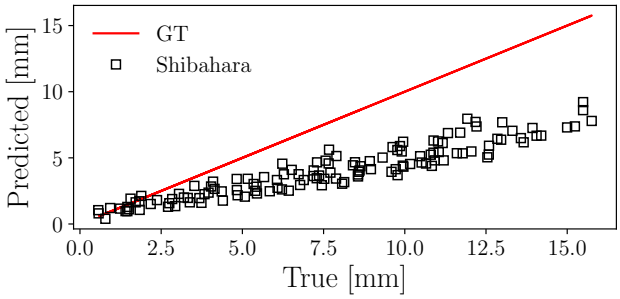

Fig. 6: Prediction vs. true plot for Shibahara model compared to the GT data.

B-spline activation functions of order 3 were used as the bases to expand the network activation functions. For the input-output (I/O) configuration of KANs, three definitions for the input space were tested. In the first setup, analytical spread predictions were used, without any further information. In the second setup, only the input space features $\{x_1, x_2, x_3\}$ were considered. In the last configuration, the combined set of the input feature set and the predictions of the analytical model were introduced to the network. The output for all of these cases were defined as GT spread values. Furthermore, as one of the tunable hyperparameters of the network, we allowed the network width to vary in a range {16, 32, 64, 128}. The network was trained for a total of 80 steps. Moreover, a train-validation-test ratio of 80-10-10 was selected as splitting criterion for the considered 128 samples. In Fig. 7, the obtained learning curves for training and validation losses are depicted for different network widths and input definitions.

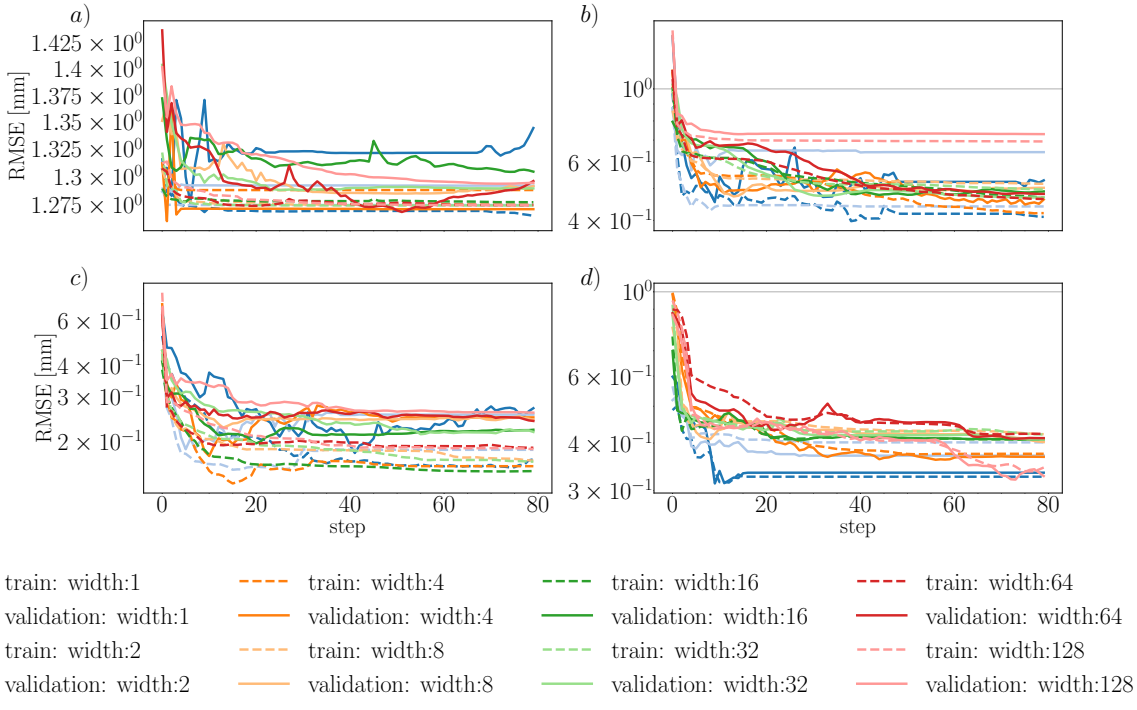

Fig. 7: Learning curves for the networks when the network input (I) is considered to be $a)\ I = \{\Delta W_{\text{Shibahara}}\}$, $b)\ I = \{x_1, x_2, x_3\}$ with $\lambda = 0.1$, $c)\ I = \{x_1, x_2, x_3, \Delta W_{\text{Shibahara}}\}$ with $\lambda = 0.01$, and $d)\ I = \{x_1, x_2, x_3, \Delta W_{\text{Shibahara}}\}$ with $\lambda = 0.1$.

