Deep Network Approximation: Achieving Arbitrary Accuracy with Fixed Number of Neurons*

Zuowei Shen† Haizhao Yang‡ Shijun Zhang§

Abstract

This paper develops simple feed-forward neural networks that achieve the universal approximation property for all continuous functions with a fixed finite number of neurons. These neural networks are simple because they are designed with a simple and computable continuous activation function σ leveraging a triangular-wave function and a softsign function. We prove that σ-activated networks with width $36d(2d+1)$ and depth 11 can approximate any continuous function on a d-dimensional hypercube within an arbitrarily small error. Hence, for supervised learning and its related regression problems, the hypothesis space generated by these networks with a size not smaller than $36d(2d+1)\times 11$ is dense in the space of continuous functions. Furthermore, classification functions arising from image and signal classification are in the hypothesis space generated by σ-activated networks with width $36d(2d+1)$ and depth 12, when there exist pairwise disjoint closed bounded subsets of $\mathbb{R}^d$ such that the samples of the same class are located in the same subset.

Key words. Nonlinear Approximation; Universal Approximation Theorem; Fixed-Size Neural Network; Periodic Function; Continuous Function; Classification Function.

1 Introduction

Deep neural networks have been widely used in data science and artificial intelligence. Their tremendous successes in various applications have motivated extensive research to establish the theoretical foundation of deep learning. Understanding the approximation capacity of deep neural networks is one of the keys to revealing the power of deep learning. The most basic layers of deep neural networks are nonlinear functions as the composition of an affine linear transform and a nonlinear activation function. The composition of these simple nonlinear functions can generate a complicated deep neural network with powerful approximation capacity, which is the key difference to classic approximation tools. In this paper, we show that the hypothesis space of deep neural networks generated from the composition of 11 such simple nonlinear functions is

*Submitted to the editors DATE.
†Department of Mathematics, National University of Singapore (matzuows@nus.edu.sg).
‡Department of Mathematics, Purdue University (haizhao@purdue.edu).
§Department of Mathematics, National University of Singapore (zhangshijun@u.nus.edu).

dense in the continuous function space $C([a,b]^d)$, when the affine linear transforms are parameterized with $\mathcal{O}(d^2)$ parameters in total and the nonlinear activation function is constructed from a simple triangular-wave function and a softsign function.

1.1 Main results

One of the key elements of a neural network is its activation functions. Searching for simple activation functions enabling powerful approximation capacity of neural networks is an important mathematical problem that probably originated in the Kolmogorov superposition theorem (KST) [29] for Hilbert's 13-th problem, where a two-hidden-layer neural network with $\mathcal{O}(d)$ neurons and complicated activation functions depending on the target functions are constructed to represent an arbitrary function in $C([0,1]^d)$. Since then, whether simple and computable activation functions independent of the target function exist to make the space of neural networks with $\mathcal{O}(d)$ neurons dense in $C([0,1]^d)$ or even equal to $C([0,1]^d)$ has been an open problem. A function $\varrho:\mathbb{R}\to\mathbb{R}$ is said to be a universal activation function (UAF) if the function space generated by ϱ-activated networks with $C_{\varrho,d}$ neurons is dense in $C([0,1]^d)$, where $C_{\varrho,d}$ is a constant determined by ϱ and d. That is, if ϱ is a UAF, then ϱ-activated networks with $C_{\varrho,d}$ neurons can approximate any continuous function within an arbitrary error on $[0,1]^d$ by only adjusting the parameters.

In this paper, we first construct a simple and computable example of UAFs. As a typical and simple UAF, this activation function is called the elementary universal activation function (EUAF), and the corresponding networks are called EUAF networks. Then, we prove that the function space generated by EUAF networks with $\mathcal{O}(d^2)$ neurons is dense in $C([a,b]^d)$. Furthermore, it is shown that EUAF networks with $\mathcal{O}(d^2)$ neurons can exactly represent d-dimensional classification functions.

While a good activation function should be simple and numerically implementable, the neural network activated by it should be able to approximate continuous functions well with a manageable size. Considering these requirements and motivated by previous works [41, 42, 50], the activation function to be chosen should have appropriate nonlinearity, periodicity, and the capacity to reproduce step functions. It is challenging to find a single activation function with all these proprieties. Here, we propose an activation function with all required properties by using two simple functions σ_1 and σ_2 defined below.

Let σ_1 be the continuous triangular-wave function with period 2, i.e.,

$$\sigma_1(x) \coloneqq |x| \quad \text{for any } x\in[-1,1] \tag{1.1}$$

and $\sigma_1(x+2)=\sigma_1(x)$ for any $x\in\mathbb{R}$. Alternatively, σ_1 can also be written as:

$$\sigma_1(x) = \left|x - 2\left\lfloor \tfrac{x+1}{2}\right\rfloor\right| \quad \text{for any } x\in\mathbb{R}, \qquad \text{where } \lfloor\cdot\rfloor \text{ is the floor function.}$$

Clearly, σ_1 is periodic and $x-\sigma_1(x)$ is a continuous variant of the floor function as desired.

To introduce high nonlinearity, let σ_2 be the softsign activation function commonly used in machine learning [30, 46]:

$$\sigma_2(x) \coloneqq \frac{x}{|x|+1} \quad \text{for any } x\in\mathbb{R}. \tag{1.2}$$

Then the activation function σ is defined as:

$$\sigma(x) \coloneqq \begin{cases} \sigma_1(x) & \text{for } x \in [0,\infty), \\ \sigma_2(x) & \text{for } x \in (-\infty,0). \end{cases} \tag{1.3}$$

See an illustration of σ in Figure 1. This activation function σ is the EUAF used to construct powerful neural networks in this paper.

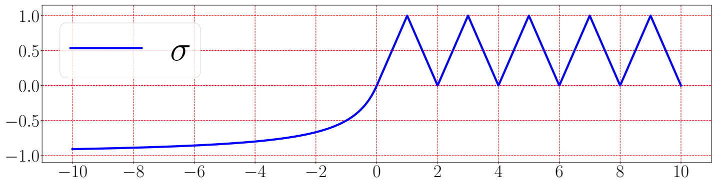


Figure 1: An illustration of σ on $[-10,10]$.

The periodicity of the triangular-wave function σ_1 and the nonlinearity of the softsign function σ_2 play crucial roles in the proof of our main results. Observing that σ_1 is an even function and σ_2 is an odd function, i.e., $\sigma(x) = \sigma_1(x) = \sigma_1(-x)$ for any $x \geq 0$ and $-\sigma(-x) = -\sigma_2(-x) = \sigma_2(x)$ for any $x \geq 0$. This implies that $\sigma(x)$ and $-\sigma(-x)$ with $x \geq 0$ have both required periodicity and nonlinearity features and play the same roles as $\sigma_1(x)$ and $\sigma_2(x)$, respectively. These requirements lead to our choice of σ as the activation function. If allowed to be more complicated, one can design many other UAFs satisfying stronger requirements for various applications. For example, the idea of designing a C^s UAF is given in Section 5.1 and a sigmoidal UAF (see Figure 15) is constructed in Section 5.2.

With the activation function σ in hand, let us introduce the network (architecture) using σ as the activation function, called σ-activated network (architecture). To be precise, a σ-activated network with a (vector) input $\boldsymbol{x} \in \mathbb{R}^d$, an output $\Phi(\boldsymbol{x},\boldsymbol{\theta}) \in \mathbb{R}$, and $L \in \mathbb{N}^+$ hidden layers can be briefly described as follows:

$$\boldsymbol{x} = \widetilde{\boldsymbol{h}}_0 \xrightarrow[\mathcal{L}_0]{\boldsymbol{A}_0,\ \boldsymbol{b}_0} \boldsymbol{h}_1 \xrightarrow{\sigma} \widetilde{\boldsymbol{h}}_1 \quad \cdots \quad \xrightarrow[\mathcal{L}_{L-1}]{\boldsymbol{A}_{L-1},\ \boldsymbol{b}_{L-1}} \boldsymbol{h}_L \xrightarrow{\sigma} \widetilde{\boldsymbol{h}}_L \xrightarrow[\mathcal{L}_L]{\boldsymbol{A}_L,\ \boldsymbol{b}_L} \boldsymbol{h}_{L+1} = \Phi(\boldsymbol{x},\boldsymbol{\theta}), \tag{1.4}$$

where $N_0 = d \in \mathbb{N}^+$, $N_1, N_2, \cdots, N_L \in \mathbb{N}^+$, $N_{L+1} = 1$, $\boldsymbol{A}_i \in \mathbb{R}^{N_{i+1}\times N_i}$ and $\boldsymbol{b}_i \in \mathbb{R}^{N_{i+1}}$ are the weight matrix and the bias vector in the i-th affine linear transform $\mathcal{L}_i$, respectively, i.e.,

$$\boldsymbol{h}_{i+1} = \boldsymbol{A}_i \cdot \widetilde{\boldsymbol{h}}_i + \boldsymbol{b}_i \eqqcolon \mathcal{L}_i(\widetilde{\boldsymbol{h}}_i) \quad \text{for } i = 0, 1, \cdots, L$$

and

$$\widetilde{h}_{i,j} = \sigma(h_{i,j}) \quad \text{for } j = 1, 2, \cdots, N_i \text{ and } i = 1, 2, \cdots, L.$$

Here, $\widetilde{h}_{i,j}$ and $h_{i,j}$ are the j-th entry of $\widetilde{\boldsymbol{h}}_i$ and $\boldsymbol{h}_i$, respectively, for $j = 1, 2, \cdots, N_i$ and $i = 1, 2, \cdots, L$. $\boldsymbol{\theta}$ is a fattened vector consisting of all parameters in $\boldsymbol{A}_0, \boldsymbol{b}_0, \cdots, \boldsymbol{A}_L, \boldsymbol{b}_L$.

If σ is applied to a vector entry wisely, i.e., given any $k \in \mathbb{N}^+$,

$$\sigma(\boldsymbol{y}) = \big[\sigma(y_1), \cdots, \sigma(y_k)\big]^T \quad \text{for any } \boldsymbol{y} = [y_1, \cdots, y_k]^T \in \mathbb{R}^k,$$

then Φ can be represented in a form of function compositions as follows:

$$\Phi(\boldsymbol{x},\boldsymbol{\theta}) = \mathcal{L}_L \circ \sigma \circ \mathcal{L}_{L-1} \circ \sigma \circ \ \cdots \ \circ \sigma \circ \mathcal{L}_1 \circ \sigma \circ \mathcal{L}_0(\boldsymbol{x}) \quad \text{for any } \boldsymbol{x} \in \mathbb{R}^d.$$

Given $N, L \in \mathbb{N}^+$, let $\Phi_{N,L}(\boldsymbol{x}, \boldsymbol{\theta})$ denote the σ-activated network architecture $\Phi(\boldsymbol{x}, \boldsymbol{\theta})$ in Equation (1.4) with $N_1 = N_2 = \cdots = N_L = N$. Let

$$W = W_{d,N,L} = d \times N + N \ + \ (N \times N + N) \times (L-1) \ + \ 1 \times N + 1 = \mathcal{O}(dN + N^2 L)$$

be the total number of parameters in $\Phi_{N,L}(\boldsymbol{x}, \boldsymbol{\theta})$, i.e., $\boldsymbol{\theta} \in \mathbb{R}^W$.

Define the hypothesis space $\mathscr{H}_d(N, L)$ as the function space generated by EUAF networks with width N and depth L, i.e.,

$$\mathscr{H}_d(N,L) \coloneqq \Big\{ \phi : \phi(\boldsymbol{x}) = \Phi_{N,L}(\boldsymbol{x}, \boldsymbol{\theta}) \text{ for any } \boldsymbol{x} \in \mathbb{R}^d, \quad \boldsymbol{\theta} \in \mathbb{R}^W \Big\}. \tag{1.5}$$

Let $C([a,b]^d)$ be the space of all continuous functions $f : [a,b]^d \to \mathbb{R}$ with the maximum norm. Our first main result, Theorem 1.1 below, shows that σ-activated networks with a fixed size $\mathcal{O}(d^2)$ enjoy the universal approximation property by only adjusting their parameters.

