

Research Article

Open Access



# Efficient prediction of potential energy surface and physical properties with Kolmogorov-Arnold Networks

Rui Wang<sup>1,2</sup>, Hongyu Yu<sup>1,2</sup>, Yang Zhong<sup>1,2</sup>, Hongjun Xiang<sup>1,2,\*</sup>

<sup>1</sup>Key Laboratory of Computational Physical Sciences (Ministry of Education), Institute of Computational Physical Sciences, State Key Laboratory of Surface Physics, and Department of Physics, Fudan University Shanghai, Shanghai 200433, China.

<sup>2</sup>Shanghai Qi Zhi Institute, Shanghai 200030, China.

\*Correspondence to: Prof. Hongjun Xiang, Key Laboratory of Computational Physical Sciences (Ministry of Education), Institute of Computational Physical Sciences, State Key Laboratory of Surface Physics, and Department of Physics, Fudan University, 220 Handan Rd., Shanghai 200433, China. E-mail: hxjiang@fudan.edu.cn

**How to cite this article:** Wang R, Yu H, Zhong Y, Xiang H. Efficient prediction of potential energy surface and physical properties with Kolmogorov-Arnold Networks. *J Mater Inf* 2024;4:32. <https://dx.doi.org/10.20517/jmi.2024.46>

**Received:** 9 Sep 2024 **First Decision:** 6 Nov 2024 **Revised:** 28 Nov 2024 **Accepted:** 5 Dec 2024 **Published:** 28 Dec 2024

**Academic Editor:** Yong Du **Copy Editor:** Ping Zhang **Production Editor:** Ping Zhang

## Abstract

The application of machine learning methods for predicting potential energy surface and physical properties within materials science has garnered significant attention. Among recent advancements, Kolmogorov-Arnold Networks (KANs) have emerged as a promising alternative to traditional Multi-Layer Perceptrons. This study evaluates the impact of substituting Multi-Layer Perceptrons with KANs within four established machine learning frameworks: Allegro, Neural Equivariant Interatomic Potentials, Higher Order Equivariant Message Passing Neural Network (MACE), and the Edge-Based Tensor Prediction Graph Neural Network. Our results demonstrate that the integration of KANs enhances prediction accuracies, especially for complex datasets such as the HfO<sub>2</sub> structures. Notably, using KANs exclusively in the output block achieves the most significant improvements, improving prediction accuracy and computational efficiency. Furthermore, employing KANs exclusively in the output block facilitates faster inference and improved computational efficiency relative to utilizing KANs throughout the entire model. The selection of optimal basis functions for KANs depends on the specific problem. Our results demonstrate the strong potential of KANs in enhancing machine learning potentials and material property predictions. Additionally, the proposed methodology offers a generalizable framework that can be applied to other ML architectures.

**Keywords:** Machine learning, property prediction, Kolmogorov-Arnold Networks



© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



## INTRODUCTION

The application of machine learning (ML) methods has become increasingly significant in materials science, offering significant advantages over traditional approaches<sup>[1-7]</sup>. By leveraging large datasets and complex algorithms, ML methods can uncover complex patterns and relationships<sup>[8,9]</sup>.

ML potentials and physical property predictions are two key applications of ML in material science. ML potentials, such as Allegro<sup>[10]</sup>, NequIP<sup>[11]</sup> and Equivariant transformers<sup>[12]</sup>, utilize ML methods to predict the potential energy surfaces of atomic interactions within material systems<sup>[13,14]</sup>. Consequently, ML potentials facilitate more efficient and precise molecular dynamics simulations over extended time scales<sup>[15-18]</sup>, significantly reducing computational costs while maintaining high accuracy. Their applications extend across diverse fields, including magnetic systems<sup>[15,19]</sup>, metal-organic frameworks<sup>[16]</sup>, and many-body systems<sup>[20]</sup>, thus advancing innovation in materials research and design. Furthermore, ML techniques offer broad applicability in predicting physical properties of materials, including tensor properties<sup>[21]</sup>, Hamiltonians<sup>[22-24]</sup>, electron-phonon coupling strengths<sup>[25]</sup>, and other properties<sup>[26-28]</sup> of solids and molecules. Employing these methods to predict physical properties of materials facilitates high-throughput searches and the design of novel materials with tailored properties for specific applications, such as superconductors<sup>[29]</sup>, high-piezoelectric materials<sup>[30]</sup>, porous Materials<sup>[31]</sup>, and direct-gap silicon materials<sup>[32]</sup>. The integration of ML in property prediction significantly accelerates the discovery and design of new materials and also enhances our understanding of existing ones. However, existing ML potentials and property prediction models often face limitations in accuracy or require extensive training times<sup>[13,33]</sup>, especially when dealing with complex systems, making it challenging to achieve precise and timely results. Addressing these issues requires the development of new models capable of improving prediction accuracy while minimizing training time.

Multi-layer perceptrons (MLPs)<sup>[34,35]</sup> are the foundational blocks of most modern ML models. Recently, Liu *et al.* proposed Kolmogorov-Arnold Networks (KANs)<sup>[36]</sup> as an alternative to MLPs. KANs are inspired by the Kolmogorov-Arnold representation theorem<sup>[37,38]</sup>, which states that any continuous function can be represented as a finite composition of continuous functions of one variable and addition. Both MLPs and KANs have fully connected structures<sup>[36]</sup>. In MLPs, the nodes are connected by linear weight parameters, and activation functions are placed on nodes to introduce non-linearity. In contrast, in KANs, the linear weight parameters are replaced by learnable univariate functions parameterized as B-splines, and only summations are performed on nodes. By utilizing the Kolmogorov-Arnold representation, KANs demonstrate the capability to approximate complex functions with high accuracy, and may outperform MLPs in both prediction accuracy and interpretability<sup>[36]</sup>.

The univariate functions in KANs can be adapted using various basis functions to better address specific problems. Since the introduction of KANs, numerous variations have been developed by replacing B-splines with different basis functions. The operations of calculating the B-spline basis and rescaling the grids can lead to severe efficiency bottlenecks in KANs<sup>[39]</sup>. Li *et al.* proposed FastKAN<sup>[39]</sup>, which utilizes Gaussian radial basis functions (RBFs) with Gaussian kernels instead of B-splines, offering a significantly faster implementation of KANs without sacrificing accuracy. Bozorgasl *et al.* introduced Wavelet KANs<sup>[40]</sup> by incorporating wavelet functions, enabling the network to capture both high- and low-frequency components of the input data efficiently. Other variations include Fourier KAN for Graph Collaborative Filtering<sup>[41]</sup>, Fractional KAN<sup>[42]</sup> incorporating fractional-orthogonal Jacobi functions, and KANs incorporating sinusoidal basis functions<sup>[43]</sup>. Additionally, KANs can be integrated into existing ML frameworks and workflows with minimal modifications<sup>[36]</sup>. This compatibility ensures that current ML

methods can leverage the advantages of KANs. Nagai *et al.* incorporated KANs into three ML potentials, and used KANs to redefine the descriptors of artificial neural network (ANN) potentials<sup>[44]</sup>. Other applications include Temporal KANs<sup>[45]</sup> for multi-step time series forecasting, Graph KANs<sup>[46]</sup> for graph-structured data, and Signature-Weighted KANs<sup>[47]</sup> using learnable path signatures, among others<sup>[48,49]</sup>.

KANs are particularly advantageous in scenarios where traditional neural networks face challenges, such as in high-dimensional spaces or when dealing with highly nonlinear functions<sup>[36]</sup>. Many ML potentials and property prediction models rely heavily on MLPs, which makes such models ideal candidates for integrating KANs to enhance prediction accuracy. Replacing MLPs with KANs allows these models to leverage the efficiency and accuracy of KANs without requiring the development of entirely new architectures, thereby saving time and resources in model development and training. Despite the potential benefits, there has been limited systematic testing in this area. In this study, we investigated the impact of replacing MLPs with KANs on various ML models in property prediction. Specifically, we substituted MLPs in different parts of the ML potential Allegro<sup>[10]</sup> with KANs employing various basis functions. Our results show that replacing the MLPs in the output block of the Allegro model not only enhances prediction accuracy but can also reduce training time in certain cases. Additionally, it improves inference speed and computation resource efficiency relative to using KANs without MLPs. We extended this approach to other models, including Neural Equivariant Interatomic Potentials (NequIP)<sup>[11]</sup>, Higher Order Equivariant Message Passing Neural Network (MACE)<sup>[50]</sup> and the edge-based tensor prediction graph neural network (ETGNN)<sup>[24]</sup>. Consistently, replacing the MLPs in the output blocks of these models improved prediction accuracy and decreased training time. Overall, using KANs with different basis functions generally enhances prediction accuracy, and the optimal basis function depends on the specific problem. Our findings highlight the significant promise of KANs in enhancing ML models for material property prediction and ML potentials.

## MATERIALS AND METHODS

In this study, we examined the effect of replacing MLPs with KANs in various ML models for property prediction. [Figure 1A](#) illustrates the differences between MLPs and KANs. In KANs, the linear weight parameters are substituted with learnable univariate functions<sup>[36]</sup>, which enhance accuracy and interpretability. [Figure 1B](#) depicts the general framework of a property prediction model. In this work, we replaced MLPs with KANs in different parts of the models. We utilized KANs with three types of basis functions: B-spline, Gaussian, and Fourier functions. [Table 1](#) summarizes the configurations utilized in this study and is included in the [Supplementary Materials](#).