As can be seen in Fig. 7 a), the input space definition $I = \{\Delta W_{\text{Shibahara}}\}$ where $W_{\text{Shibahara}}$ is appropriately nondimensionalized, yields higher values of training and validation loss errors compared to the cases in $b) - d$). Moreover, although the expanded definitions in b) and c) results in lower root mean square error (RMSE) values, overfitting remains a problem in these cases. Choosing the expanded input space definition, including the input space features and analytical model predictions, together with a smaller regularization parameter of 0.01, results in models with less overfitting, while retaining the model accuracy (Fig. 7d)). The number of grid extensions is one of the trainable hyperparameters referring to fine-graining the spline grids in KANs. For the network with $I = \{x_1, x_2, x_3, \Delta W_{\text{Shibahara}}\}$ and $\lambda = 0.1$, fine-graining was performed for the considered width spectrum with a grid range of $g \in \{2, 3, 4, 5, 8, 10, 15, 20\}$. As it can be seen in Fig. 8, the stair-case loss reduction can be seen up to a grid-size of 3.

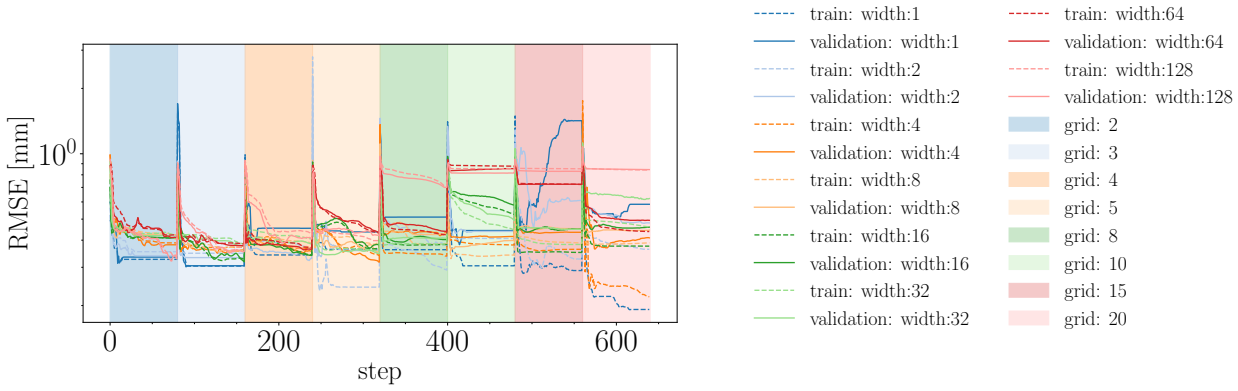

Fig. 8: Learning curves for network grid and width updates.

As an ultimate metric to evaluate the selected KAN network, the selected KAN network with 4 inputs and overall regularization parameter of 0.1 and a grid size of 2, the network was applied to the test set, for which the results are shown in Fig. 9. In Fig. 9 a), the learning curves between the training and validation loss are not suggestive of overfitting in the network and in part b) the network shows an acceptable performance on the test set with R^2 and RMSE metrics equal to 0.99 and 0.38 mm, respectively. Also, the pruned network is shown in Fig. 9 d), from which a closed-form expression can be obtained in the symbolification step. The resulting nondimensional expression for ΔW_{FE} reads:

$$\begin{aligned}\Delta W_{\text{FE}}(x_1, x_2, x_3, \Delta W_{\text{Shibahara}}) = &-0.164\, x_1^2 + 0.108\, x_1 - 17.378\, x_2 + 0.026\, x_3 \\ &- 0.132\, \Delta W_{\text{Shibahara}}^2 + 3.255\, \Delta W_{\text{Shibahara}} + 1.530. \end{aligned} \tag{12}$$

The complexity of the symbolified expression is dependent on the second regularization parameter, corresponding to μ_2 in Eq. 8. We set this parameter to 2 and in Fig. 9c), the performance of the closed-form of the discrepancy model is evaluated against the test set, which is closely aligned with the network, prior to symbolification.

Conclusions

In this work, we investigated the applicability of KANs for obtaining interpretable models that capture spread in hot rolling processes of steel slabs. Initially, the process-specific geometrical parameters were used to define the sampling input space in terms of non-dimensional parameters. To better represent the problem non-linearity, different space-filling sampling methods, including random, Halton, LHS, and Sobol, were compared in terms of their CD value in lower data regimes. A total of 128 sampled points were considered, and they were used to define the configurations subjected to FE simulation. For increasing computational efficiency, the spread values at SS were captured for each simulation and the results were post-processed to form the GT for the network, with a train-validation-test ratio of 80-10-10. The obtained spread results were compared to Shibahara's closed-form solution,