Theorem 1.1. *Let $f \in C([a,b]^d)$ be a continuous function and $\mathscr{H}_d(N,L)$ be the hypothesis space defined in (1.5) with $N = 36d(2d+1)$ and $L = 11$. Then, for an arbitrary $\varepsilon > 0$, there exists $\phi \in \mathscr{H}_d(N,L)$ such that*

$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Remark. The network realizing ϕ in Theorem 1.1 has

$$d \times N + N \ + \ (N \times N + N) \times (L-1) \ + \ N \times 1 + 1 \sim d^4$$

parameters, where $N = 36d(2d+1)$ and $L = 11$. However, as shown in our constructive proof of Theorem 1.1, it is enough to adjust $5437(d+1)(2d+1) = \mathcal{O}(d^2) \ll d^4$ parameters and set all the others to 0.

Since for an arbitrary $M > 0$, $2M\sigma(\frac{x+M}{2M}) - M = x$ for all $x \in [-M, M]$, we can manually add hidden layers to EUAF networks without changing the output. This leads to the following immediate corollary of Theorem 1.1.

Corollary 1.2. *Assume $N \geq 36d(2d+1)$ and $L \geq 11$, then the hypothesis space $\mathscr{H}_d(N,L)$ defined in (1.5) is dense in $C([a,b]^d)$.*

The stable and accurate approximation of discontinuities has many real applications and has been widely studied [2,4,18,19,25]. Most of common discontinuous functions are in Lebesgue spaces. Let us consider the denseness of our hypothesis space in Lebesgue spaces. Since $C([a,b]^d)$ is dense in $L^p([a,b]^d)$ for $p \in [1,\infty)$ (e.g., see Theorem 2.4 of [45]), the hypothesis space in Corollary 1.2 is also dense in $L^p([a,b]^d)$ as shown in the following corollary.

Corollary 1.3. *Assume $N \geq 36d(2d+1)$, $L \geq 11$, and $p \in [1,\infty)$, then the hypothesis space $\mathscr{H}_d(N,L)$ defined in (1.5) is dense in $L^p([a,b]^d)$.*

This corollary implies that, for $f \in L^p([a,b]^d)$ and an arbitrary $\varepsilon > 0$, there exists $\phi \in \mathscr{H}_d(N,L)$ such that $\|\phi - f\|_{L^p([a,b]^d)} \leq \varepsilon$.

One can ask whether the arbitrary error $\varepsilon > 0$ in Theorem 1.1 can be further reduced to 0. This is not true in general, but it is true for a class of interesting functions widely used in image classifications. Given any pairwise disjoint closed bounded subsets $E_1, E_2, \cdots, E_J \subseteq \mathbb{R}^d$, define "the classification function space" of these subsets as

$$\mathscr{C}_d(E_1, E_2, \cdots, E_J) \coloneqq \left\{ f : f = \sum_{j=1}^{J} r_j \cdot \mathbb{1}_{E_j} \text{ for any } \ r_1, r_2, \cdots, r_J \in \mathbb{Q} \right\},$$

where $\mathbb{1}_{E_n}$ is the indicator function of E_j for each j. Our second main result, Theorem 1.4 below, shows that each element of $\mathscr{C}_d(E_1, E_2, \cdots, E_J)$ can be exactly represented by a σ-activated network with $\mathcal{O}(d^2)$ neurons in $\bigcup_{j=1}^{J} E_j$.

Theorem 1.4. *Let $E_1, E_2, \cdots, E_J \subseteq \mathbb{R}^d$ be pairwise disjoint closed bounded subsets and $\mathscr{H}_d(N, L)$ be the hypothesis space defined in (1.5) with $N = 36d(2d+1)$ and $L = 12$. Then, for $f \in \mathscr{C}_d(E_1, E_2, \cdots, E_J)$, there exists $\phi \in \mathscr{H}_d(N, L)$ such that*

$$\phi(\boldsymbol{x}) = f(\boldsymbol{x}) \quad \text{for any } \boldsymbol{x} \in \bigcup_{j=1}^{J} E_j.$$

Remark. The network realizing ϕ in Theorem 1.4 has

$$d \times N + N \ + \ (N \times N + N) \times (L-1) \ + \ N \times 1 + 1 \sim d^4$$

parameters, where $N = 36d(2d+1)$ and $L = 12$. However, as shown in our constructive proof of Theorem 1.4, it is enough to adjust $5509(d+1)(2d+1) = \mathcal{O}(d^2) \ll d^4$ parameters and set all the others to 0.

For a general function space $\mathscr{F}$, define $\mathscr{F}|_E \coloneqq \left\{ f|_E : f \in \mathscr{F} \right\}$, where $f|_E$ is the function achieved via limiting f on E. Then, we have a corollary of Theorem 1.4 as follows.

Corollary 1.5. *Let $E_1, E_2, \cdots, E_J \subseteq \mathbb{R}^d$ be pairwise disjoint closed bounded subsets and $\mathscr{H}_d(N, L)$ be the hypothesis space defined in (1.5). Assume $N \geq 36d(2d+1)$ and $L \geq 12$, then*

$$\mathscr{C}_d(E_1, E_2, \cdots, E_J)\big|_E \subseteq \mathscr{H}_d(N, L)\big|_E,$$

where $E = \bigcup_{j=1}^{J} E_j$.

One of the most successful applications of deep learning is the image and signal classifications. In supervised classification problems, given a few samples and their labels (usually integers), the goal of the task is to learn how to assign a label to a new sample. For example, in binary classification via deep learning, a neural network is trained based on given samples (and labels) to approximate a classification function mapping one class of samples to 0 and the other class of samples to 1. Theorem 1.4 (or Corollary 1.5) implies that the classification function can be exactly realized by an EUAF network with a size depending only on the dimension of the problem domain via adjusting its parameters. This means that the best approximation error of EUAF networks to classification functions in the classification problem is 0.

Remark that, in the worst scenario, there might exist complicated high-dimensional functions such that, the parameters of the EUAF network in Theorem 1.1 (or 1.4) require high computer precision for storage, and the precision might be exponentially large in the problem dimension. We refer to this as the curse of memory, which may make Theorem 1.1 and 1.4 less interesting in real applications, though the number of parameters can be very small. The key question to be addressed is how rare the curse of memory would happen in real applications. If the target functions in real applications typically have no curse of memory with a high probability, then EUAF networks would be very useful in real applications. In future work, we will explore the statistical characterization of high-dimensional functions for the curse of memory of EUAF networks. Another approach to reducing the memory requirement is to increase the network size. Our main result has provided a network size $\mathcal{O}(d^2)$ to achieve an arbitrary error. If a larger network size is used, the curse of memory can be lessened as we shall discuss in Section 1.4.

1.2 Related work

In recent years, there has been an increasing amount of literature on the approximation power of neural networks as a special case of nonlinear approximation [8,10,11]. In the early works of approximation theory for neural networks, the universal approximation theorem [9,23,24] without approximation errors showed that there exists a sufficiently large neural network approximating a target function in a certain function space within any given error $\varepsilon > 0$. There are also other versions of the universal approximation theorem. For example, it was shown in [33] that the ReLU-activated residual neural networks with one neuron per hidden layer and a sufficiently large depth are a universal approximator. The universal approximation property for general residual neural networks was proved in [31] via a dynamical system approach. In all papers discussed above, the network size goes to infinity when the target approximation error approaches 0. However, our result in Theorem 1.1 implies that EUAF networks with a fixed size ($\mathcal{O}(d^2)$ neurons in total) can achieve an arbitrary small error for approximating $f \in C([a,b]^d)$.

The approximation errors in terms of the total number of parameters of ReLU networks are well studied for basic function spaces with (nearly) optimal approximation errors, e.g., (nearly) optimal asymptotic errors for continuous functions [48], C^s functions [50], piecewise smooth functions [39], solutions of special PDEs [3,17], functions that can be optimally approximated by affine systems [5], and Sobolev spaces [22,47]. Approximation errors in terms of width and depth would be more useful than those in terms of the total number of nonzero parameters in practice, because width and depth are two essential hyper-parameters in every numerical algorithm instead of the number of nonzero parameters. This motivated the works on the (nearly) optimal non-asymptotic errors in terms of width and depth with explicit pre-factors for approximating continuous functions in [40,43,51] and for C^s functions in [34,51]. As the errors are optimal, there are two possible directions to improve the approximation error in order to reduce the effect of the curse of dimensionality. The first one is to consider smaller target function spaces, e.g., analytic functions [6,14], Barron spaces [1,13,16,44], and band-limited functions [7,36].

Another direction is to design advanced activation functions, where one can use multiple activation functions, to enhance the power of neural networks, especially to conquer the curse of dimensionality in network approximation. There have been several papers designing activation functions to achieve good approximation errors. The results in [50] imply that $(\sin, \mathrm{ReLU})$-activated neural networks (i.e., the activation function of a neuron can be chosen from either sin or ReLU) with W parameters can approximate Lipschitz continuous functions with an asymptotic approximation error $\mathcal{O}(e^{-c_d\sqrt{W}})$, where c_d is a constant depending on d and might cause the curse of dimensionality, though the approximation error is root-exponentially small in W. In [41], it was shown that $(\mathrm{Floor}, \mathrm{ReLU})$-activated neural networks with width $\mathcal{O}(N)$ and depth $\mathcal{O}(L)$ admit an quantitative approximation error $\mathcal{O}(\sqrt{d}N^{-\sqrt{L}})$ for Lipschitz continuous functions, conquering the curse of dimensionality in approximation with a root-exponentially small error in depth L.[①] In [42], it was shown that, even if the depth is as small as 3, neural networks with width N and $\mathcal{O}(d+N)$ nonzero parameters can approximate Lipschitz continuous functions with an exponentially small error $\mathcal{O}(\sqrt{d}\,2^{-N})$, if the floor function $\lfloor x \rfloor$, the exponential function 2^x, and the step function $\mathbb{1}_{\{x\geq 0\}}$ are used as activation functions. Recently in [26], the results in [42, 50] were combined to avoid the curse of dimensionality using ReLU, sin, and 2^x activation functions. Corollary 1.2 implies that the hypothesis space of EUAF networks activated by a single activation function with $\mathcal{O}(d^2)$ neurons is dense in $C([a,b]^d)$. Particularly, all continuous functions can be arbitrarily approximated by fixed-size EUAF networks with width N and depth L on a d-dimensional hypercube, whenever $N \geq 36d(2d+1)$ and $L \geq 11$.

There is another research line for the approximation error of neural networks: apply KST [29] or its variants to explore new activation functions for a fixed-size network to achieve an arbitrary error. The original KST shows that any multivariate function $f \in C([0,1]^d)$ can be represented as $f(\boldsymbol{x}) = \sum_{i=0}^{2d} g_i\big(\sum_{j=1}^{d} h_{i,j}(x_j)\big)$ for any $\boldsymbol{x} = [x_1, \cdots, x_d]^T \in [0,1]^d$, where g_i and $h_{i,j}$ are univariate continuous functions. In fact, the composition architecture of KST can be regarded as a special neural network with (complicated) activation functions depending on the target function, which results in the failure of KST in practice. To alleviate this issue, a single activation function independent of the target function is designed in [35] to construct networks with a fixed size ($\mathcal{O}(d)$ neurons) to achieve an arbitrary error for approximating functions in $C([-1,1]^d)$. However, the activation function in [35] has no closed form and is hardly computable. See Section 2.2 for a detailed discussion of [35]. The computability issue of activation functions was addressed recently in [49]. It was shown in [49] that, for an arbitrary $\varepsilon > 0$ and any function f in $C([0,1]^d)$, there exists a network of size only depending on d constructed with multiple activation functions either (sin & arcsin) or ($\lfloor\cdot\rfloor$ & a non-polynomial analytic function) to approximate f within an error ε. To the best of our knowledge, there is no explicit characterization of the size dependence on d in [49]. For example, a very important question is whether the dependence can be mild, e.g., only a polynomial of d, or has to be severe, e.g., exponentially in d. The results of current paper provide positive answers to all the issues discussed above: we show that EUAF networks with a single simple and computable activation function, width $36d(2d+1)$, and depth 11 can approximate functions in $C([a,b]^d)$ within an arbitrary pre-specified error $\varepsilon > 0$.