### Machine learning potential Allegro using KAN

First, we utilized Allegro<sup>[10]</sup> to assess the impact of replacing MLPs in various parts of ML potentials with KAN networks employing different basis functions. Allegro<sup>[10]</sup> is an equivariant deep-learning interatomic potential. By integrating equivariant message-passing neural networks (MPNN)<sup>[51]</sup> with strict locality, Allegro achieves high prediction accuracy, generalizes well to out-of-distribution data, and scales effectively to large system sizes.

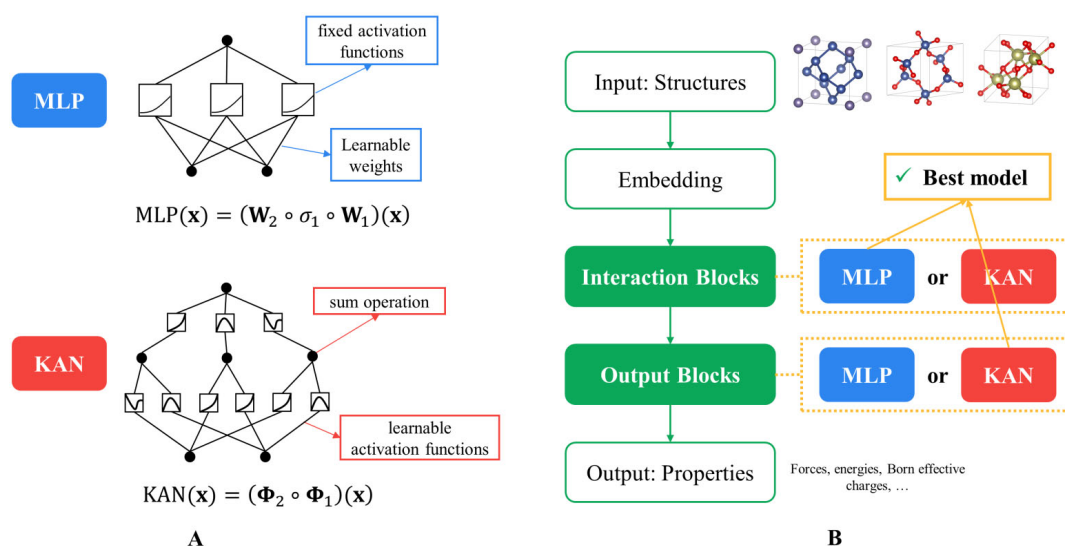
#### *Replacing all MLPs with KANs with different basis functions*

First, we tried replacing all MLPs in the Allegro model with KANs using different basis functions. We substituted MLPs in three parts of the Allegro model with KANs: the two-body latent embedding part, the latent MLP part, and the output block, as the second model shown in [Figure 2](#). The function of the two-body latent embedding part is to embed the initial scalar features into the latent features of atom pairs. The latent MLP passes information from the tensor products of the current features to the scalar latent space. The output block predicts pairwise energies using the output from the final layer. We did not replace the

**Table 1. Summary table of the configurations utilized in the study**

Original model	Notes	Besides the output block	Output block	Basis functions of KANs
Allegro	To identify the optimal basis functions	MLP	MLP	
		KAN	KAN	B-splines
		KAN	KAN	Gaussian functions
	To identify the optimal configuration	KAN	KAN	Fourier functions
		MLP	KAN	Gaussian functions
NequIP		MLP	MLP	
		MLP	KAN	Gaussian functions
		MLP	KAN	B-splines
MACE	Each model used three different random seeds	MLP	MLP	
		MLP	KAN	B-spline functions
ETGNN		MLP	MLP	
		MLP	KAN	Gaussian functions
		MLP	KAN	B-splines

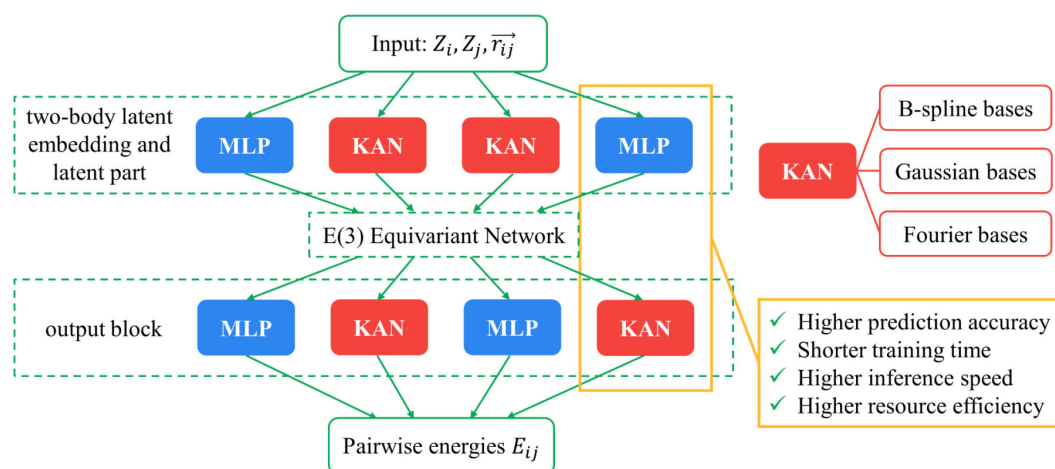
MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; NequIP: Neural equivariant interatomic potentials; ETGNN: Edge-based tensor prediction graph neural network; MACE: Higher order equivariant message passing neural network.



**Figure 1.** Efficient prediction of potential energy surface and physical properties with KAN. (A) Comparison of MLP and KAN<sup>[36]</sup>. MLPs utilize learnable weights on the edges and fixed activation functions on nodes. In contrast, KANs employ learnable activation functions parameterized as various basis functions on edges with sum operations on nodes; (B) Replacing MLPs in ML potentials and property prediction models with KANs. The left side illustrates the general framework of ML potentials and property prediction models. In this study, MLPs in different parts of the ML potentials and property prediction models are replaced with KANs employing various basis functions. Our results demonstrate that replacing MLPs with KANs in the output blocks leads to higher prediction accuracy and reduced training times compared to using MLPs, and higher inference speed and computation resource efficiency compared to using KANs without MLPs. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; ML: Machine learning.

MLPs in the environment embedding part, as it typically consists of a simple one-layer linear projection, making it trivial to substitute with KANs.

For the Allegro model utilizing KANs, we tested KANs with the original B-spline basis functions, Gaussian functions, and Fourier functions. B-spline basis functions were chosen due to their use in the original KAN



**Figure 2.** Replacing MLPs in different parts of the ML potential Allegro<sup>[10]</sup> with KANs employing various basis functions.  $Z_i$  stands for the chemical species of atom  $i$ .  $\vec{r}_{ij}$  stands for the relative displacement vector from atom  $i$  to atom  $j$ . Substituting MLPs with KANs generally enhances prediction accuracy. Specifically, replacing MLPs in the output block of the Allegro model results in higher prediction accuracy and shorter training time than using MLP, and higher inference speed and higher computation resource efficiency than using KANs throughout the entire model. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; ML: Machine learning.

study<sup>[36]</sup>, and their ability to provide smooth and compact representations. For KANs with B-spline basis functions, we employed the efficient-kan package<sup>[52]</sup>, a re-implementation of the original KAN with enhanced efficiency. Gaussian and Fourier functions were chosen due to their well-known properties of universal approximation and their compatibility with the existing implementation fastkan<sup>[39]</sup>. For KAN implementations with Gaussian and Fourier functions, we used the fastkan package<sup>[39]</sup>. The details of different models are included in [Supplementary Materials](#).

We evaluated the accuracy and efficiency of various models using the Ag dataset<sup>[10]</sup>. This dataset was derived from ab-initio molecular dynamics simulations of a bulk face-centered-cubic structure with a vacancy, consisting of 71 atoms. The simulations were performed using the Vienna Ab-Initio Simulation Package (VASP)<sup>[53]</sup> with the Perdew-Burke-Ernzerhof (PBE) exchange-correlation functional<sup>[54]</sup>. The dataset includes 1,000 distinct structures, with 950 used for training and 50 for validation.

#### Replacing some of MLP in Allegro with KAN

We subsequently replaced MLPs in various parts of the Allegro model with KANs to identify the optimal configuration. We selected KANs with Gaussian basis functions based on the results in section 3.1.1, which provides higher prediction accuracy while maintaining relatively short training times. Specifically, we evaluated two configurations: incorporating KANs in the two-body latent embedding and latent MLP parts and incorporating KANs solely in the output block, as shown in [Figure 2](#). The details of different models are included in [Supplementary Materials](#).

We initially evaluated the performance of various models using the Ag dataset<sup>[10]</sup>. The Ag dataset is identical to the one described in the previous section. We also evaluated the inference speeds and GPU memory usage of various models by performing molecular dynamics simulations using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS)<sup>[55]</sup>. The simulations employed the Allegro pair style implemented in the Allegro interface<sup>[42]</sup>. The initial structure was obtained from the Ag dataset<sup>[10]</sup>. The simulations were conducted under a canonical (NVT) ensemble at a temperature of 300K with a time step of 1 ps. For each model, we ran 5,000 time steps to measure the inference speed.