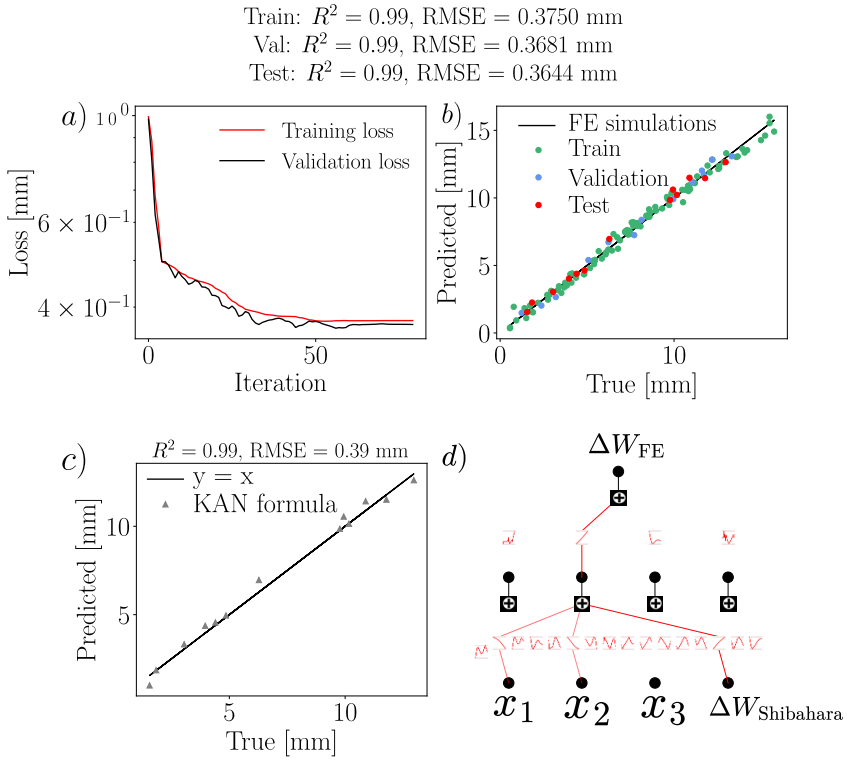

Fig. 9: Performance of a trained KAN network for capturing the width with a) learning curve, b) prediction vs. true plot before pruning and symbolification, c) prediction vs. true plot for the test set, after pruning and symbolification (Eq. 12), and d) network pruning, considering different input contributions.

which was the subject of network correction. As the output of the network, GT spread values were selected and B-spline functions of order 3 were chosen as the trainable activation functions. Also, with the focus on grid-extension and network width as the tunable hyperparameters, it was observed that expanded input space definition with the inclusion of the non-dimensional process parameters and the analytical model predictions resulted in a reduced over-fitting on the validation set. Moreover, the same effect was observed via reducing the overall regularization coefficient, which was set equal to 0.1. With the tuned grid-size and network width of 2 and 4, the network also showed an optimal performance on the test set with R^2 and RMSE measures of 0.99 and 0.3750 mm, respectively. Lastly, the output of the network was symbolified and resulted in a closed-form expression for the existing model discrepancy, with a similar prediction performance on the test set to the KAN.

Acknowledgment

This research was carried out under project number T22008 in the framework of the Research Program of the Materials innovation institute (M2i) (www.m2i.nl) supported by the Dutch government. The authors also gratefully acknowledge Tata Steel for valuable discussions.

References

[1] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[2] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995. PMLR, 2017.

[3] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. In *International Conference on Neural Information Processing*, pages 264–274. Springer, 2019.

[4] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pages 5301–5310. PMLR, 2019.

[5] Wei Cai and Zhi-Qin John Xu. Multi-scale deep neural networks for solving high dimensional pdes. *ArXiv*, abs/1910.11710, 2019.

[6] George Philipp, Dawn Song, and Jaime G Carbonell. The exploding gradient problem demystified-definition, prevalence, impact, origin, tradeoffs, and solutions. *arXiv preprint arXiv:1712.05577*, 2017.

[7] Miles Cranmer. Interpretable machine learning for science with pysr and symbolicregression.jl. *ArXiv*, abs/2305.01582, 2023.

[8] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Y. Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *ArXiv*, abs/2404.19756, 2024.

[9] A. K. Kolmogorov. On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition. *Doklady Akademii Nauk SSSR*, 114:953–956, 1957.

[10] Amir Noorizadegan, Sifan Wang, and Leevan Ling. A practitioner's guide to kolmogorov-arnold networks. *ArXiv*, abs/2510.25781, 2025.

[11] Xiong Xiong, Kang Lu, Zhuo Zhang, Zheng Zeng, Sheng Zhou, Zichen Deng, and Rongchun Hu. J-pikan: A physics-informed kan network based on jacobi orthogonal polynomials for solving fluid dynamics. *Communications in Nonlinear Science and Numerical Simulation*, 152:109414, 2026.