In summary, the aim of this paper is to design a simple and computable activa-

① Although there is no curse of dimensionality in network approximation, the construction requires exponentially many data samples of the target function and computer memory. Hence, there would be a curse of dimensionality in inferring a target function from its finite samples when standard learning techniques are applied on a computer.

tion function σ to construct fixed-size neural networks with the universal approximation property. The network sizes of the width and depth have an explicit characterization that only depends on the dimension d. The fixed-size neural network is designed to approximate any continuous functions on a hypercube within an arbitrary error by only adjusting $\mathcal{O}(d^2)$ network parameters. Moreover, we prove that an arbitrary classification function can be exactly represented by such a fixed-size network architecture via only adjusting $\mathcal{O}(d^2)$ network parameters. The main contribution of this paper is to develop a rigorous mathematical analysis for the universal approximation property of fixed-size neural networks. The mathematical analysis developed here may be applied to understand other neural networks. The approximation results discussed here can be applied to the full error analysis of deep learning in the next subsection.

1.3 Error analysis

The error analysis of deep learning generally includes approximation, generalization, and optimization errors. Our results in this paper only deal with the approximation error. Here, we give a brief discussion on these three errors to illustrate the importance of controlling approximation errors in the applications of deep neural networks. One may find more details in [34, 41]. Let $\Phi(\boldsymbol{x}, \boldsymbol{\theta})$ denote a function in $\boldsymbol{x} \in \mathbb{R}^d$ generated by a network architecture parameterized with $\boldsymbol{\theta} \in \mathbb{R}^W$. Given a target function f, the final goal is to find the expected risk minimizer

$$\boldsymbol{\theta}_{\mathcal{D}} \in \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{D}}(\boldsymbol{\theta}), \quad \text{where } R_{\mathcal{D}}(\boldsymbol{\theta}) \coloneqq \mathbb{E}_{\boldsymbol{x} \sim U(\mathcal{X})} \left[\ell\big(\Phi(\boldsymbol{x}, \boldsymbol{\theta}), f(\boldsymbol{x})\big) \right]$$

with a loss function $\ell(\cdot, \cdot)$ and an unknown data distribution $U(\mathcal{X})$.

Theorem 1.1 implies $\inf_{\boldsymbol{\theta} \in \mathbb{R}^W} \|\Phi(\cdot, \boldsymbol{\theta}) - f(\cdot)\|_{L^\infty([a,b]^d)} = 0$ for all $f \in C([a,b]^d)$ with $\mathcal{X} = [a,b]^d$. However, $\boldsymbol{\theta}_{\mathcal{D}}$ may not be always achievable. When $\boldsymbol{\theta}_{\mathcal{D}}$ is achievable, $\mathbb{E}_{\boldsymbol{x} \sim U(\mathcal{X})} [\ell(\Phi(\boldsymbol{x}, \boldsymbol{\theta}), f(\boldsymbol{x}))] = R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{D}}) = 0$. When $\boldsymbol{\theta}_D$ is not attainable, for any pre-specified $\eta > 0$, one could identify $\boldsymbol{\theta}_{\mathcal{D},\eta} \in \mathbb{R}^W$ as the parameter set satisfying

$$R_{\mathcal{D}}\big(\boldsymbol{\theta}_{\mathcal{D},\eta}\big) \le \inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{D}}\big(\boldsymbol{\theta}\big) + \eta/2. \tag{1.6}$$

In practice, for given samples $\{(\boldsymbol{x}_i, f(\boldsymbol{x}_i))\}_{i=1}^n$, the goal of supervised learning is to identify the empirical risk minimizer

$$\boldsymbol{\theta}_{\mathcal{S}} \in \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{S}}(\boldsymbol{\theta}), \quad \text{where } R_{\mathcal{S}}(\boldsymbol{\theta}) \coloneqq \frac{1}{n} \sum_{i=1}^n \ell\big(\Phi(\boldsymbol{x}_i, \boldsymbol{\theta}), f(\boldsymbol{x}_i)\big).$$

Similarly, when $\boldsymbol{\theta}_{\mathcal{S}}$ is not attainable, our goal is to identify $\boldsymbol{\theta}_{\mathcal{S},\eta}$ instead of $\boldsymbol{\theta}_{\mathcal{S}}$ for any pre-specified $\eta > 0$, where $\boldsymbol{\theta}_{\mathcal{S},\eta} \in \mathbb{R}^W$ satisfies

$$R_{\mathcal{S}}\big(\boldsymbol{\theta}_{\mathcal{S},\eta}\big) \le \inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{S}}\big(\boldsymbol{\theta}\big) + \eta/2. \tag{1.7}$$

In practical implementation, only a numerical minimizer $\boldsymbol{\theta}_{\mathcal{N}}$ of $R_{\mathcal{S}}(\boldsymbol{\theta})$ can be achieved via a numerical optimization method. The discrepancy between the learned function

$\Phi(\boldsymbol{x}, \boldsymbol{\theta}_{\mathcal{N}})$ and the target function f is measured by $R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{N}})$, which is bounded by

$$
\begin{aligned}
R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{N}}) = & \underbrace{[R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{N}}) - R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{N}})]}_{\text{GE}} + \underbrace{[R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{N}}) - R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S},\eta})]}_{\text{OE}} + \underbrace{[R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{S},\eta}) - R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{D},\eta})]}_{\le \eta/2 \text{ by } (1.7)} + \underbrace{[R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{D},\eta}) - R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{D},\eta})]}_{\text{GE}} + \underbrace{R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{D},\eta})}_{\le \inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{D}}(\boldsymbol{\theta}) + \eta/2 \text{ by } (1.6)} \\
\le & \underbrace{\eta}_{\text{Perturbation}} + \underbrace{\inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{D}}(\boldsymbol{\theta})}_{\text{Approximation error}} + \underbrace{[R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{N}}) - \inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{S}}(\boldsymbol{\theta})]}_{\text{Optimization error (OE)}} + \underbrace{[R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{N}}) - R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{N}})] + [R_{\mathcal{S}}(\boldsymbol{\theta}_{\mathcal{D},\eta}) - R_{\mathcal{D}}(\boldsymbol{\theta}_{\mathcal{D},\eta})]}_{\text{Generalization error (GE)}}.
\end{aligned}
$$

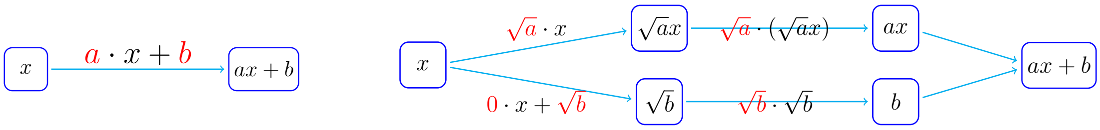
The pre-specified hyper-parameter η can be arbitrarily small and Theorem 1.1 guarantees $\inf_{\boldsymbol{\theta} \in \mathbb{R}^W} R_{\mathcal{D}}(\boldsymbol{\theta}) = 0$. Therefore, the error analysis of deep learning can be reduced to the analysis of the optimization and generalization errors, which depends on data samples, optimization algorithms, etc. One could refer to [12, 13, 15, 21, 27, 28, 32, 37, 38] for the analysis of the generalization and optimization errors.

1.4 Computability

The EUAF network is simple and computable in the sense that the output and subgradient of EUAF networks can be efficiently evaluated. The computability of EUAF implies that we can numerically implement the optimization algorithm to find a minimizer of the empirical risk. Therefore, EUAF can be directly applied to existing deep learning software in the same way as other popular activation functions (such as ReLU or sigmoid). As opposed to the computability of our EUAF, the powerful activation function proposed in [35] is not computable in the sense that there is no numerical algorithm to evaluate the output and subgradient of the corresponding network.

As we shall see later in the proof of Theorem 1.1, our EUAF network may require sufficiently large parameters to achieve an arbitrarily small error. Theorem 1.1 has provided an example of width $\mathcal{O}(d^2)$ and depth $\mathcal{O}(1)$ to achieve an arbitrarily small error. The magnitude of parameters can be dramatically reduced by increasing the network size. In particular, if we replace each elemental block like Figure 2(a) by a block like Figure 2(b), then the magnitude of parameters can be roughly reduced to its square root. By repeatedly applying this idea, it is easy to prove that the magnitude of parameters can be exponentially reduced as the network size increases linearly. If we fix the size of these larger networks and only tune their parameters, they can still approximate high-dimensional continuous functions within an arbitrarily small error. How to fix a network size to balance between the number of parameters and their memory depends on both the computer hardware and software. The goal of this paper is to demonstrate the existence of a simple network with a small and fixed size achieving an arbitrary error in spite of the magnitude of parameters and we have shown that the network size can be as small as $\mathcal{O}(d^2)$. It is interesting to investigate the balance between the network size and the memory requirement in the future.

In real applications, the parameters of the EUAF network are learned from the samples of the target function, which involves sophisticated numerical optimization. We refer to the learnability of network parameters as the existence of a numerical optimization algorithm that can identify network parameters to achieve a target approximation error. The computability of the EUAF networks does not imply learnability, which involves approximation, optimization, and generalization error analysis. The result in this paper shows that there exist computable EUAF networks achieving an arbitrarily small



(a) (b)

Figure 2: Illustrations of the magnitude reduction of parameters for a sub-network. The parameters are marked in red. Without loss of generality, $a \gg 1$ and $b \gg 1$. (a) Return $ax + b$ via two large parameters a and b. (b) Return $ax + b$ via several small parameters bounded by $\max\{\sqrt{a}, \sqrt{b}\}$.

approximation error. This means the learnability of the best approximation is reduced to achieving small generalization and optimization errors, which depends on the given data, the empirical risk model, and the optimization algorithm. Therefore, whether or not EUAF networks would be useful in real applications also depends on optimization and generalization, which is out of the scope of this paper. The optimization and generalization error analysis of practical deep neural networks including EUAF networks is a challenging problem. To the best of our knowledge, there is no complete error analysis to address the learnability of neural networks with nonlinear activation functions.

The rest of this paper is organized as follows. In Section 2, we first prove Theorem 1.1 by assuming Theorem 2.1 is true, and then prove Theorem 1.4. Next, Theorem 2.1 is proved in Section 3 based on Proposition 2.2, the proof of which can be found in Section 4. Then, several UAFs with better properties are proposed in Section 5. Finally, Section 6 concludes this paper with a short discussion.

2 Proof of main theorems

In this section, we first prove Theorem 1.1 by assuming Theorem 2.1 is true, and then prove Theorem 1.4. Notation throughout this paper are summarized in Section 2.1.

2.1 Notation

Let us summarize all basic notation used in this paper as follows.

- Let $\mathbb{R}$, $\mathbb{Q}$, and $\mathbb{Z}$ denote the set of real numbers, rational numbers, and integers, respectively.
- Let $\mathbb{N}$ and $\mathbb{N}^+$ denote the set of natural numbers and positive natural numbers, respectively. That is, $\mathbb{N}^+ = \{1, 2, 3, \cdots\}$ and $\mathbb{N} = \mathbb{N}^+ \bigcup \{0\}$.
- For any $x \in \mathbb{R}$, let $\lfloor x \rfloor := \max\{n : n \le x,\ n \in \mathbb{Z}\}$ and $\lceil x \rceil := \min\{n : n \ge x,\ n \in \mathbb{Z}\}$.
- Let $\mathbb{1}_S$ be the indicator (characteristic) function of a set S, i.e., $\mathbb{1}_S$ is equal to 1 on S and 0 outside S.
- The set difference of two sets A and B is denoted by $A \backslash B := \{x : x \in A,\ x \notin B\}$.

- Matrices are denoted by bold uppercase letters. For instance, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is a real matrix of size $m \times n$, and $\boldsymbol{A}^T$ denotes the transpose of $\boldsymbol{A}$. Vectors are denoted as bold lowercase letters. For example, $\boldsymbol{v} = [v_1, \cdots, v_d]^T = \begin{bmatrix} v_1 \\ \vdots \\ v_d \end{bmatrix} \in \mathbb{R}^d$ is a column vector. Besides, "[" and "]" are used to partition matrices (vectors) into blocks, e.g., $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} \end{bmatrix}$.