In order to assess the impact of dataset complexity on the relative performance of KANs and MLPs, we proceeded to evaluate these models on the more complex HfO<sub>2</sub> structures. HfO<sub>2</sub> structures exhibit complex interatomic interactions, including mixed ionic-covalent character due to *p-d* hybridization, making it challenging to develop accurate ML potentials<sup>[56]</sup>. The HfO<sub>2</sub> dataset<sup>[56]</sup> was generated using density functional theory calculations performed with the VASP package<sup>[53]</sup>. The structures were initially generated by perturbing ground-state HfO<sub>2</sub> structures, followed by sampling through NPT simulations at various temperatures. We selected 10,000 structures from the dataset, with 9,000 used for training and 1,000 for validation.

### Machine learning potential NequIP using KAN

We also investigated replacing MLPs with KANs in the NequIP model<sup>[11]</sup>, a deep-learning interatomic potential. NequIP utilizes E(3)-equivariant convolutions to capture interactions between geometric tensors, resulting in exceptional prediction accuracy and remarkable data efficiency.

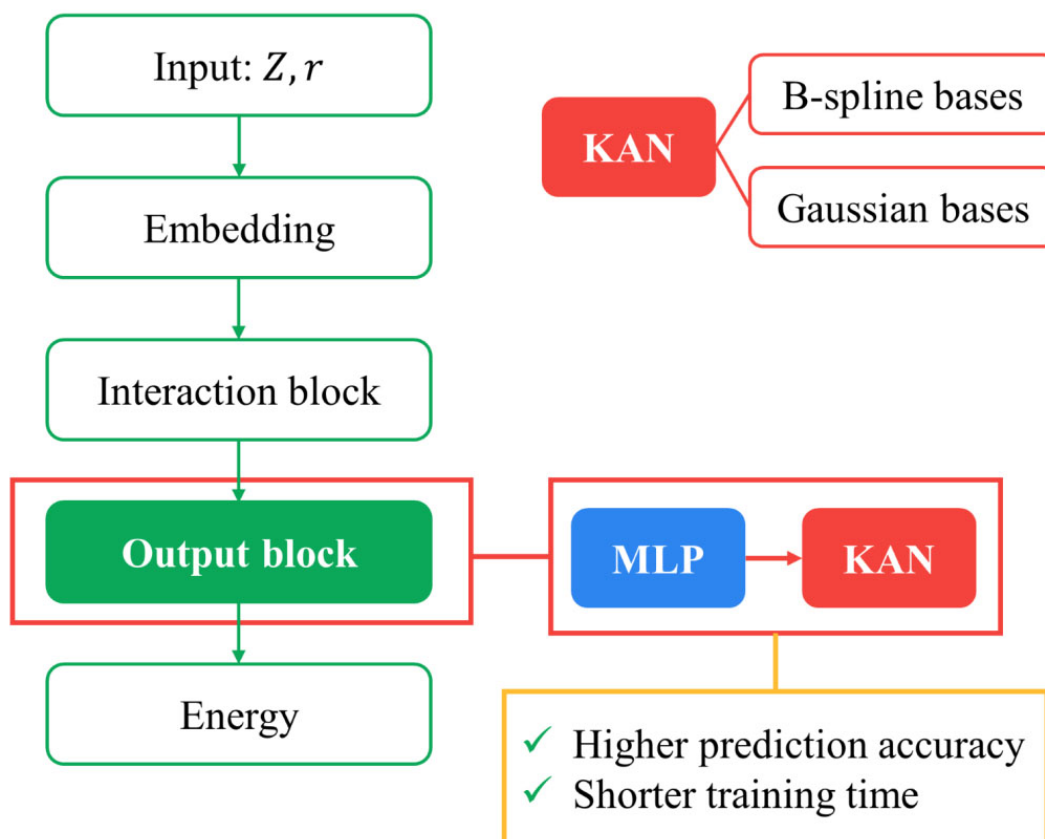
The NequIP architecture is based on an atomic embedding that generates initial features from atomic numbers. This embedding is followed by interaction blocks that integrate interactions between neighboring atoms through self-interactions, convolutions, and concatenations. The final output block converts the output features of the last convolution into atomic potential energy. As with the optimal model in section 3.1.2, we only replaced the MLPs in the output blocks with KANs, as shown in [Figure 3](#). We tested KANs with Gaussian and B-spline bases, utilizing the efficient-kan package<sup>[52]</sup> for the B-spline bases and the fastkan package<sup>[39]</sup> for the Gaussian bases. The details of different models are included in [Supplementary Materials](#). We tested NequIP with MLPs and KANs on the Ag dataset identical to the one used in previous sections.

### Machine learning potential MACE using KAN

We also investigated replacing MLPs with KANs in the MACE model<sup>[50]</sup>, a MPNN<sup>[51]</sup> model designed for creating fast and accurate force fields. Unlike other MPNN models, MACE utilizes higher-body messages instead of two-body messages, significantly reducing the number of required message-passing iterations. This design makes MACE both computationally efficient and highly parallelizable while achieving state-of-the-art accuracy<sup>[50]</sup>.

The MACE architecture is based on the framework of MPNN<sup>[51]</sup>. A forward pass of the network consists of multiple message construction, update, and readout steps<sup>[50]</sup>. In the message construction, features are generated by embedding the edges and previous node features and pooling over neighbors. Then, higher-order features are constructed through tensor products and summarization. The update step applies a linear transformation to the message combined with a residual connection. In the readout step, the invariant components of the node features are mapped to the site energy contribution using linear combinations for all layers except the last layer. In the last layer, these node features are mapped to the site energy contribution using MLPs. The total site energy is represented as the sum of these contributions. Consistent with the optimal configuration described in Section 3.1.2, we replaced the MLPs in the final layer's readout section, analogous to the output blocks in the Allegro model, with KANs. Additionally, we introduced a KAN layer besides the linear combinations in other layers as illustrated in [Figure 4](#). We tested KANs with the original B-spline bases, utilizing the pykan package<sup>[36]</sup> for the Gaussian bases. To evaluate the robustness of KAN-based models under varying input parameters, we tested each model using three different random seeds (1111, 2222, and 3333). The details of different models are included in [Supplementary Materials](#).





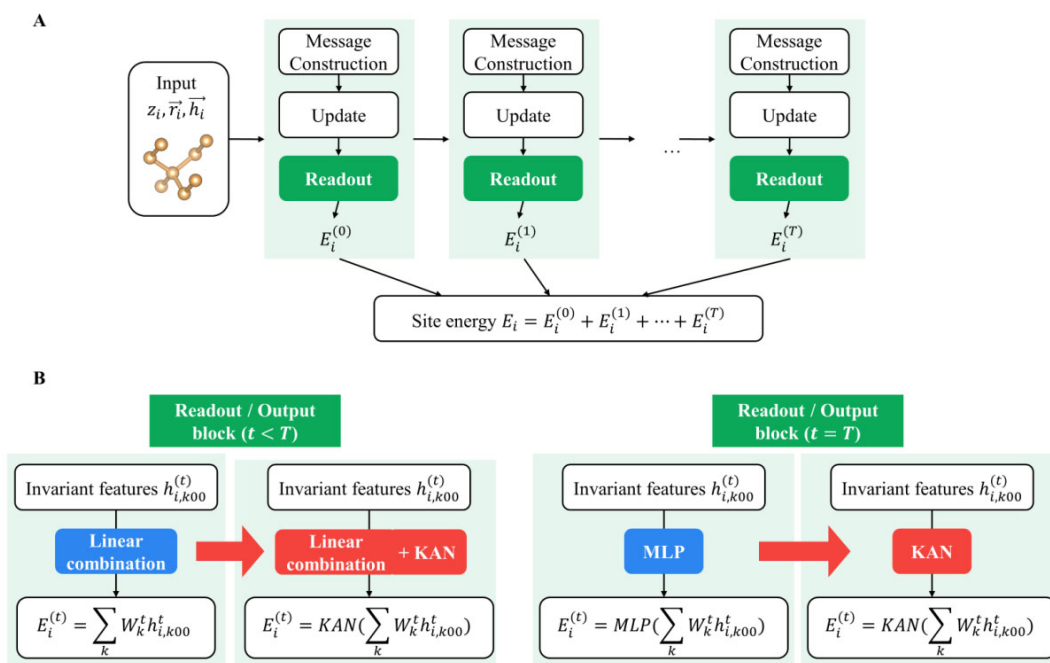
**Figure 3.** Replacing MLPs in the output block of the NeqIP<sup>[11]</sup> model with KANs employing B-spline and Gaussian basis functions.  $e$  and  $\alpha$  stand for the lengths of the edges and the angles between the edges in the cluster. Substituting the MLP with the B-spline bases KAN improves prediction accuracy and significantly shortens the training time. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; NeqIP: Neural equivariant interatomic potentials.

We tested MACE with MLPs and KANs on the carbon dataset<sup>[57,58]</sup>, which includes 4,080 structures in the training set and 450 in the test set. This dataset comprises structural snapshots obtained from *ab initio* molecular dynamics and simulations employing Gaussian approximation potentials<sup>[57]</sup>. It contains a diverse range of carbon structures, including amorphous surfaces, bulk crystals, and liquid and amorphous carbon. The dataset was selected for its structural complexity, particularly the amorphous materials, which lack regular repeating patterns and present challenges in accurately modeling atomic interactions<sup>[57]</sup>. The carbon dataset was chosen to assess the robustness of KAN-based models across various material types.