[12] Juan Diego Toscano, Theo Käufer, Zhibo Wang, Martin Maxey, Christian Cierpka, and George Em Karniadakis. Aivt: Inference of turbulent thermal convection from measured 3d velocity data by physics-informed kolmogorov-arnold networks. *Science Advances*, 11, 2025.

[13] SS Sidharth and R Gokul. Chebyshev polynomial-based kolmogorov-arnold networks: An efficient architecture for nonlinear function approximation. *ArXiv*, abs/2405.07200, 2024.

[14] Qi Qiu, Tao Zhu, Helin Gong, Liming Luke Chen, and Huansheng Ning. Relu-kan: New kolmogorov-arnold networks that only need matrix addition, dot multiplication, and relu. *ArXiv*, abs/2406.02075, 2024.

[15] Ali Kashefi. Kolmogorov–arnold pointnet: Deep learning for prediction of fluid fields on irregular geometries. *Computer Methods in Applied Mechanics and Engineering*, 439:117888, 2025.

[16] Ziyao Li. Kolmogorov-arnold networks are radial basis function networks. *arXiv preprint arXiv:2405.06721*, 2024.

[17] Jinfeng Xu, Zheyu Chen, Jinze Li, Shuo Yang, Wei Wang, Xiping Hu, and Edith C-H Ngai. Fourierkan-gcf: Fourier kolmogorov-arnold network–an effective and efficient feature transformation for graph collaborative filtering. *arXiv preprint arXiv:2406.01034*, 2024.

[18] Danli Li, Bo Yan, Quan Long, and Bin Wang. De-kan: A differential evolution-based optimization framework for enhancing kolmogorov-arnold networks in complex nonlinear modeling. *2025 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2025.

[19] Lin Zhang, Lei Chen, Fuxiang An, Zixuan Peng, Yuhang Yang, Tingting Peng, Yongshi Song, and Yanzheng Zhao. A physics-informed neural network for nonlinear deflection prediction of ionic polymer-metal composite based on kolmogorov-arnold networks. *Eng. Appl. Artif. Intell.*, 144:110126, 2025.

[20] Prakash Thakolkaran, Yaqi Guo, Shivam Saini, Mathias Peirlinck, Benjamin Alheit, and Siddhant Kumar. Can kan cans? input-convex kolmogorov-arnold networks (kans) as hyperelastic constitutive artificial neural networks (cans). *Computer Methods in Applied Mechanics and Engineering*, 443:118089, 2025.

[21] Amanda A. Howard, Bruno Jacob, Sarah H. Murphy, Alexander Heinlein, and Panos Stinis. Finite basis kolmogorov-arnold networks: domain decomposition for data-driven and physics-informed problems. *ArXiv*, abs/2406.19662, 2024.

[22] Mehrdad Kiamari, Mohammad Kiamari, and Bhaskar Krishnamachari. Gkan: Graph kolmogorov-arnold networks. *ArXiv*, abs/2406.06470, 2024.

[23] Y. Xin, Z. Zhang, Z. Zhong, and Y. Li. Lateral spread prediction based on hybrid CNN–LSTM model for hot strip finishing mill. *Materials Letters*, 378:137594, 2025.

[24] Yanjiu Zhong, Jingcheng Wang, Jiahui Xu, Jun Rao, and Kangbo Dang. Data-driven width spread prediction model improvement and parameters optimization in hot strip rolling process. *Applied Intelligence*, 53:25752–25770, 2023.

[25] Tomaso Poggio. How deep sparse networks avoid the curse of dimensionality: Efficiently computable functions are compositionally sparse. *CBMM Memos*, (138), October 2022.

[26] E. Buckingham. On physically similar systems; illustrations of the use of dimensional equations. *Physical Review*, 4(4):345–376, 1914.

[27] Kai-Tai Fang, Runze Li, and Agus Sudjianto. *Design and Modeling for Computer Experiments*, volume 6 of *Computer Science and Data Analysis Series*. Chapman & Hall/CRC, Boca Raton, FL, 2005.

[28] Mario F. Buchely, Shouvik Ganguly, David C. Van Aken, Ronald O'Malley, Simon Lekakh, and K. Chandrashekhara. Experimental development of johnson–cook strength model for different carbon steel grades and application for single-pass hot rolling. *steel research international*, 91(7):1900670, 2020.

[29] ABAQUS, Inc. Abaqus analysis user's manual, version 6.6. https://classes.engineering.wustl.edu/2009/spring/mase5513/abaqus/docs/v6.6/books/usb/default.htm?startat=pt04ch11s08aus64.html.

[30] Takashi Shibahara, Yoshisuke Misaka, Teruo Kono, Mitsuru Koriki, and Hiroshi Takemoto. Edger set-up model at roughing train in hot strip mill. *Tetsu To Hagane-journal of The Iron and Steel Institute of Japan*, 67:2509–2515, 1981.