- For any $p \in [1, \infty)$, the p-norm (or ℓ^p-norm) of a vector $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in \mathbb{R}^d$ is defined by
$$\|\boldsymbol{x}\|_p = \|\boldsymbol{x}\|_{\ell^p} \coloneqq \big(|x_1|^p + |x_2|^p + \cdots + |x_d|^p\big)^{1/p}.$$
In the case $p = \infty$,
$$\|\boldsymbol{x}\|_\infty = \|\boldsymbol{x}\|_{\ell^\infty} \coloneqq \max\big\{|x_i| : i = 1, 2, \cdots, d\big\}.$$

- For any $a_1, a_2, \cdots, a_J \in \mathbb{R}$, we say $a_1, a_2, \cdots, a_J$ are **rationally independent** if they are linearly independent over the rational numbers $\mathbb{Q}$. That is, if there exist $\lambda_1, \lambda_2, \cdots, \lambda_J \in \mathbb{Q}$ such that $\sum_{j=1}^J \lambda_j \cdot a_j = 0$, then $\lambda_1 = \lambda_2 = \cdots = \lambda_J = 0$. For a simple example, 1, $\sqrt{2}$, and $\sqrt{3}$ are rationally independent.

- An **algebraic** number is any complex number (including real numbers) that is a root of a polynomial equation with rational coefficients, i.e., α is an algebraic number if and only if there exist $\lambda_0, \lambda_1, \cdots, \lambda_J \in \mathbb{Q}$ with $\sum_{j=0}^J \lambda_j \alpha^j = 0$.② Denote the set of all algebraic numbers by $\mathbb{A}$. A complex number is called **transcendental** if it is not in $\mathbb{A}$. The set $\mathbb{A}$ is countable, and, therefore, almost all numbers are transcendental. The best known transcendental numbers are π (the ratio of a circle's circumference to its diameter) and e (the natural logarithmic base).

- The expression "a network (architecture) with width N and depth L" means
  - The maximum width of this network (architecture) for all **hidden** layers is no more than N.
  - The number of **hidden** layers of this network (architecture) is no more than L.

2.2 Key ideas of proving Theorem 1.1

The proof of Theorem 1.1 has two main steps: 1) prove the one-dimensional case; 2) reduce the d-dimensional approximation to the one-dimensional case via KST [29]. In fact, in the case of $d = 1$, the size of the network in Theorem 1.1 can be further reduced as shown in Theorem 2.1 below. Theorem 2.1 is actually an enhanced version of Theorem 1.1, and, therefore, implies Theorem 1.1 in the case $d = 1$.

Theorem 2.1. *Let $f \in C([a, b])$ be a continuous function. Then, for an arbitrary $\varepsilon > 0$, there exists a function ϕ generated by an EUAF network with width 36 and depth 5 such that*
$$|\phi(x) - f(x)| < \varepsilon \quad \text{for any } x \in [a, b] \subseteq \mathbb{R}.$$

②For simplicity, we denote $1 = x^0$ for any $x \in \mathbb{R}$, including the case 0^0.

The detailed proof of Theorem 2.1 can be found in Section 3. The main ideas of proving Theorem 2.1 are developed from some ideas of our early works [41, 42]. Roughly speaking, we eventually convert a function approximation problem to a point-fitting problem via the composition architecture of neural networks in the following three steps.

- Divide $[0,1)$ into small intervals $\mathcal{I}_k = [\frac{k-1}{K}, \frac{k}{K})$ with a left endpoint x_k for $k \in \{1, 2, \cdots, K\}$, where K is an integer determined by the given error and the target function f.

- Construct a sub-network to generate a function ϕ_1 mapping the whole interval $\mathcal{I}_k$ to k for each k. The floor function $\lfloor\cdot\rfloor$ is a good choice to implement this step. Precisely, we can define $\phi_1(x) = \lfloor Kx \rfloor$. The floor function is not continuous and has zero-derivative almost everywhere. As we shall see later, σ_1 (or σ) can be a continuous alternative to implement this step, but the construction is more complicated.

- The final step is to design another sub-network to generate a function ϕ_2 mapping k approximately to $f(x_k)$ for each k. Then $\phi_2 \circ \phi_1(x) = \phi_2(k) \approx f(x_k) \approx f(x)$ for any $x \in \mathcal{I}_k$ and $k \in \{1, 2, \cdots, K\}$, which implies $\phi_2 \circ \phi_1 \approx f$ on $[0,1)$. After the above two steps, we simplify the approximation problem to a point-fitting problem, where k is approximately mapped to $f(k)$. This step is the bottleneck of the construction in our previous papers [41, 42]. Roughly speaking, the final approximation error is essentially determined by how many points we can fit using a neural network.

For the second step, the capacity to generate step functions with sufficiently many "steps" via a sub-network with a limited number of neurons plays an important role. The reproduced step functions can be considered as a continuous version of the floor function ($\lfloor\cdot\rfloor$) in [41, 42], which is a perfect step function with infinite "steps" that improves the approximation power of networks as shown in [41, 42]. The key ingredient in the third step of the proof of Theorem 2.1 is essentially a point-fitting problem with arbitrarily many points. This requires the following proposition motivated by the well-known fact that an irrational winding on the torus is dense. See Figure 3 for illustrations of such a fact. Here, we propose a new point-fitting technique that can fit arbitrarily many points within an arbitrary error using neural networks.

Proposition 2.2. *For any $K \in \mathbb{N}^+$, the following point set*

$$\Big\{ \big[\sigma_1(\tfrac{w}{\pi+1}),\ \sigma_1(\tfrac{w}{\pi+2}),\ \cdots,\ \sigma_1(\tfrac{w}{\pi+K})\big]^T : w \in \mathbb{R} \Big\} \subseteq [0,1]^K$$

is dense in $[0,1]^K$, where π is the ratio of the circumference of a circle to its diameter.

The proof of this proposition can be found in Section 4. This proposition implies that for any given sample points $(k, y_k) \in \mathbb{R}^2$ with $y_k \in [0,1]$ for $k = 1, 2, \cdots, K$ and any $K \in \mathbb{N}^+$, there exists $w_0 \in \mathbb{R}$ such that the function $x \mapsto \sigma_1(\frac{w_0}{\pi+x})$ can fit the points $(k, y_k) \in \mathbb{R}^2$ for $k = 1, 2, \cdots, K$ within an arbitrary pre-specified error $\varepsilon > 0$. To put it another way, for any $\varepsilon > 0$, there exists $w_0 \in \mathbb{R}$ such that $|\sigma_1(\frac{w_0}{\pi+k}) - y_k| < \varepsilon$ for all k.

As we shall see later in the proof of Proposition 2.2, the key point is the periodicity of the outer function σ_1. Of course, the inner function $x \mapsto \frac{w_0}{\pi+x}$ is also necessary since it

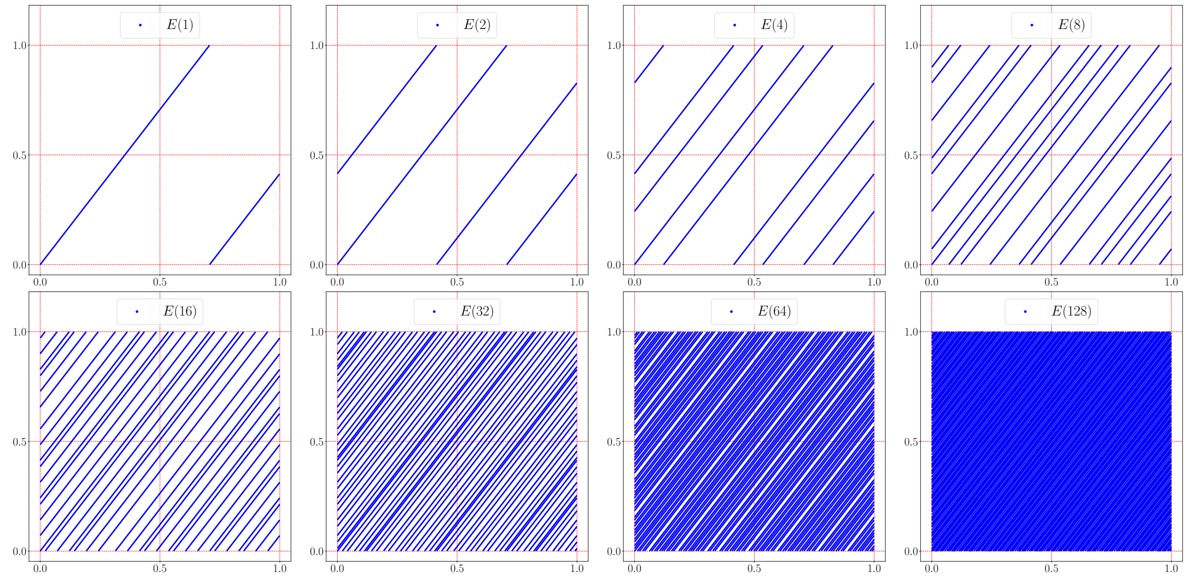


Figure 3: Illustrations of the denseness of $E(\infty)$ in $[0,1]^2$, where $E(r)$ is a winding of an "irrational" direction $[1,\sqrt{2}]^T$ on $[0,r)$, i.e., $E(r) = \{[\tau(t),\tau(\sqrt{2}t)]^T : t \in [0,r)\}$ with $\tau(t) = t - \lfloor t \rfloor$.

helps to adjust sample points for $x = 1,2,\cdots,K$. In fact, the inner function $x \mapsto \frac{w_0}{\pi+x}$ can be regarded as a variant of σ_2 via scaling and shifting. The periodicity has been explored to improve neural network approximation in the literature, e.g. the sin function in [50] is periodic and the floor function ($\lfloor\cdot\rfloor$) in [41, 42] is implicitly periodic because $x - \lfloor x \rfloor$ is periodic. Remark that a similar result holds if we replace σ_1 by a non-trivial periodic function and replace the sample locations $x = 1,2,\cdots,K$ by distinct rational numbers $r_1, r_2, \cdots, r_K \in \mathbb{Q}$. See Section 4 for a further discussion.

Theorem 2.1 essentially proves Theorem 1.1 for the univariate case. To prove the general case, we need KST [29] given below to reduce a multivariate problem to a one-dimensional case.

Theorem 2.3 (Kolmogorov superposition theorem (KST) [29])**.** *There exist continuous functions $h_{i,j} \in C([0,1])$ for $i = 0,1,\cdots,2d$ and $j = 1,2,\cdots,d$ such that any continuous function $f \in C([0,1]^d)$ can be represented as*

$$f(\boldsymbol{x}) = \sum_{i=0}^{2d} g_i\Big(\sum_{j=1}^{d} h_{i,j}(x_j)\Big) \quad \text{for any } \boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [0,1]^d,$$

where $g_i : \mathbb{R} \to \mathbb{R}$ is a continuous function for each $i \in \{0,1,\cdots,2d\}$.

KST [29] is often used to reduce a multidimensional problem to a one-dimensional one. In fact, the compositional representation in KST can be regarded as a special neural network with (complicated) activation functions depending on the target function, which makes KST useless in practical computation. To avoid this dependency, an activation function was designed in [35] to construct neural network representations with $\mathcal{O}(d)$ neurons that can approximate functions in $C([-1,1]^d)$ within an arbitrary error. Let us briefly summarize the main ideas in [35]: 1) Identify a dense and countable subset $\{u_k\}_{k=1}^{\infty}$

of $C([-1,1])$, e.g., polynomials with rational coefficients. 2) Construct an activation function ϱ to encode all $u_k(x)$ for $x \in [-1,1]$. In fact, for each k, $u_k|_{[-1,1]}$ is "stored" in ϱ on $[4k, 4k+2]$, and the values of ϱ on $[4k+2, 4k+4]$ are properly assigned to make ϱ a smooth and monotonically increasing function. That is, let $\varrho(x+4k+1) = a_k + b_k x + c_k u_k(x)$ for any $x \in [-1,1]$ with carefully chosen constants a_k, b_k, and $c_k \neq 0$ such that $\varrho(x)$ can be a sigmoid function. 3) For any $g \in C([-1,1])$, there exists a one-hidden-layer ϱ-activated network with width 3 approximating g within an arbitrary error δ, i.e., there exists k such that $g \overset{\delta}{\approx} u_k =: \frac{\varrho(x+4k+1)-a_k-b_k x}{c_k}$. 4) Replace the inner and outer functions in KST with these one-hidden-layer networks to achieve a two-hidden-layer ϱ-activated network with width $\mathcal{O}(d)$ to approximate $f \in C([0,1]^d)$ within an arbitrary error ε. As we can see, the key point of the construction in [35] is to encode a dense and countable subset of the target function space in an activation function.