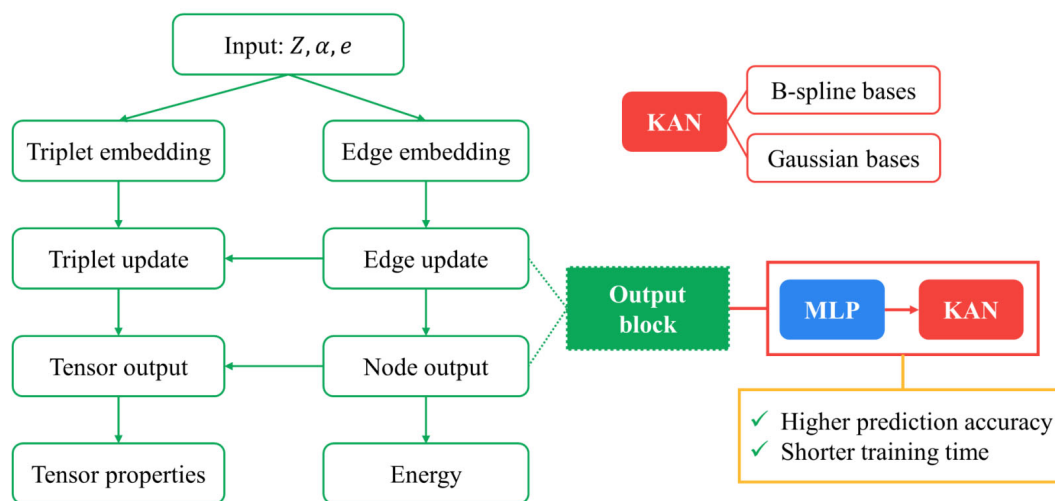
### Tensor prediction networks using KAN

In this study, we utilized the tensor prediction networks (ETGNN)<sup>[21]</sup> to predict the tensorial properties of crystals. In ETGNN, tensorial properties are represented by averaging the contributions of atomic tensors within the crystal. The tensor contribution of each atom is decomposed into a linear combination of local spatial components, which are projected onto the edge directions of clusters with varying sizes. This approach enables ETGNN to predict the tensorial properties of crystals with efficiency and accuracy while maintaining equivariance.

In the ETGNN architecture, the initial features are generated in the embedding block and subsequently updated through a series of update blocks. The output of the final update block is then aggregated into node features by the node output block to produce scalar outputs. As represented in Figure 5, consistent with our



**Figure 4.** Replacing MLPs in the output block of the MACE model<sup>[50]</sup> with KANs employing B-spline basis functions. (A) The general framework of MACE model.  $z_i$  stands for the chemical species of atom  $i$ .  $\vec{r}_i$  indicates the atomic positions of atom  $i$ .  $\vec{h}_i$  represents the learnable features of atom  $i$ . (B) Replacing the linear combinations and MLPs in the output block with KANs. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; MACE: Higher order equivariant message passing neural network.



**Figure 5.** Replacing MLPs in the output block of the ETGNN model<sup>[21]</sup> with KANs employing B-spline and Gaussian basis functions.  $e$  and  $\alpha$  stand for the lengths of the edges and the angles between the edges in the cluster. Replacing the MLP in the output block with a KAN using Gaussian basis functions significantly improves prediction accuracy while also reducing training time. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; ETGNN: Tensor prediction networks.

modifications to the ML potentials, we only replaced the projection part of the MLPs from the edge update block and the node output block, which, similar to the output block in ML potential models, convert the output features into scalars. We replaced the MLPs with KANs using Gaussian and B-spline bases. The details of different models are included in [Supplementary Materials](#).



We compared the accuracy of ETGNN using MLPs and KANs with different basis functions on a SiO<sub>2</sub> dataset<sup>[21]</sup>. The dataset consists of 3,992 randomly perturbed SiO<sub>2</sub> structures calculated using density functional perturbation theory (DFPT). The dataset was split into training, validation, and test sets in a 6:2:2 ratio. We calculated the Born effective charges using ETGNN with MLPs and KANs with Gaussian and B-spline basis functions.

## RESULTS AND DISCUSSIONS

### Machine learning potential Allegro using KAN

#### *Replacing all MLPs with KANs with different basis functions*

First, we tried replacing all MLPs in the Allegro model with KANs using different basis functions. The mean absolute error (MAE) and training times of the predicted potentials are presented in [Table 2](#) and [Figure 6](#). Notably, all three Allegro models using KANs demonstrated lower force and energy MAE than the original Allegro model with MLPs. Specifically, the force MAE for the KAN-based model with Gaussian bases is 0.014 eV/Å, which is 12.5% lower than that of the MLP-based Allegro model. The model utilizing KANs with B-spline bases achieved the lowest validation energy MAE of 0.029 eV/atom, which is 17.1% lower than the MLP-based model. However, this model required nearly five times the training time. Conversely, the Allegro model with KANs using Gaussian bases also exhibited a lower validation energy MAE than the MLP-based model, 0.032 eV/atom, while maintaining a comparable training time. The model with Fourier bases resulted in a validation energy MAE similar to the MLP-based Allegro model but required a longer training time.

All three Allegro models using KANs demonstrated superior prediction accuracy compared to Allegro using MLPs. This improved performance may be attributed to the fact that basis functions such as splines offer better fitting capabilities than MLPs<sup>[36,49]</sup>, providing significant advantages in solving complex problems such as predicting potential energy surfaces and physical properties of materials.

The Allegro model using B-spline basis functions demonstrated the highest prediction accuracy, likely due to the flexibility of B-splines as piecewise polynomial functions, which are well-suited for approximating complex functions. The Gaussian basis functions, which yield comparable accuracy, are particularly effective for modeling the underlying data distribution. In contrast, the Fourier basis functions, which are particularly effective for capturing periodic or oscillatory patterns in the data, may be less useful than the other two types of basis functions for predicting potential energy surfaces.

However, the Allegro model using B-spline basis functions required significantly longer training times compared to models using other basis functions. This is likely due to the substantial computational time required for operations of calculating the B-spline basis and rescaling the grids<sup>[39,49]</sup>. Employing more efficient basis functions, such as Gaussian and Fourier functions, can significantly accelerate the model calculation with comparable accuracy<sup>[36,39,41]</sup>. Among these, Gaussian-based KANs offer an optimal balance between accuracy and training efficiency, achieving prediction performance similar to B-spline-based KANs with significantly shorter training times. When training other ML methods, the choice of basis functions should be guided by the specific requirements of the application, such as whether accuracy or computational efficiency is the priority.

#### *Replacing some of MLP in Allegro with KAN*

We subsequently replaced MLPs in various parts of the Allegro model with KANs to identify the optimal configuration. The results are shown in [Table 3](#) and [Figure 7](#). Remarkably, the Allegro model incorporating KANs in the output block achieved the highest prediction accuracy, a validation energy MAE of

**Table 2. Results of Allegro with MLPs and KANs with B-spline, Gaussian, and Fourier basis functions**

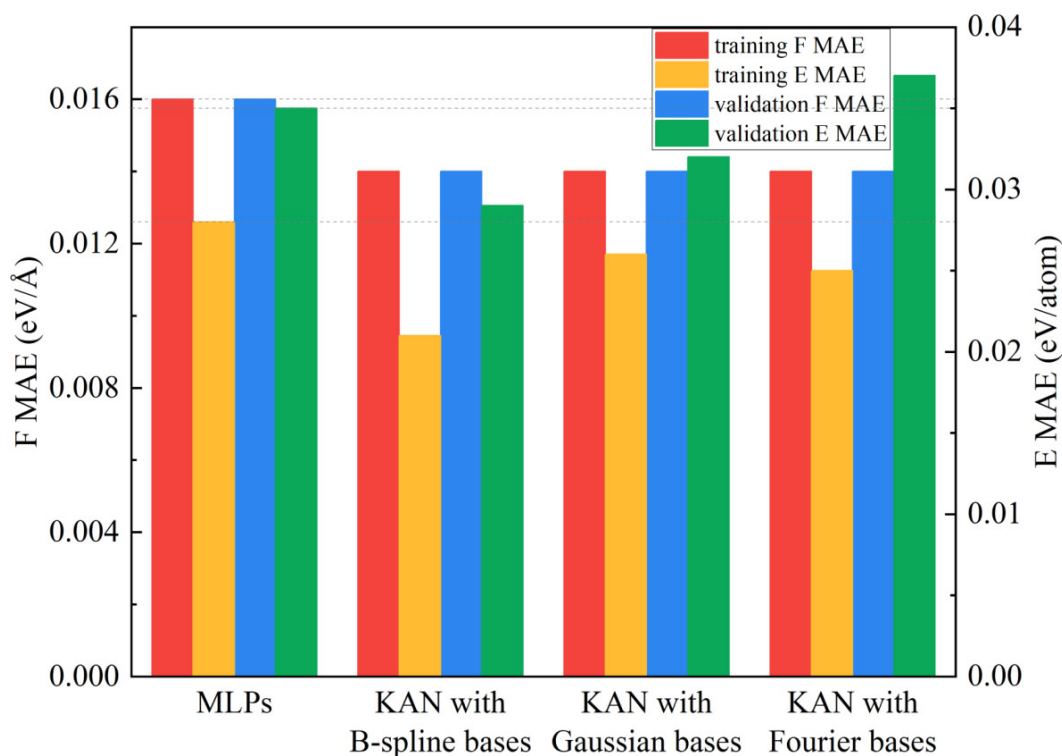
Model	Training F MAE (eV/Å)	Training E MAE (eV/atom)	Validation F MAE (eV/Å)	Validation E MAE (eV/atom)	Training time
Allegro using MLPs	0.016	0.028	0.016	0.035	<b>4h 51m</b>
Allegro using KAN with B-spline bases	<b>0.014</b>	<b>0.021</b>	<b>0.014</b>	<b>0.029</b>	22h 45m
Allegro using KAN with Gaussian bases	<b>0.014</b>	0.026	<b>0.014</b>	0.032	4h 56m
Allegro using KAN with Fourier bases	<b>0.014</b>	0.025	<b>0.014</b>	0.037	6h 54m