We note that both [35] and this paper use KST to reduce dimension. However, the activation function of [35] is complicated without any close form and there is no efficient numerical algorithm to evaluate it. After encoding a dense subset of continuous function into a single but complicated activation function, one only needs to construct affine linear transformations to select appropriate functions of this dense subset from this complicated activation function to construct approximation. Hence, such a complicated activation function simplifies the proof of the denseness, since the denseness is encoded in the activation function. As a contrast, we design a simple activation function with efficient numerical implementation (see Figure 1 for an illustration) achieving the universal approximation property with fixed-size networks, because simple and implementable activation functions are a basic requirement for a neural network to be used in applications. However, the proof of the denseness of a neural network generated by such a simple activation function becomes difficult. A sophisticated analysis will be developed in the rest of this paper to overcome the difficulties.

We start with proving Theorem 1.1 by assuming Theorem 2.1, whose proof will be given in Section 3.

2.3 Proof of Theorem 1.1

The detailed proof of Theorem 1.1 converts the above ideas to implementations using neural networks with fixed sizes. The whole construction procedure can be divided into three steps.

(1) Apply KST to reduce dimension, i.e., represent $f \in C([a,b]^d)$ by the compositions and combinations of univariate continuous functions.

(2) Apply Theorem 2.1 to design sub-networks to approximate the univariate continuous functions in the previous step within the desired error.

(3) Integrate the sub-networks to form the final network and estimate its size.

Step 1: Apply KST to reduce dimension.

To apply KST, we define a linear function $\mathcal{L}_1(t) = (b-a)t - a$ for any $t \in [0,1]$. Clearly, $\mathcal{L}_1$ is a bijection from $[0,1]$ to $[a,b]$. Define

$$\widetilde{f}(\boldsymbol{y}) := f\big(\mathcal{L}_1(y_1), \mathcal{L}_1(y_2), \cdots, \mathcal{L}_1(y_d)\big) \quad \text{for any } \boldsymbol{y} = [y_1, y_2, \cdots, y_d]^T \in [0,1]^d.$$

Then $\widetilde{f}:[0,1]^d\to\mathbb{R}$ is a continuous function since $f\in C([a,b]^d)$. By Theorem 2.3, there exists $\widetilde{h}_{i,j}\in C([0,1])$ and $\widetilde{g}_i\in C(\mathbb{R})$ for $i=0,1,\cdots,2d$ and $j=1,2,\cdots,d$ such that

$$\widetilde{f}(\boldsymbol{y})=\sum_{i=0}^{2d}\widetilde{g}_i\Big(\sum_{j=1}^{d}\widetilde{h}_{i,j}(y_j)\Big)\quad\text{for any } \boldsymbol{y}=[y_1,y_2,\cdots,y_d]^T\in[0,1]^d.$$

Let $\widetilde{\mathcal{L}}_1$ be the inverse of $\mathcal{L}_1$, i.e., define $\widetilde{\mathcal{L}}_1(t)=(t-a)/(b-a)$ for any $t\in[a,b]$. Then, for any $x_j\in[a,b]$, there exists a unique $y_j\in[0,1]$ such that $\mathcal{L}_1(y_j)=x_j$ and $y_j=\widetilde{\mathcal{L}}_1(x_j)$ for any $j=1,2,\cdots,d$, which implies

$$\begin{aligned}f(\boldsymbol{x})&=f(x_1,x_2,\cdots,x_d)=f\big(\mathcal{L}_1(y_1),\mathcal{L}_1(y_2),\cdots,\mathcal{L}_1(y_d)\big)=\widetilde{f}(\boldsymbol{y})\\&=\sum_{i=0}^{2d}\widetilde{g}_i\Big(\sum_{j=1}^{d}\widetilde{h}_{i,j}(y_j)\Big)=\sum_{i=0}^{2d}\widetilde{g}_i\Big(\sum_{j=1}^{d}\widetilde{h}_{i,j}\big(\widetilde{\mathcal{L}}_1(x_j)\big)\Big)=\sum_{i=0}^{2d}\widetilde{g}_i\Big(\sum_{j=1}^{d}\widetilde{h}_{i,j}\circ\widetilde{\mathcal{L}}_1(x_j)\Big).\end{aligned}$$

It follows that

$$f(\boldsymbol{x})=\sum_{i=0}^{2d}\widetilde{g}_i\Big(\sum_{j=1}^{d}\widetilde{h}_{i,j}\circ\widetilde{\mathcal{L}}_1(x_j)\Big)=\sum_{i=0}^{2d}\widetilde{g}_i\circ\widehat{h}_i(\boldsymbol{x})\quad\text{for any } \boldsymbol{x}\in[a,b]^d,$$

where

$$\widehat{h}_i(\boldsymbol{x})=\sum_{j=1}^{d}\widetilde{h}_{i,j}\circ\widetilde{\mathcal{L}}_1(x_j)\quad\text{for any } \boldsymbol{x}=[x_1,x_2,\cdots,x_d]^T\in[a,b]^d.\tag{2.1}$$

Denote

$$M=\max_{i\in\{0,1,\cdots,2d\}}\|\widehat{h}_i\|_{L^\infty([a,b]^d)}+1>0.$$

Define $\mathcal{L}_2(t)=(t+2M)/4M$ and $\widetilde{\mathcal{L}}_2(t)=4Mt-2M$ for any $t\in\mathbb{R}$. Then $\mathcal{L}_2$ is a bijection from $[-M,M]$ to $[\frac14,\frac34]$ and $\widetilde{\mathcal{L}}_2$ is the inverse of $\mathcal{L}_2$. Clearly, $\widetilde{\mathcal{L}}_2\circ\mathcal{L}_2(t)=t$ for any $t\in[-M,M]$, which implies $\widehat{h}_i(\boldsymbol{x})=\widetilde{\mathcal{L}}_2\circ\mathcal{L}_2\circ\widehat{h}_i(\boldsymbol{x})$ for any $\boldsymbol{x}\in[a,b]^d$. Therefore, for any $\boldsymbol{x}\in[a,b]^d$, we have

$$f(\boldsymbol{x})=\sum_{i=0}^{2d}\widetilde{g}_i\circ\widehat{h}_i(\boldsymbol{x})=\sum_{i=0}^{2d}\widetilde{g}_i\circ\widetilde{\mathcal{L}}_2\circ\mathcal{L}_2\circ\widehat{h}_i(\boldsymbol{x})=\sum_{i=0}^{2d}g_i\circ h_i(\boldsymbol{x}),$$

where

$$g_i=\widetilde{g}_i\circ\widetilde{\mathcal{L}}_2\quad\text{and}\quad h_i=\mathcal{L}_2\circ\widehat{h}_i\quad\text{for } i=0,1,\cdots,2d.\tag{2.2}$$

Clearly, $\mathcal{L}_2(t)\in[\frac14,\frac34]$ for any $t\in[-M,M]$, which implies

$$h_i(\boldsymbol{x})=\mathcal{L}_2\circ\widehat{h}_i(\boldsymbol{x})\in[\tfrac14,\tfrac34]\quad\text{for any } \boldsymbol{x}\in[a,b]\text{ and } i=0,1,\cdots,2d.$$

Step 2: Design sub-networks to approximate g_i and h_i.

Next, we represent g_i and h_i by sub-networks. Obviously, $g_i=\widetilde{g}_i\circ\widetilde{\mathcal{L}}_2$ is continuous on $\mathbb{R}$, and, therefore, uniformly continuous on $[0,1]$ for each i. Thus, for $i=0,1,\cdots,2d$, there exists $\delta_i>0$ such that

$$|g_i(z_1)-g_i(z_2)|<\varepsilon/(4d+2)\quad\text{for any } z_1,z_2\in[0,1]\text{ with } |z_1-z_2|<\delta_i.$$

Set $\delta = \min\left(\{\delta_i : i = 0, 1, \cdots, 2d\} \cup \{\frac{1}{4}\}\right)$. Then, for $i = 0, 1, \cdots, 2d$, we have

$$|g_i(z_1) - g_i(z_2)| < \varepsilon/(4d+2) \quad \text{for any } z_1, z_2 \in [0,1] \text{ with } |z_1 - z_2| < \delta. \tag{2.3}$$

For each $i \in \{0, 1, \cdots, 2d\}$, by Theorem 2.1, there exists a function ϕ_i generated by an EUAF network with width 36 and depth 5 such that

$$|g_i(z) - \phi_i(z)| < \varepsilon/(4d+2) \quad \text{for any } z \in [0,1]. \tag{2.4}$$

Fix $i \in \{0, 1, \cdots, 2d\}$, we will design an EUAF network to generate a function $\psi_i : [a,b]^d \to \mathbb{R}$ satisfying

$$|h_i(\boldsymbol{x}) - \psi_i(\boldsymbol{x})| < \delta \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

For any $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [a,b]^d$, by Equations (2.1) and (2.2), we have

$$\begin{aligned} h_i(\boldsymbol{x}) = \mathcal{L}_2 \circ \widehat{h}_i(\boldsymbol{x}) = \mathcal{L}_2\Big(\sum_{j=1}^d \widetilde{h}_{i,j} \circ \widetilde{\mathcal{L}}_1(x_j)\Big) &= \frac{\Big(\sum_{j=1}^d \widetilde{h}_{i,j} \circ \widetilde{\mathcal{L}}_1(x_j)\Big) + 2M}{4M} \\ &= \sum_{j=1}^d \Big(\frac{\widetilde{h}_{i,j} \circ \widetilde{\mathcal{L}}_1(x_j)}{4M} + \frac{1}{2d}\Big) =: \sum_{j=1}^d h_{i,j}(x_j), \end{aligned}$$

where

$$h_{i,j}(t) \coloneqq \frac{\widetilde{h}_{i,j} \circ \widetilde{\mathcal{L}}_1(t)}{4M} + \frac{1}{2d} \quad \text{for any } t \in [a,b] \text{ and } j = 1, 2, \cdots, d.$$

For each $j \in \{1, 2, \cdots, d\}$, by Theorem 2.1, there exists a function $\psi_{i,j}$ generated by an EUAF network with width 36 and depth 5 such that

$$|h_{i,j}(t) - \psi_{i,j}(t)| < \delta/d \quad \text{for any } t \in [a,b].$$

Define $\psi_i(\boldsymbol{x}) \coloneqq \sum_{j=1}^d \psi_{i,j}(x_j)$ for any $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [a,b]^d$. Then, for any $\boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [a,b]^d$, we have

$$|h_i(\boldsymbol{x}) - \psi_i(\boldsymbol{x})| = \Big|\sum_{j=1}^d h_{i,j}(x_j) - \sum_{j=1}^d \psi_{i,j}(x_j)\Big| = \sum_{j=1}^d \Big|h_{i,j}(x_j) - \psi_{i,j}(x_j)\Big| < \sum_{j=1}^d \delta/d = \delta.$$

Step 3: Integrate sub-networks.

Finally, we build an integrated network with the desired size to approximate the target function f. The desired function ϕ can be defined as

$$\phi(\boldsymbol{x}) \coloneqq \sum_{i=0}^{2d} \phi_i \circ \psi_i(\boldsymbol{x}) = \sum_{i=0}^{2d} \phi_i\Big(\sum_{j=1}^d \psi_{i,j}(x_j)\Big) \quad \text{for any } \boldsymbol{x} = [x_1, x_2, \cdots, x_d]^T \in [a,b]^d.$$

Let us first estimate the approximation error and then determine the size of the target network realizing ϕ. See Figure 4 for an illustration of the target network realizing ϕ for the case $d = 2$.

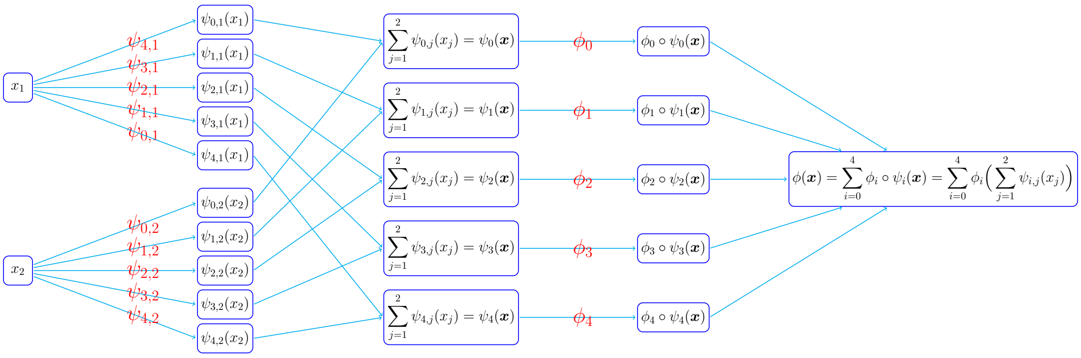


Figure 4: An illustration of the target network realizing ϕ for any $\boldsymbol{x}\in[a,b]^d$ in the case of $d=2$. This network contains $(2d+1)d+(2d+1)=(d+1)(2d+1)$ sub-networks that realize $\psi_{i,j}$ and ϕ_i for $i=0,1,\cdots,2d$ and $j=1,2,\cdots,d$.

Fix $\boldsymbol{x}\in[a,b]^d$ and $i\in\{0,1,\cdots,2d\}$. Recall that $h_i(\boldsymbol{x})\in[\frac{1}{4},\frac{3}{4}]$ and $|h_i(\boldsymbol{x})-\psi_i(\boldsymbol{x})|<\delta\le\frac{1}{4}$, which implies $\psi_i(\boldsymbol{x})\in[0,1]$. Then by Equation (2.3) (set $z_1=h_i(\boldsymbol{x})$ and $z_2=\psi_i(\boldsymbol{x})$ therein), we have

$$\Big|g_i\circ h_i(\boldsymbol{x})-g_i\circ\psi_i(\boldsymbol{x})\Big|=\Big|g_i\big(h_i(\boldsymbol{x})\big)-g_i\big(\psi_i(\boldsymbol{x})\big)\Big|<\varepsilon/(4d+2).$$

By Equation (2.4) (set $z=\psi_i(\boldsymbol{x})\in[0,1]$ therein), we have

$$\Big|g_i\circ\psi_i(\boldsymbol{x})-\phi_i\circ\psi_i(\boldsymbol{x})\Big|=\Big|g_i\big(\psi_i(\boldsymbol{x})\big)-\phi_i\big(\psi_i(\boldsymbol{x})\big)\Big|<\varepsilon/(4d+2).$$

Therefore, for any $\boldsymbol{x}\in[a,b]^d$, we have

$$\begin{aligned}\big|f(\boldsymbol{x})-\phi(\boldsymbol{x})\big|&=\Big|\sum_{i=0}^{2d}g_i\circ h_i(\boldsymbol{x})-\sum_{i=0}^{2d}\phi_i\circ\psi_i(\boldsymbol{x})\Big|=\sum_{i=0}^{2d}\Big|g_i\circ h_i(\boldsymbol{x})-\phi_i\circ\psi_i(\boldsymbol{x})\Big|\\&\le\sum_{i=0}^{2d}\bigg(\Big|g_i\circ h_i(\boldsymbol{x})-g_i\circ\psi_i(\boldsymbol{x})\Big|+\Big|g_i\circ\psi_i(\boldsymbol{x})-\phi_i\circ\psi_i(\boldsymbol{x})\Big|\bigg)\\&<\sum_{i=0}^{2d}\Big(\varepsilon/(4d+2)+\varepsilon/(4d+2)\Big)=\varepsilon.\end{aligned}$$

It remains to show ϕ can be generated by an EUAF network with the desired size. Recall that, for each $i\in\{0,1,\cdots,2d\}$ and each $j\in\{1,2,\cdots,d\}$, $\psi_{i,j}$ can be generated by an EUAF network with width 36, depth 5, and, therefore, at most

$$(36+36)+(36\times36+36)\times4+(36+1)=5437$$

nonzero parameters. Hence, for each $i\in\{0,1,\cdots,2d\}$, ψ_i, given by $\psi_i(\boldsymbol{x})=\sum_{j=1}^{d}\psi_{i,j}(x_j)$, can be generated by an EUAF network with width $36d$, depth 5, and at most $5437d$ nonzero parameters.

Since $\psi_i(\boldsymbol{x})\in[0,1]$ for any $\boldsymbol{x}\in[a,b]^d$ and $i=0,1,\cdots,2d$, we have $\sigma\big(\psi_i(\boldsymbol{x})\big)=\psi_i(\boldsymbol{x})$ for any $\boldsymbol{x}\in[a,b]^d$. Hence, $\phi_i\circ\psi_i$ can be generated by an EUAF network as visualized in Figure 5.

Recall that ϕ_i can be generated by an EUAF network with width 36 and depth 5. Hence, the network generating ϕ_i has at most 5437 nonzero parameters. As we can see

$$\boxed{\boldsymbol{x}} \xrightarrow{\ \psi_i\ } \boxed{\sigma\big(\psi_i(\boldsymbol{x})\big) = \psi_i(\boldsymbol{x})} \xrightarrow{\ \phi_i\ } \boxed{\phi_i\big(\psi_i(\boldsymbol{x})\big) = \phi_i \circ \psi_i(\boldsymbol{x})}$$

Figure 5: An illustration of the target EUAF network generating $\phi_i \circ \psi_i(\boldsymbol{x})$ for any $\boldsymbol{x} \in [a,b]^d$ and $i = 0, 1, \cdots, 2d$.

from Figure 5, $\phi_i \circ \psi_i$ can be generated by an EUAF network with width $36d$, depth $5 + 1 + 5 = 11$, and at most $5437d + 5437 = 5437(d+1)$ nonzero parameters. This means $\phi = \sum_{i=0}^{2d} \phi_i \circ \psi_i$ can be generated by an EUAF network with width $36d(2d+1)$, depth 11, and, therefore, at most $5437(d+1)(2d+1)$ nonzero parameters as desired. So we finish the proof.

2.4 Proof of Theorem 1.4

The proof of Theorem 1.4 relies on a basic result of real analysis given in the following lemma.

Lemma 2.4. *Suppose $A, B \subseteq \mathbb{R}^d$ are two disjoint bounded closed sets. Then there exists a continuous function $f \in C(\mathbb{R}^d)$ such that $f(\boldsymbol{x}) = 1$ for any $\boldsymbol{x} \in A$ and $f(\boldsymbol{y}) = 0$ for any $\boldsymbol{y} \in B$.*

Proof. Define $\mathrm{dist}(\boldsymbol{x}, A) = \inf\{\|\boldsymbol{x} - \boldsymbol{y}\|_2 : \boldsymbol{y} \in A\}$ and $\mathrm{dist}(\boldsymbol{x}, B) = \inf\{\|\boldsymbol{x} - \boldsymbol{y}\|_2 : \boldsymbol{y} \in B\}$ for any $\boldsymbol{x} \in \mathbb{R}^d$. It is easy to verify that $\mathrm{dist}(\boldsymbol{x}, A)$ and $\mathrm{dist}(\boldsymbol{x}, B)$ are continuous in $\boldsymbol{x} \in \mathbb{R}^d$. Since $A, B \in \mathbb{R}^d$ are two disjoint bounded closed subsets, we have $\mathrm{dist}(\boldsymbol{x}, A) + \mathrm{dist}(\boldsymbol{x}, B) > 0$ for any $\boldsymbol{x} \in \mathbb{R}^d$. Finally, define

$$f(\boldsymbol{x}) \coloneqq \frac{\mathrm{dist}(\boldsymbol{x}, B)}{\mathrm{dist}(\boldsymbol{x}, A) + \mathrm{dist}(\boldsymbol{x}, B)} \quad \text{for any } \boldsymbol{x} \in \mathbb{R}^d.$$

Then f meets the requirements. So we finish the proof. □

With Lemma 2.4, we can prove Theorem 1.4.

Proof of Theorem 1.4. For any $f = \sum_{j=1}^J r_j \cdot \mathbb{1}_{E_j} \in \mathscr{C}_d(E_1, E_2, \cdots, E_J)$, our goal is to construct a function ϕ generated by a σ-activated network such that $\phi(\boldsymbol{x}) = f(\boldsymbol{x})$ for any $\boldsymbol{x} \in \bigcup_{j=1}^J E_j$, where $E_1, E_2, \cdots, E_J$ are pairwise disjoint bounded closed subsets of $\mathbb{R}^d$. Set $E \coloneqq \bigcup_{j=1}^J E_j$ and choose $a, b \in \mathbb{R}$ properly such that $E \subseteq [a,b]^d$.

For each $j \in \{1, 2, \cdots, J\}$, E_j and $\widetilde{E}_j \coloneqq E \backslash E_j$ are two disjoint bounded closed subsets. Then, for each j, by Lemma 2.4, there exists $g_j \in C(\mathbb{R}^d)$ such that $g_j(\boldsymbol{x}) = 1$ for any $\boldsymbol{x} \in E_j$ and $g_j(\boldsymbol{y}) = 0$ for any $\boldsymbol{y} \in \widetilde{E}_j$. By defining $g \coloneqq \sum_{j=1}^J r_j \cdot g_j \in C(\mathbb{R}^d)$, we have $g(\boldsymbol{x}) = \sum_{j=1}^J r_j \cdot \mathbb{1}_{E_j}(\boldsymbol{x}) = f(\boldsymbol{x})$ for any $\boldsymbol{x} \in E = \bigcup_{j=1}^J E_j$.

Since $r_1, r_2, \cdots, r_J$ are rational numbers and $g : [a,b]^d \to \mathbb{R}$ is continuous, there exist $n_1, n_2 \in \mathbb{Z}$ such that

- $n_1 \cdot r_j + n_2 \in \mathbb{N}^+$ for $j = 1, 2, \cdots, J$;
- $n_1 \cdot g(\boldsymbol{x}) + n_2 \geq 0$ for any $\boldsymbol{x} \in [a,b]^d$.

By applying Theorem 1.1 to $2(n_1 \cdot g + n_2) + 1$, there exists a function ϕ_1 generated by an EUAF network with width $36d(2d+1)$, depth 11, and at most $5437(d+1)(2d+1)$ nonzero parameters such that

$$\left|2(n_1 \cdot g(\boldsymbol{x}) + n_2) + 1 - \phi_1(\boldsymbol{x})\right| \le 1/2 \quad \text{for any } \boldsymbol{x} \in [a,b]^d. \tag{2.5}$$

It follows that

$$\left|2\Big(n_1 \cdot \sum_{j=1}^{J} r_j \cdot \mathbb{1}_{E_j}(\boldsymbol{x}) + n_2\Big) + 1 - \phi_1(\boldsymbol{x})\right| \le 1/2 \quad \text{for any } \boldsymbol{x} \in E = \bigcup_{j=1}^{J} E_j.$$

Since $E_1, E_2, \cdots, E_J$ are pairwise disjoint, we have

$$\left|2(n_1 \cdot r_j + n_2) + 1 - \phi_1(\boldsymbol{x})\right| \le 1/2 \quad \text{for any } \boldsymbol{x} \in E_j \text{ and each } j \in \{1, 2, \cdots, J\}. \tag{2.6}$$

Define $\phi_2(x) = x + 1/2 - \sigma(x + 3/2)$ for any $x \in \mathbb{R}$. See Figure 6 for an illustration. It is

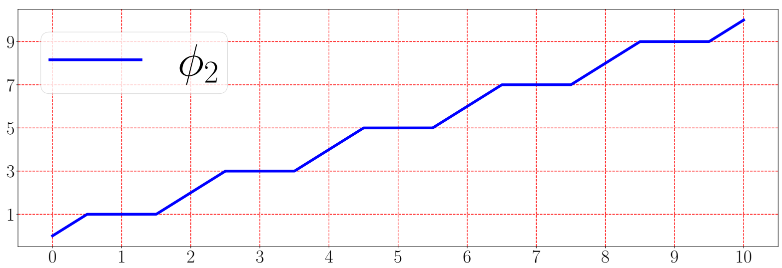


Figure 6: An illustration of ϕ_2 on $[0, 10]$.

easy to verify that

$$\phi_2(y) = 2k + 1 \qquad \text{for any } y \text{ and } k \in \mathbb{N}^+ \text{ with } |2k + 1 - y| \le 1/2. \tag{2.7}$$

Therefore, by Equations (2.6) and (2.7) (set $y = \phi_1(\boldsymbol{x})$ and $k = n_1 \cdot r_j + n_2$ therein), we have $\phi_2 \circ \phi_1(\boldsymbol{x}) = 2(n_1 \cdot r_j + n_2) + 1$ for any $\boldsymbol{x} \in E_j$ and any $j \in \{1, 2, \cdots, J\}$, which implies

$$\frac{\phi_2 \circ \phi_1(\boldsymbol{x}) - 2n_2 - 1}{2n_1} = r_j \quad \text{for any } \boldsymbol{x} \in E_j \text{ and any } j \in \{1, 2, \cdots, J\}.$$

Define

$$\phi(\boldsymbol{x}) \coloneqq \frac{\phi_2 \circ \phi_1(\boldsymbol{x}) - 2n_2 - 1}{2n_1} \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Clearly, we have $\phi(\boldsymbol{x}) = r_j$ for any $\boldsymbol{x} \in E_j$ and each $j \in \{1, 2, \cdots, J\}$, which implies $\phi(\boldsymbol{x}) = \sum_{j=1}^{J} r_j \cdot \mathbb{1}_{E_j}(\boldsymbol{x}) = f(\boldsymbol{x})$ for any $\boldsymbol{x} \in E = \bigcup_{j=1}^{J} E_j$ as desired.