The best results are written in bold. MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

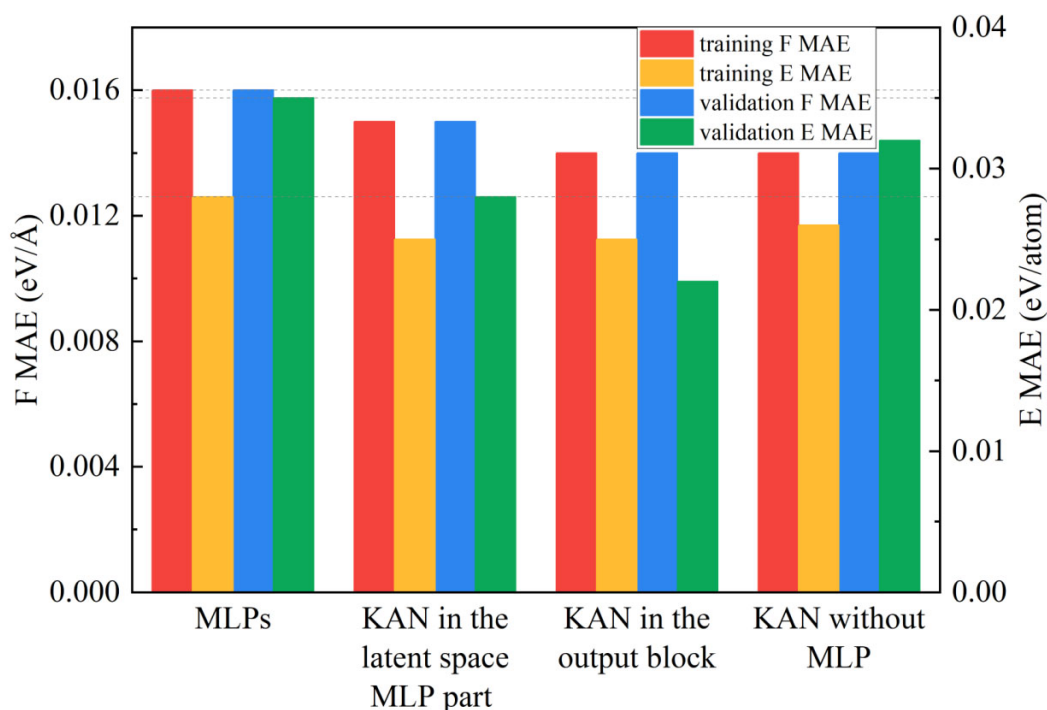
**Table 3. Results of replacing MLP from different parts of Allegro with KAN using Gaussian bases on the Ag dataset**

Model	Training F MAE (eV/Å)	Training E MAE (eV/atom)	Validation F MAE (eV/Å)	Validation E MAE (eV/atom)	Training time
Allegro using MLPs	0.016	0.028	0.016	0.035	<b>4h 51m</b>
Allegro using KAN in the two-body latent embedding and latent MLP part	0.015	<b>0.025</b>	0.015	0.028	5h 20m
Allegro using KAN in the output block	<b>0.014</b>	<b>0.025</b>	<b>0.014</b>	<b>0.022</b>	9h 40m
Allegro using KAN without MLP	<b>0.014</b>	0.026	<b>0.014</b>	0.032	4h 56m

The best results are written in bold. MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.



**Figure 6.** The mean absolute error (MAE) of replacing MLPs in the Allegro model with KANs using various basis functions. All three Allegro models using KANs demonstrated lower force and energy MAE than the original Allegro model with MLPs. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.



**Figure 7.** The mean absolute error (MAE) of replacing MLPs in various components of the Allegro model with KANs on the Ag dataset. All three Allegro models using KANs demonstrated lower force and energy MAE than the original Allegro model with MLPs. Remarkably, the Allegro model incorporating KANs in the output block achieved the highest prediction accuracy. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

0.022 eV/atom, which is 37.1% lower than that of Allegro using MLPs. The Allegro model utilizing KANs in the two-body latent embedding and latent MLP parts demonstrated slightly improved prediction accuracy and reduced training time compared to the model using MLPs.

We also evaluated the inference speeds and GPU memory usage of various models by performing molecular dynamics simulations. The inference speeds and GPU memory usage of various models are shown in Table 4. In general, the Allegro models using KANs with different basis functions exhibited slightly higher GPU memory usage compared to those using MLPs. This suggests that Allegro models employing MLPs are more efficient in terms of model design and data handling, leading to better computation resource efficiency. Replacing only some of the MLPs in the Allegro model with KANs led to a reduction in GPU memory usage. Specifically, the Allegro model with KANs in the output block required 1,945 MB GPU memory, just 4 MB more than the 1,941 MB used by the Allegro model with MLPs. The inference speed of Allegro using KANs was only slightly slower than that of the model using MLPs. Specifically, the inference speed of the Allegro model with KANs in the output block is 8.92 ms per time step, only 0.70 ms per time step slower than the Allegro model using MLPs. Using KAN solely in the output block improves prediction accuracy compared to using MLP, and also improves inference speed and computation resource efficiency compared to using KANs throughout the entire Allegro model.

The improvements observed in the results on the Ag dataset were modest, likely due to the simplicity of the dataset, which significantly limited benefits of KANs<sup>[36]</sup>. Therefore, we proceeded to evaluate these models on the more complex HfO<sub>2</sub> structures. The results, as presented in Table 5 and Figure 8, demonstrate that replacing the MLP in the output block of Allegro significantly improves prediction accuracy for both energies and forces. The validation force MAE is reduced to 0.054 eV/Å, a decrease of 27.0% compared to

**Table 4. The inference speed and GPU memory usage of replacing MLP from different parts of Allegro with KAN using Gaussian bases**

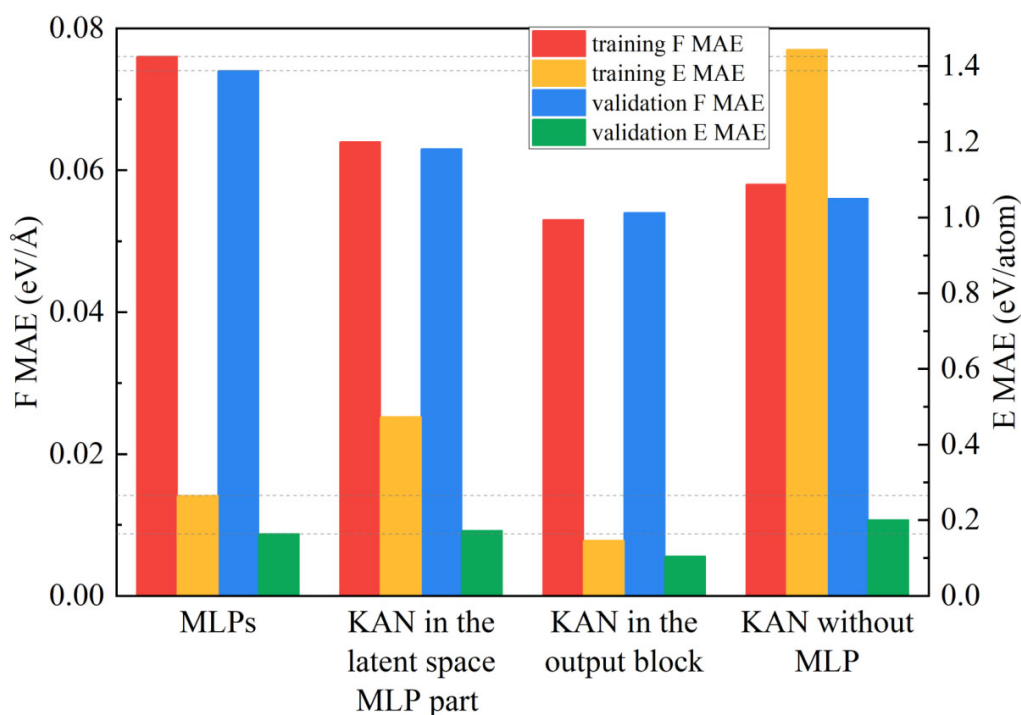
Model	Inference speed (ms per time step)	GPU memory usage (MB)
Allegro using MLPs	8.22	1941
Allegro using KAN in the two-body latent embedding and latent MLP part	9.24	1963
Allegro using KAN in the output block	8.92	1945
Allegro using KAN without MLP	9.44	1963

MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

**Table 5. Results of replacing MLP from different parts of Allegro with KANs using Gaussian bases on the HfO<sub>2</sub> dataset. The best results are written in bold**

Model	Training F MAE (eV/Å)	Training E MAE (eV/atom)	Validation F MAE (eV/Å)	Validation E MAE (eV/atom)	Training time
Allegro using MLPs	0.076	0.265	0.074	0.164	7d 3m
Allegro using KANs in the two-body latent embedding and latent MLP part	0.064	0.473	0.063	0.172	7d 2m
Allegro using KANs in the output block	<b>0.053</b>	<b>0.146</b>	<b>0.054</b>	<b>0.104</b>	<b>4d 11h 40m</b>
Allegro using KANs without MLP	0.058	1.444	0.056	0.200	7d 10m

MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.