Set $M = 2\|n_1 g + n_2\|_{L^\infty([a,b]^d)} + 3/2 > 0$. By Equation (2.5) and the fact $n_1 \cdot g(\boldsymbol{x}) + n_2 \ge 0$ for any $\boldsymbol{x} \in [a,b]^d$, we have

$$\phi_1(\boldsymbol{x}) \in \left[1/2,\ 2\|n_1 g + n_2\|_{L^\infty([a,b]^d)} + 1 + 1/2\right] \subseteq [0, M] \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Then, for any $\boldsymbol{x} \in [a,b]^d$, we have

$$\phi_2 \circ \phi_1(\boldsymbol{x}) = \phi_1(\boldsymbol{x}) + 1/2 - \sigma\big(\phi_1(\boldsymbol{x}) + 3/2\big) = M\sigma\Big(\phi_1(\boldsymbol{x})/M\Big) + 1/2 - \sigma\big(\phi_1(\boldsymbol{x}) + 3/2\big).$$

It follows that

$$\phi(\boldsymbol{x}) = \frac{\phi_2 \circ \phi_1(\boldsymbol{x}) - 2n_2 - 1}{2n_1} = \frac{M\sigma\Big(\phi_1(\boldsymbol{x})/M\Big) - \sigma\Big(\phi_1(\boldsymbol{x}) + 3/2\Big) - 2n_2 - 1/2}{2n_1},$$

for any $\boldsymbol{x} \in [a,b]^d$. The network realizing ϕ has just one more hidden layer with 2 neurons, compared to the network realizing ϕ_1. Recall that ϕ_1 can be generated by an EUAF network with width $36d(2d+1)$, depth 11, and at most $5437(d+1)(2d+1)$ nonzero parameters. Therefore, ϕ, limited on $[a,b]^d$, can be generated by an EUAF network with width $36d(2d+1)$, depth 12, and at most

$$5437(d+1)(2d+1) \;+\; \underbrace{36d(2d+1)\times 2 + 2 \;+\; 2+1}_{\text{all possible new parameters}} \le 5509(d+1)(2d+1)$$

nonzero parameters. So we finish the proof. □

3 Proof of Theorem 2.1

To prove Theorem 2.1, we need to introduce two auxiliary theorems, Theorems 3.1 and 3.2, which serve as two important intermediate steps.

Theorem 3.1. *Let $f \in C([0,1])$ be a continuous function. Given any $\varepsilon > 0$, if K is a positive integer satisfying*

$$|f(x_1) - f(x_2)| < \varepsilon/2 \quad \text{for any } x_1, x_2 \in [0,1] \text{ with } |x_1 - x_2| < 1/K, \tag{3.1}$$

then there exists a function ϕ generated by an EUAF network with width 2 and depth 3 such that $\|\phi\|_{L^\infty([0,1])} \le \|f\|_{L^\infty([0,1])} + 1$ and

$$|\phi(x) - f(x)| < \varepsilon \quad \text{for any } x \in \bigcup_{k=0}^{K-1} \Big[\frac{2k}{2K}, \frac{2k+1}{2K}\Big].$$

Theorem 3.2. *Let $f \in C([0,1])$ be a continuous function. Then, for an arbitrary $\varepsilon > 0$, there exists a function ϕ generated by an EUAF network with width 36 and depth 5 such that*[③]

$$|\phi(x) - f(x)| < \varepsilon \quad \text{for any } x \in [0, \tfrac{9}{10}].$$

To prove Theorem 3.1, we only need to care about the approximation on "half" of $[0,1]$. It is not necessary to care about the approximation on the other "half" of $[0,1]$. Such an idea is similar to the "trifling region" in [34, 51]. As we shall see later, the proof of Theorem 3.1 can eventually be converted to a point-fitting problem, which can be solved by applying Proposition 2.2.

The key idea to prove Theorem 3.2 is to apply Theorem 3.1 to several horizontally shifted variants of the target function. Then a good approximation can be constructed via the combinations and multiplications of these variants, similar to the idea of [34, 51]

③Theorem 3.2 still holds via replacing $\frac{9}{10}$ by any number in $[0,1)$. In fact, it is true for $[0, \frac{1}{K}]$, and K can be arbitrarily large.

to obtain an error estimation with the L^∞-norm from a result with the L^p-norm for $p \in [1, \infty)$.

The proofs of Theorems 3.1 and 3.2 will be presented in Sections 3.1 and 3.2, respectively. Let us first prove Theorem 2.1 by assuming Theorem 3.2 is true.

Proof of Theorem 2.1. Define a linear function $\mathcal{L}$ by $\mathcal{L}(x) = a + \frac{10(b-a)}{9} x$ for any $x \in [0, \frac{9}{10}]$. Then $\mathcal{L}$ is a bijection from $[0, \frac{9}{10}]$ to $[a, b]$. It follows that $f \circ \mathcal{L}$ is a continuous function on $[0, \frac{9}{10}]$. By Theorem 3.2, there exists a function $\widetilde{\phi}$ generated by an EUAF network with width 36 and depth 5 such that

$$|f \circ \mathcal{L}(x) - \widetilde{\phi}(x)| < \varepsilon \quad \text{for any } x \in [0, \tfrac{9}{10}].$$

Define $\widetilde{\mathcal{L}}(y) = \frac{9(y-a)}{10(b-a)}$ for any $y \in [a, b]$. Clearly, it is the inverse of $\mathcal{L}$, i.e., $\mathcal{L} \circ \widetilde{\mathcal{L}}(y) = y$ for any $y \in [a, b]$. Therefore, for any $y \in [a, b]$, we have $x = \widetilde{\mathcal{L}}(y) \in [0, \frac{9}{10}]$, which implies

$$\begin{aligned} |f(y) - \widetilde{\phi} \circ \widetilde{\mathcal{L}}(y)| &= \left| f \circ \mathcal{L} \circ \widetilde{\mathcal{L}}(y) - \widetilde{\phi} \circ \widetilde{\mathcal{L}}(y) \right| \\ &= \left| f \circ \mathcal{L}\big(\widetilde{\mathcal{L}}(y)\big) - \widetilde{\phi}\big(\widetilde{\mathcal{L}}(y)\big) \right| \le |f \circ \mathcal{L}(x) - \widetilde{\phi}(x)| < \varepsilon. \end{aligned}$$

By defining $\phi \coloneqq \widetilde{\phi} \circ \widetilde{\mathcal{L}}$, we have $|f(y) - \phi(y)| < \varepsilon$ for any $y \in [a, b]$ as desired. Note that $\widetilde{\phi}$ can be realized by an EUAF network with width 36 and depth 5. We can compose $\widetilde{\mathcal{L}}$ and the affine linear map of the network $\widetilde{\phi}$ that connects the input layer and the first hidden layer. Therefore, $\phi = \widetilde{\phi} \circ \widetilde{\mathcal{L}}$ can also be realized by an EUAF network with width 36 and depth 5. So we finish the proof. □

3.1 Proof of Theorem 3.1

Partition $[0, 1]$ into $2K$ small intervals $\mathcal{I}_k$ and $\widetilde{\mathcal{I}}_k$ for $k = 1, 2, \cdots, K$, i.e.,

$$\mathcal{I}_k = \Big[\frac{2k-2}{2K}, \frac{2k-1}{2K}\Big] \quad \text{and} \quad \widetilde{\mathcal{I}}_k = \Big[\frac{2k-1}{2K}, \frac{2k}{2K}\Big].$$

Clearly, $[0, 1] = \bigcup_{k=1}^K (\mathcal{I}_k \cup \widetilde{\mathcal{I}}_k)$. Let x_k be the right endpoint of $\mathcal{I}_k$, i.e., $x_k = \frac{2k-1}{2K}$ for $k = 1, 2, \cdots, K$. See an illustration of $\mathcal{I}_k$, $\widetilde{\mathcal{I}}_k$, and x_k in Figure 7 for the case $K = 5$.

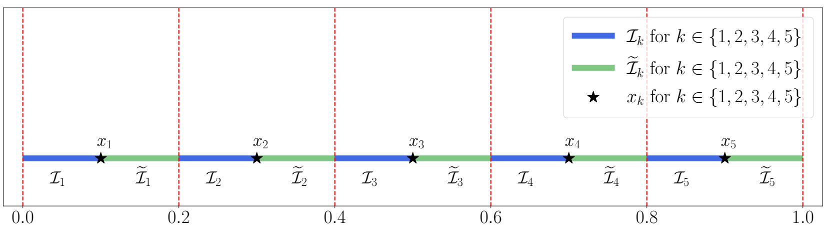


Figure 7: An illustration of $\mathcal{I}_k$ and $\widetilde{\mathcal{I}}_k$ for $k \in \{1, 2, \cdots, K\}$ with $K = 5$.

Our goal is to construct a function ϕ generated by an EUAF network with the desired size to approximate f well on $\mathcal{I}_k$ for $k = 1, 2, \cdots, K$. It is not necessary to care about the values of ϕ on $\widetilde{\mathcal{I}}_k$ for all k. In other words, we only need to care about the approximation on a "half" of $[0, 1]$, which is the key for our proof.

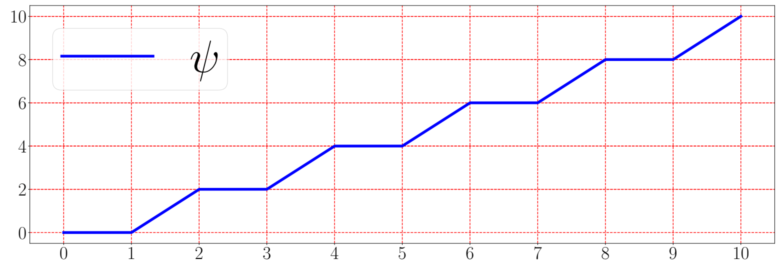


Figure 8: An illustration of ψ on $[0, 10]$.

Define $\psi(x) = x - \sigma(x)$ for any $x \in \mathbb{R}$, where σ is defined in Equation (1.3). See Figure 8 for an illustration of ψ.

It is easy to verify that

$$\psi(y) = 2k - 2 \quad \text{for any } y \in [2k-2, 2k-1] \text{ and each } k \in \{1, 2, \cdots, K\}.$$

It follows that

$$\psi(2Kx)/2 + 1 = k \quad \text{for any } x \in [\tfrac{2k-2}{2K}, \tfrac{2k-1}{2K}] = \mathcal{I}_k \text{ and each } k \in \{1, 2, \cdots, K\}.$$

Recall that x_k is the right endpoint of $\mathcal{I}_k$ for $k = 1, 2, \cdots, K$. Set $M = \|f\|_{L^\infty([0,1])} + 1$ and define

$$\xi_k \coloneqq \tfrac{f(x_k)+M}{2M} \in [0, 1] \quad \text{for } k = 1, 2, \cdots, K.$$

Then $[\xi_1, \xi_2, \cdots, \xi_K]^T$ is in $[0, 1]^K$. By Proposition 2.2, there exists $w_0 \in \mathbb{R}$ such that

$$\Big|\sigma_1(\tfrac{w_0}{\pi+k}) - \xi_k\Big| < \varepsilon/(4M) \quad \text{for } k = 1, 2, \cdots, K.$$

Let m_0 be an integer larger than $|w_0|$, e.g., set $m_0 = \lfloor |w_0| \rfloor + 1$. It is easy to verify that

$$\tfrac{w_0}{\pi+k} + 2m_0 \geq 0 \quad \text{for any } x \in [0, 1].$$

Since $\sigma(x) = \sigma_1(x)$ for $x \geq 0$ and σ_1 is periodic with period 2, we have

$$\Big|\sigma(\tfrac{w_0}{\pi+k} + 2m_0) - \xi_k\Big| = \Big|\sigma_1(\tfrac{w_0}{\pi+k} + 2m_0) - \xi_k\Big| = \Big|\sigma_1(\tfrac{w_0}{\pi+k}) - \xi_k\Big| < \varepsilon/(4M),$$

for $k = 1, 2, \cdots, K$. It follows that

$$\begin{aligned} \Big|2M\sigma(\tfrac{w_0}{\pi+k} + 2m_0) - M - f(x_k)\Big| &= \Big|2M\sigma(\tfrac{w_0}{\pi+k} + 2m_0) - M - (2M\xi_k - M)\Big| \\ &= 2M\Big|\sigma(\tfrac{w_0}{\pi+k} + 2m_0) - \xi_k\Big| < 2M\frac{\varepsilon}{4M} = \varepsilon/2, \end{aligned} \tag{3.2}$$

for $k = 1, 2, \cdots, K$.