**Figure 8.** The mean absolute error (MAE) of replacing MLPs in various components of the Allegro model with KANs on the HfO<sub>2</sub> dataset. Replacing the MLP in the output block of Allegro significantly improves prediction accuracy for both energies and forces. MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

Allegro with MLPs. Similarly, the validation energy MAE is reduced to 0.104 eV/atom, which is 36.6% lower

than with MLPs. Additionally, the training time is notably shortened. For the Allegro model using KAN without MLP, the training F MAE is 0.058 eV/Å, while the training E MAE is 1.444 eV/atom. This discrepancy is attributed to the model's relatively slow convergence speed, resulting in incomplete convergence by the end of the training process. In contrast, the Allegro model using KAN exclusively in the output block effectively combines the advantages of KANs and MLPs. This hybrid configuration leverages the expressive power and flexibility of KANs while retaining the efficiency of MLPs in other parts of the architecture. Consequently, using KANs in the output block facilitates faster convergence during training and better prediction accuracy for both forces and energies. Furthermore, the GPU memory allocated during the training process of the Allegro model using KANs in the output block is 45.63%, only 0.03% higher than using MLP. However, replacing MLPs in other parts of the Allegro model has minimal impact on either prediction accuracy or training time.

These findings are generally consistent with the results obtained from the Ag dataset. The improvements on prediction accuracies and training times are more pronounced in the HfO<sub>2</sub> dataset compared to the Ag dataset. This difference arises from the impact of dataset complexity on the relative performance of KANs versus MLPs. In simpler datasets, such as the Ag dataset, the differences in performance between KANs and MLPs are minimal, as both models can effectively capture the underlying patterns. However, with increasing dataset complexity, KANs tend to outperform MLPs due to their ability to represent more intricate relationships and dependencies within the data<sup>[36]</sup>. Consequently, incorporating KANs in ML models may be particularly advantageous when dealing with datasets with high complexity and variability.

Replacing the MLP in the output block of the Allegro model with KAN significantly improves prediction accuracy. In some cases, this substitution also reduces training time. This improvement occurs because KANs are more effective at fitting functions<sup>[36,49]</sup>. However, basis functions such as splines are less capable of exploiting compositional structures and therefore inferior to MLPs in feature learning<sup>[36]</sup>. Consequently, the output block, which predicts energies from the final layer's output, is well-suited for KANs to enhance prediction accuracy. Therefore, using KANs in other parts of the Allegro model, such as the embedding layer, results in smaller improvements in prediction accuracy compared to the output block.

#### **Machine learning potential NequIP using KAN**

We also investigated replacing MLPs with KANs in the NequIP model<sup>[11]</sup>. The results are shown in [Table 6](#). All three models exhibited similar accuracy, likely due to the simplicity of the Ag dataset<sup>[36,49]</sup>. Additionally, replacing the MLP with the Gaussian bases KAN did not reduce training time. However, substituting the MLP with the B-spline bases KAN significantly shortened the training time.

#### **Machine learning potential MACE using KAN**

We also investigated replacing MLPs with KANs in the MACE model<sup>[50]</sup>. The root-mean-square errors (RMSE) of the forces, energies and stresses on the test set are summarized in [Table 7](#). The MACE models with KANs and MLPs in the output block demonstrate comparable accuracy. Notably, the MACE model using KANs achieves significantly shorter training times compared to using MLPs.

For all three MACE models utilizing KANs in the output block with different random seeds, the results are consistently comparable to those using MLPs. Remarkably, these KAN-based models also demonstrate shorter training times across all scenarios compared to MLP-based models. This result demonstrates the ability of KANs to efficiently learn and generalize despite variations in the initialization parameters, highlighting their robustness, stability, and adaptability.

**Table 6. Results of replacing MLP from the output block of NequIP with KANs using Gaussian bases and B-spline bases on the Ag dataset**

Model	Training F MAE (eV/Å)	Training E MAE (eV/atom)	Validation F MAE (eV/Å)	Validation E MAE (eV/atom)	Training time
NequIP using MLPs	0.011	0.015	0.013	0.015	2d 8h 55m
NequIP using KANs with Gaussian bases in the output block	0.011	0.015	0.013	0.015	2d 11h 46m
NequIP using KANs with B-spline bases in the output block	0.011	0.016	0.013	<b>0.013</b>	<b>1d 13h 2m</b>

The best results are written in bold. NequIP: Neural equivariant interatomic potentials; MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

**Table 7. Results of replacing MLP from the output block of MACE with KANs using B-spline bases on the carbon dataset**

Model	Seed	RMSE F (meV/Å)	RMSE E (meV/atom)	RMSE stress (meV/Å <sup>3</sup> )	Training time
MACE using MLPs	1111	307.6	8.0	119.2	4d 13h 10m
	2222	306.5	8.0	119.4	2d 4h 51m
	3333	309.2	7.8	<b>119.0</b>	2d 12h 30m
MACE using KANs in the output block	1111	309.4	8.1	119.4	1d 42m
	2222	<b>305.1</b>	<b>7.7</b>	119.1	<b>21h 41m</b>
	3333	320.8	8.2	119.4	23h 6m

The best results are written in bold. MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks; MACE: Higher order equivariant message passing neural network; RMSE: Root-mean-square errors.

We also compared the performance of the MACE model using KANs in the output block with models from other literature<sup>[59]</sup>, as summarized in Table 8. The MACE model with KANs demonstrated significantly higher prediction accuracy, highlighting the effectiveness of our approach.

### ETGNN using KAN

We calculated the Born effective charges for the SiO<sub>2</sub> dataset using ETGNN models with MLPs and KANs with Gaussian and B-spline basis functions. The results are shown in Table 9. Replacing the MLP in the output block with a KAN using a Gaussian basis significantly improves prediction accuracy while also reducing training time. The result is consistent with what we achieved on the Allegro model. The training time for ETGNN with KANs using B-spline bases is shorter than with MLPs, and longer compared to using KANs with Gaussian bases.

These results are consistent with the results on the Ag dataset, the HfO<sub>2</sub> dataset and the carbon dataset, indicating that the advantages of KANs, such as improving accuracy and computational efficiency, are consistent across different material systems.

## CONCLUSIONS

In this study, we assessed the impact of replacing MLPs with KANs in various ML models, including ML potentials allegro, NequIP and MACE and property prediction model ETGNN. By systematically replacing MLPs with KANs in different parts of these models, we demonstrated that KANs enhance prediction accuracy. Specifically, replacing MLPs in the output block of the ML model significantly improves accuracy and, in some instances, reduces training time. Moreover, using KANs exclusively in the output block increases inference speed and computation resource efficiency compared to using KANs without MLPs in the property prediction model. Using KANs exclusively in the output block strikes a balance between



**Table 8. Comparison of the performance of the MACE model using KANs in the output block with models from other literature**

Model	Seed	RMSE F (meV/Å)	RMSE E (meV/atom)
MACE using KANs in the output block	1111	309.4	8.1
	2222	305.1	7.7
	3333	320.8	8.2
GAP <sup>[59]</sup>		1100	46
DP <sup>[59]</sup>		800	44
MTP <sup>[59]</sup>		630	35
REANN <sup>[59]</sup>		640	31
NEP <sup>[59]</sup>		690	42
HotPP <sup>[58]</sup>		395	16

MAE: Mean absolute error; F: Force; E: Energy; GAP: Gaussian approximation potential; DP: Deep potential; MTP: Moment tensor potential; REANN: Recursive embedded-atom neural network; NEP: Neuroevolution potential; HotPP: High-order tensor message passing interatomic potential; RMSE: Root-mean-square errors.

**Table 9. Results of replacing MLP from the output block of ETGNN with KAN using Gaussian bases and B-spline bases on the SiO<sub>2</sub> dataset**

Model	Training MAE (e)	Validation MAE (e)	Test MAE (e)	Training time
ETGNN using MLPs	0.00452	0.00517	0.00502	2h 55m
ETGNN using KAN with Gaussian bases in the output block	<b>0.00439</b>	<b>0.00473</b>	<b>0.00450</b>	<b>1h 36m</b>
ETGNN using KAN with B-spline bases in the output block	0.00547	0.00564	0.00542	1h 51m

The best results are written in bold. ETGNN: Edge-based tensor prediction graph neural network; MAE: Mean absolute error; F: Force; E: Energy; MLPs: Multi-layer perceptrons; KANs: Kolmogorov-Arnold Networks.

prediction accuracy, computational efficiency, and resource efficiency. The choice of the optimal basis function for KANs depends on the specific problem.

Our results validate the effectiveness of substituting MLPs with KANs for improving ML models in predicting potential energy surfaces and physical properties. These findings demonstrate the strong potential of KANs in material science. This study offers a promising outlook for extending the use of KANs to broader applications in materials science, where MLPs are commonly employed.

Future research could explore the use of data augmentation techniques<sup>[60]</sup> to further improve the robustness of KAN models. For instance, using synthetic data generation<sup>[61]</sup>, such as generating additional structures using molecular dynamics simulations or perturbing existing datasets, could expand the diversity and size of training data. This approach may enhance the ability of KAN-based models to generalize across a broader range of materials. Additionally, using domain adaptation techniques<sup>[62]</sup>, such as transfer learning<sup>[63]</sup> or fine-tuning KAN models on related datasets, might extend the applicability of KAN-based models to new material classes where labeled data may be limited. Additionally, future research might also investigate incorporating KANs into emerging universal models, high-throughput screening in materials discovery, inverse design methods and other broader applications. For instance, relying on KAN's powerful expressive capabilities, Universal MLPs such as M3GNet<sup>[64]</sup> and CHGNet<sup>[65]</sup>, which are designed to generalize across a wide range of chemical and physical systems, could be able to achieve better prediction accuracy and generalization capability. Additionally, the computational efficiency and improved accuracy of KANs make them highly suitable for high-throughput screening workflows. For example, the accurate and efficient ML potentials developed in this study can be applied in molecular dynamics simulations to precisely determine thermal<sup>[66,67]</sup> and mechanical<sup>[68]</sup> material properties such as thermal conductivity<sup>[69,70]</sup> and elastic modulus<sup>[68]</sup>.

Furthermore, KAN-refined models may offer substantial potential for advancing inverse design, which focuses on identifying material structures with specific target properties<sup>[71-73]</sup>. Many existing inverse design approaches address such problems by first training a predictive model and then exploring the optimum in the design space using the trained model<sup>[74]</sup>. By enhancing the accuracy and efficiency of these models, KANs can enable more precise and effective design processes. Moreover, leveraging the interpretability of KANs<sup>[36]</sup> can also facilitate the discovery of new materials by revealing underlying design principles and guiding inverse design processes. In conclusion, KANs show significant promise for addressing a wide range of scientific and engineering challenges.

## DECLARATIONS

### Authors' contributions

Contributed equally to this work: Wang R, Yu H

Made substantial contributions to the conception and design of the study and performed data analysis and interpretation: Wang R, Yu H

Provided administrative, technical, and material support: Zhong Y, Xiang H

### Availability of data and materials

Details for the networks, datasets and the LAMMPS simulations are contained in [Supplementary Materials](#). The source code of the MACE model using KANs in the output block is available at: <https://github.com/Hongyu-yu/mace-kan>.

### Financial support and sponsorship

We acknowledge financial support from the National Key R&D Program of China (No. 2022YFA1402901), NSFC (grants Nos. 11991061 and 12188101), Shanghai Science and Technology Program (No. 23JC1400900), and the Guangdong Major Project of the Basic and Applied Basic Research (Future functional materials under extreme conditions-2021B0301030005).

### Conflicts of interest

All authors declared that there are no conflicts of interest.

### Ethical approval and consent to participate

Not applicable.

### Consent for publication

Not applicable.

### Copyright

© The Author(s) 2024.

## REFERENCES

1. Frank JT, Unke OT, Müller KR, Chmiela S. A Euclidean transformer for fast and stable machine learned force fields. *Nat Commun* 2024;15:6539. [DOI](#) [PubMed](#) [PMC](#)
2. Choung S, Park W, Moon J, Han JW. Rise of machine learning potentials in heterogeneous catalysis: developments, applications, and prospects. *Chem Eng J* 2024;494:152757. [DOI](#)
3. Tang D, Ketkaew R, Lubner S. Machine learning interatomic potentials for heterogeneous catalysis. *Chem A Eur J* 2024;30:e202401148. [DOI](#)
4. Damewood J, Karaguesian J, Lunger JR, et al. Representations of materials for machine learning. *Annu Rev Mater Res* 2023;53:399-426. [DOI](#)
5. Song Z, Chen X, Meng F, et al. Machine learning in materials design: algorithm and application\*. *Chinese Phys B* 2020;29:116103. [DOI](#)

6. Dieb S, Song Z, Yin W, Ishii M. Optimization of depth-graded multilayer structure for x-ray optics using machine learning. *J Appl Phys* 2020;128:074901. DOI
7. Cheng G, Gong XG, Yin WJ. Crystal structure prediction by combining graph network and optimization algorithm. *Nat Commun* 2022;13:1492. DOI PubMed PMC
8. Zendejboudi S, Rezaei N, Lohi A. Applications of hybrid models in chemical, petroleum, and energy systems: a systematic review. *Appl Energy* 2018;228:2539-66. DOI
9. Leukel J, Scheurer L, Sugumaran V. Machine learning models for predicting physical properties in asphalt road construction: a systematic review. *Constr Build Mater* 2024;440:137397. DOI
10. Musaelian A, Batzner S, Johansson A, et al. Learning local equivariant representations for large-scale atomistic dynamics. *Nat Commun* 2023;14:579. DOI PubMed PMC
11. Batzner S, Musaelian A, Sun L, et al. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat Commun* 2022;13:2453. DOI PubMed PMC
12. Thölke P, Fabritiis GD. Equivariant transformers for neural network based molecular potentials. International Conference on Learning Representations. 2022. Available from: <https://openreview.net/forum?id=zNHZqZ9wrRB>. [Last accessed on 27 Dec 2024].
13. Wang G, Wang C, Zhang X, Li Z, Zhou J, Sun Z. Machine learning interatomic potential: bridge the gap between small-scale models and realistic device-scale simulations. *iScience* 2024;27:109673. DOI PubMed PMC
14. Noda K, Shibuta Y. Prediction of potential energy profiles of molecular dynamic simulation by graph convolutional networks. *Comput Mater Sci* 2023;229:112448. DOI
15. Yu H, Zhong Y, Hong L, et al. Spin-dependent graph neural network potential for magnetic materials. *Phys Rev B* 2024;109:14426. DOI
16. Vandenhoute S, Cools-ceuppens M, Dekeyser S, Verstraelen T, Van Speybroeck V. Machine learning potentials for metal-organic frameworks using an incremental learning approach. *npj Comput Mater* 2023;9:1-8. DOI
17. Song K, Zhao R, Liu J, et al. General-purpose machine-learned potential for 16 elemental metals and their alloys. Available from: <http://arxiv.org/abs/2311.04732>. [Last accessed on 27 Dec 2024].
18. Sun H, Zhang C, Tang L, Wang R, Xia W, Wang C. Molecular dynamics simulation of Fe-Si alloys using a neural network machine learning potential. *Phys Rev B* 2023;107:224301. DOI
19. Kostiuchenko TS, Shapeev AV, Novikov IS. Interatomic interaction models for magnetic materials: recent advances. *Chinese Phys Lett* 2024;41:066101. DOI
20. Fan Z, Chen W, Vierimaa V, Harju A. Efficient molecular dynamics simulations with many-body potentials on graphics processing units. *Comput Phys Commun* 2017;218:10-6. DOI
21. Zhong Y, Yu H, Gong X, Xiang H. A general tensor prediction framework based on graph neural networks. *J Phys Chem Lett* 2023;14:6339-48. DOI
22. Zhong Y, Yu H, Su M, Gong X, Xiang H. Transferable equivariant graph neural networks for the hamiltonians of molecules and solids. *npj Comput Mater* 2023;9:182. DOI
23. Zhong Y, Yu H, Yang J, Guo X, Xiang H, Gong X. Universal machine learning kohn-sham hamiltonian for materials. *Chinese Phys Lett* 2024;41:077103. DOI
24. Li H, Wang Z, Zou N, et al. Deep-learning density functional theory hamiltonian for efficient ab initio electronic-structure calculation. *Nat Comput Sci* 2022;2:367-77. DOI PubMed PMC
25. Zhong Y, Liu S, Zhang B, et al. Accelerating the calculation of electron-phonon coupling strength with machine learning. *Nat Comput Sci* 2024;4:615-25. DOI
26. Zhang C, Zhong Y, Tao ZG, et al. Advancing nonadiabatic molecular dynamics simulations for solids: achieving supreme accuracy and efficiency with machine learning. Available from: <https://arxiv.org/html/2408.06654v1>. [Last accessed on 27 Dec 2024].
27. Xie T, Grossman JC. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys Rev Lett* 2018;120:145301. DOI PubMed
28. Choudhary K, Decost B. Atomistic line graph neural network for improved materials property predictions. *npj Comput Mater* 2021;7:185. DOI
29. Choudhary K, Garrity K. Designing high- $T_c$  superconductors with BCS-inspired screening, density functional theory, and deep-learning. *npj Comput Mater* 2022;8:244. DOI
30. Choudhary K, Garrity KF, Sharma V, Biacchi AJ, Walker ARH, Tavazza F. High-throughput density functional perturbation theory and machine learning predictions of infrared, piezoelectric and dielectric responses. *npj Comput Mater* 2020;6:64. DOI PubMed PMC
31. Clayson IG, Hewitt D, Hutereau M, Pope T, Slater B. High throughput methods in the synthesis, characterization, and optimization of porous materials. *Adv Mater* 2020;32:e2002780. DOI PubMed
32. Wang R, Yu H, Zhong Y, Xiang H. Identifying direct bandgap silicon structures with high-throughput search and machine learning methods. *J Phys Chem C* 2024;128:12677-85. DOI
33. Stergiou K, Ntakolia C, Varytis P, Koumoulos E, Karlsson P, Moustakidis S. Enhancing property prediction and process optimization in building materials through machine learning: a review. *Comput Mater Sci* 2023;220:112031. DOI
34. Cybenko G. Approximation by superpositions of a sigmoidal function. *Math Control Signal Syst* 1989;2:303-14. DOI
35. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. *Neural Netw* 1989;2:359-66. DOI

36. Liu Z, Wang Y, Vaidya S, et al. KAN: Kolmogorov-Arnold Networks. Available from: <http://arxiv.org/abs/2404.19756>. [Last accessed on 27 Dec 2024].
37. Braun J, Griebel M. On a constructive proof of kolmogorov's superposition theorem. *Constr Approx* 2009;30:653-75. DOI
38. Arnold VI. On the representation of functions of several variables as a superposition of functions of a smaller number of variables. In: Givental AB, Khesin BA, Marsden JE, Varchenko AN, Vassiliev VA, Viro OY, Zakalyukin VM, editors. *Collected Works*. Berlin: Springer Berlin Heidelberg; 2009. pp. 25-46. DOI
39. Li Z. Kolmogorov-Arnold Networks are radial basis function networks. Available from: <http://arxiv.org/abs/2405.06721>. [Last accessed on 27 Dec2024].
40. Bozorgasl Z, Chen H. Wav-KAN: Wavelet Kolmogorov-Arnold Networks. Available from: <https://arxiv.org/abs/2405.12832>. [Last accessed on 27 Dec2024].
41. Xu J, Chen Z, Li J, et al. FourierKAN-GCF: Fourier Kolmogorov-Arnold Network - an effective and efficient feature transformation for graph collaborative filtering. Available from: <http://arxiv.org/abs/2406.01034>. [Last accessed on 27 Dec2024].
42. Aghaei AA. fKAN: Fractional Kolmogorov-Arnold Networks with trainable Jacobi basis functions. Available from: <http://arxiv.org/abs/2406.07456>. [Last accessed on 27 Dec2024].
43. Reinhardt EAF, Dinesh PR, Gleyzer S. SineKAN: Kolmogorov-Arnold Networks using sinusoidal activation functions. Available from: <http://arxiv.org/abs/2407.04149>. [Last accessed on 27 Dec2024].
44. Nagai Y, Okumura M. Kolmogorov-Arnold Networks in molecular dynamics. Available from: <https://arxiv.org/abs/2407.17774>. [Last accessed on 27 Dec2024].
45. Genet R, Inzirillo H. TKAN: Temporal Kolmogorov-Arnold Networks. Available from: <https://arxiv.org/abs/2405.07344>. [Last accessed on 27 Dec2024].
46. Kiamari M, Kiamari M, Krishnamachari B. GKAN: Graph Kolmogorov-Arnold Networks. Available from: <http://arxiv.org/abs/2406.06470>. [Last accessed on 27 Dec2024].
47. Inzirillo H, Genet R. SigKAN: Signature-Weighted Kolmogorov-Arnold Networks for rime series. Available from: <http://arxiv.org/abs/2406.17890>. [Last accessed on 27 Dec2024].
48. Bresson R, Nikolentzos G, Panagopoulos G, Chatzianastasis M, Pang J, Vazirgiannis M. KAGNNs: Kolmogorov-Arnold Networks meet graph learning. Available from: <http://arxiv.org/abs/2406.18380>. [Last accessed on 27 Dec2024].
49. Wang Y, Sun J, Bai J, et al. Kolmogorov-arnold-informed neural network: a physics-informed deep learning framework for solving forward and inverse problems based on kolmogorov-arnold networks. *Comput Methods Appl Mech Eng* 2025;433:117518. DOI
50. Batatia I, Kovacs DP, Simm GNC, Ortner C, Csanyi G. MACE: higher order equivariant message passing neural networks for fast and accurate force fields. 2022. Available from: <https://openreview.net/forum?id=YPPSngE-ZU>. [Last accessed on 27 Dec2024].
51. Gilmer J, Schoenholz SS, Riley PF, Vinyals O, Dahl GE. Neural message passing for quantum chemistry. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. Sydney, NSW, Australia: JMLR.org; 2017. p. 1263-72. (ICML'17). Available from: <https://dl.acm.org/doi/10.5555/3305381.3305512>. [Last accessed on 28 Dec2024].
52. Blealtan/efficient-kan. Available from: <https://github.com/Blealtan/efficient-kan>. [Last accessed on 27 Dec2024].
53. Kresse G, Hafner J. Ab initio molecular dynamics for liquid metals. *Phys Rev B* 1993;47:558-61. DOI
54. Perdew JP, Burke K, Ernzerhof M. Generalized gradient approximation made simple. *Phys Rev Lett* 1997;78:1396-1396. DOI
55. Thompson AP, Aktulga HM, Berger R, et al. LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput Phys Commun* 2022;271:108171. DOI
56. Wu J, Zhang Y, Zhang L, Liu S. Deep learning of accurate force field of ferroelectric HfO<sub>2</sub>. *Phys Rev B* 2021;103:024108. DOI
57. Deringer VL, Csányi G. Machine learning based interatomic potential for amorphous carbon. *Phys Rev B* 2017;95. DOI
58. Wang J, Wang Y, Zhang H, et al. E(n)-equivariant cartesian tensor message passing interatomic potential. *Nat Commun* 2024;15:7607. DOI PubMed PMC
59. Fan Z, Wang Y, Ying P, et al. GPUMD: a package for constructing accurate machine-learned potentials and performing highly efficient atomistic simulations. *J Chem Phys* 2022;157:114801. DOI
60. Mumuni A, Mumuni F. Data augmentation: a comprehensive survey of modern approaches. *Array* 2022;16:100258. DOI
61. Lu Y, Shen M, Wang H, Wang X, van Rechem C, Fu T, Wei W. Machine learning for synthetic data generation: a review. Available from: <https://arxiv.org/abs/2302.04062>. [Last accessed on 27 Dec2024].
62. Farahani A, Voghoei S, Rasheed K, Arabnia HR. A brief review of domain adaptation. In: Stahlbock R, Weiss GM, Abou-nasr M, Yang C, Arabnia HR, Deligiannidis L, editors. *Advances in data science and information engineering*. Cham: Springer International Publishing; 2021. pp. 877-94. DOI
63. Zhuang F, Qi Z, Duan K, et al. A comprehensive survey on transfer learning. *Proc IEEE* 2021;109:43-76. DOI
64. Chen C, Ong SP. A universal graph deep learning interatomic potential for the periodic table. *Nat Comput Sci* 2022;2:718-28. DOI PubMed
65. Deng B, Zhong P, Jun K, et al. CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nat Mach Intell* 2023;5:1031-41. DOI
66. Arabha S, Aghbolagh ZS, Ghorbani K, Hatam-lee SM, Rajabpour A. Recent advances in lattice thermal conductivity calculation using machine-learning interatomic potentials. *J Appl Phys* 2021;130:210903. DOI
67. Qian X, Yang R. Machine learning for predicting thermal transport properties of solids. *Mater Sci Eng R Rep* 2021;146:100642. DOI
68. Mortazavi B, Zhuang X, Rabczuk T, Shapeev AV. Atomistic modeling of the mechanical properties: the rise of machine learning

- interatomic potentials. *Mater Horiz* 2023;10:1956-68. [DOI](#) [PubMed](#)
69. Mortazavi B, Podryabinkin EV, Roche S, Rabczuk T, Zhuang X, Shapeev AV. Machine-learning interatomic potentials enable first-principles multiscale modeling of lattice thermal conductivity in graphene/borophene heterostructures. *Mater Horiz* 2020;7:2359-67. [DOI](#)
  70. Luo Y, Li M, Yuan H, Liu H, Fang Y. Predicting lattice thermal conductivity via machine learning: a mini review. *npj Comput Mater* 2023;9:964. [DOI](#)
  71. Kim Y, Yang C, Kim Y, Gu GX, Ryu S. Designing an adhesive pillar shape with deep learning-based optimization. *ACS Appl Mater Interfaces* 2020;12:24458-65. [DOI](#)
  72. Yu CH, Chen W, Chiang YH, et al. End-to-end deep learning model to predict and design secondary structure content of structural proteins. *ACS Biomater Sci Eng* 2022;8:1156-65. [DOI](#) [PubMed](#) [PMC](#)
  73. Zhang Z, Zhang Z, Di Caprio F, Gu GX. Machine learning for accelerating the design process of double-double composite structures. *Compos Struct* 2022;285:115233. [DOI](#)
  74. Lee J, Park D, Lee M, et al. Machine learning-based inverse design methods considering data characteristics and design space size in materials design and manufacturing: a review. *Mater Horiz* 2023;10:5436-56. [DOI](#)