The desired ϕ is defined as

$$\phi(x) \coloneqq 2M\sigma\Big(\tfrac{w_0}{\pi+\psi(2Kx)/2+1} + 2m_0\Big) - M \quad \text{for any } x \in [0, 1].$$

Recall that $m_0 \geq |w_0|$ and $\psi(x) \geq 0$ for any $x \geq 0$, which implies $\tfrac{w_0}{\pi+\psi(2Kx)/2+1} + 2m_0 \geq 0$ for any $x \in [0, 1]$. Thus, $\|\phi\|_{L^\infty([0,1])} \leq M = \|f\|_{L^\infty([0,1])} + 1$ since $0 \leq \sigma(y) \leq 1$ for any $y \geq 0$.

For any $x \in \mathcal{I}_k$ and each $k \in \{1, 2, \cdots, K\}$, we have $\psi(2Kx)/2 + 1 = k$, which implies

$$\phi(x) = 2M\sigma\Big(\tfrac{w_0}{\pi+\psi(2Kx)/2+1} + 2m_0\Big) - M = 2M\sigma\Big(\tfrac{w_0}{\pi+k} + 2m_0\Big) - M.$$

For any $x \in \mathcal{I}_k$ and each $k \in \{1, 2, \cdots, K\}$, we have $|x_k - x| < 1/K$, which implies $|f(x_k) - f(x)| < \varepsilon/2$ by Equation (3.1). Therefore, by Equation (3.2), we have

$$\begin{aligned} |\phi(x) - f(x)| &= \Big|2M\sigma\Big(\tfrac{w_0}{\pi+k} + 2m_0\Big) - M - f(x)\Big| \\ &\le \Big|2M\sigma\Big(\tfrac{w_0}{\pi+k} + 2m_0\Big) - M - f(x_k)\Big| + |f(x_k) - f(x)| < \varepsilon/2 + \varepsilon/2 = \varepsilon, \end{aligned}$$

for any $x \in \mathcal{I}_k$ and each $k \in \{1, 2, \cdots, K\}$. It follows that

$$|\phi(x) - f(x)| < \varepsilon \quad \text{for any } x \in \bigcup_{j=1}^{K} \mathcal{I}_j = \bigcup_{j=1}^{K} \Big[\frac{2j-2}{2K}, \frac{2j-1}{2K}\Big] = \bigcup_{k=0}^{K-1} \Big[\frac{2k}{2K}, \frac{2k+1}{2K}\Big].$$

It remains to show that ϕ can be generated by an EUAF network with the desired size. Observe that

$$\sigma(y) + 1 = \frac{y}{|y| + 1} + 1 = \frac{y}{-y + 1} + 1 = \frac{1}{-y + 1} \quad \text{for any } y \le 0.$$

By setting $y = -\pi - \psi(2Kx)/2 \le 0$ for any $x \in [0, 1]$, we have

$$\begin{aligned} \tfrac{1}{\pi+\psi(2Kx)/2+1} = \tfrac{1}{-y+1} = \sigma(y) + 1 &= \sigma\Big(-\pi - \psi(2Kx)/2\Big) + 1 \\ &= \sigma\Big(-\pi - \big(2Kx - \sigma(2Kx)\big)/2\Big) + 1 \\ &= \sigma\Big(-\pi - Kx + \sigma(2Kx)/2\Big) + 1, \end{aligned}$$

where the large equality comes from $\psi(z) = z - \sigma(z)$ for any $z \in \mathbb{R}$. Therefore, we get

$$\begin{aligned} \phi(x) &= 2M\sigma\Big(\tfrac{w_0}{\pi+\psi(2Kx)/2+1} + 2m_0\Big) - M \\ &= 2M\sigma\bigg(w_0\sigma\Big(-\pi - Kx + \sigma(2Kx)/2\Big) + w_0 + 2m_0\bigg) - M. \end{aligned} \tag{3.3}$$

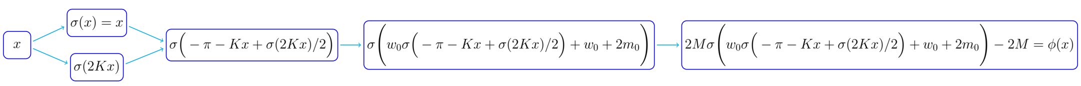


Figure 9: An illustration of the target EUAF network realizing $\phi(x)$ for $x \in [0, 1]$ based on Equation (3.3).

Thus, the desired EUAF network realizing ϕ is shown in Figure 9. Clearly, the network in Figure 9 has width 2 and depth 3 as desired. It is easy to verify that the network architecture of ϕ is independent of the target function f and the desired error ε. That is, we can fix the architecture and only adjust parameters to achieve the desired approximation error. So we finish the proof.

3.2 Proof of Theorem 3.2

The key idea of proving Theorem 3.2 is to apply Theorem 3.1 to several horizontally shifted variants of the target function. Then a good approximation can be expected via combinations and multiplications of these variants. Thus, we need to reproduce $f(x,y) = xy$ locally via an EUAF network as shown in the following lemma.

Lemma 3.3. *For any $M > 0$, there exists a function ϕ generated by an EUAF network with width 9 and depth 2 such that*

$$\phi(x,y) = xy \quad \text{for any } x,y \in [-M,M].$$

The proof of this lemma is given in Section 3.3. Now let us first prove Theorem 3.2 by assuming this lemma is true.

Proof of Theorem 3.2. Set $\widetilde{\varepsilon} = \varepsilon/4$ and extend f from $[0,1]$ to $[-1,1]$ by defining $f(x) = f(0)$ for $x \in [-1,0)$. Then f is continuous on $[-1,1]$, and, therefore, uniformly continuous. Thus, there exists $K = K(f,\varepsilon) \in \mathbb{N}^+$ with $K \ge 10$ such that

$$|f(x_1) - f(x_2)| < \widetilde{\varepsilon}/2 \quad \text{for any } x_1, x_2 \in [-1,1] \text{ with } |x_1 - x_2| < 1/K.$$

For $i = 1,2,3,4$, define

$$f_i(x) \coloneqq f\big(x - \tfrac{i}{4K}\big) \quad \text{for any } x \in [0,1].$$

For each $i \in \{1,2,3,4\}$ and any $x_1, x_2 \in [0,1]$ with $|x_1 - x_2| < 1/K$, we have $x_1 - \frac{i}{4K}, x_2 - \frac{i}{4K} \in [-1,1]$ and $\big|(x_1 - \frac{i}{4K}) - (x_2 - \frac{i}{4K})\big| = |x_1 - x_2| < 1/K$, which implies

$$|f_i(x_1) - f_i(x_2)| = |f(x_1 - \tfrac{i}{4K}) - f(x_2 - \tfrac{i}{4K})| < \widetilde{\varepsilon}/2.$$

That is, for $i = 1,2,3,4$, we have

$$|f_i(x_1) - f_i(x_2)| < \widetilde{\varepsilon}/2 \quad \text{for any } x_1, x_2 \in [0,1] \text{ with } |x_1 - x_2| < 1/K.$$

For each $i \in \{1,2,3,4\}$, by Theorem 3.1, there exist a function ϕ_i generated by an EUAF network with width 2 and depth 3 such that $\|\phi_i\|_{L^\infty([0,1])} \le \|f_i\|_{L^\infty([0,1])} + 1 \le \|f\|_{L^\infty([-1,1])} + 1$ and

$$\big|\phi_i(x) - f_i(x)\big| < \widetilde{\varepsilon} = \varepsilon/4 \quad \text{for any } x \in \bigcup_{k=0}^{K-1} \Big[\frac{2k}{2K}, \frac{2k+1}{2K}\Big].$$

Define

$$\psi(x) = \sigma\big(x + 1 - \sigma(x+1)\big) \quad \text{for any } x \in \mathbb{R}.$$

See an illustration of ψ on $[0, 2K]$ for $K = 5$ in Figure 10.

Clearly, $0 \le \psi(2Kx) \le 1$ for any $x \in [0,1]$, which results in

$$\Big|\big(\phi_i(x) - f_i(x)\big)\psi(2Kx)\Big| \le \big|\phi_i(x) - f_i(x)\big| < \varepsilon/4 \quad \text{for any } x \in \bigcup_{k=0}^{K-1} \Big[\frac{2k}{2K}, \frac{2k+1}{2K}\Big].$$

Observe that $\psi(y) = 0$ for $y \in \bigcup_{k=0}^{K-1}[2k+1, 2k+2]$, which implies

$$\psi(2Kx) = 0 \quad \text{for any } x \in \bigcup_{k=0}^{K-1}\Big[\frac{2k+1}{2K}, \frac{2k+2}{2K}\Big] \supseteq [0,1] \Big\backslash \bigcup_{k=0}^{K-1} \Big[\frac{2k}{2K}, \frac{2k+1}{2K}\Big].$$

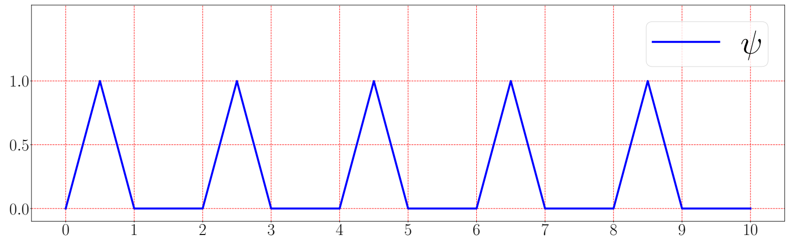


Figure 10: An illustration of ψ on $[0, 2K]$ for $K = 5$.

It follows that

$$\left|\big(\phi_i(x) - f_i(x)\big)\psi(2Kx)\right| < \varepsilon/4 \quad \text{for any } x \in [0,1] \text{ and } i = 1,2,3,4. \tag{3.4}$$

For each $i \in \{1,2,3,4\}$ and any $z \in [0, \frac{9}{10}] \subseteq [0, 1 - \frac{i}{4K}]$, we have $y_i = z + \frac{i}{4K} \in [\frac{i}{4K}, 1] \subseteq [0,1]$. Therefore, by bringing $y_i \in [0,1]$ into Equation (3.4) (set $x = y_i$ therein), we have

$$\begin{aligned} \varepsilon/4 &> \left|\big(\phi_i(y_i) - f_i(y_i)\big)\psi(2Ky_i)\right| = \left|\phi_i(y_i)\psi(2Ky_i) - f_i(y_i)\psi(2Ky_i)\right| \\ &= \left|\phi_i(z + \tfrac{i}{4K})\psi\big(2K(z + \tfrac{i}{4K})\big) - f_i(z + \tfrac{i}{4K})\psi\big(2K(z + \tfrac{i}{4K})\big)\right| \\ &= \left|\phi_i(z + \tfrac{i}{4K})\psi\big(2Kz + \tfrac{i}{2}\big) - f(z)\psi\big(2Kz + \tfrac{i}{2}\big)\right|, \end{aligned} \tag{3.5}$$

where the last equality comes from the fact that $f_i(x) = f(x - \frac{i}{4K})$ for any $x \in [0,1] \supseteq [\frac{i}{4K}, 1]$. The desired ϕ is defined as

$$\phi(x) \coloneqq \sum_{i=1}^{4} \phi_i(x + \tfrac{i}{4K})\psi\big(2Kx + \tfrac{i}{2}\big) \quad \text{for any } x \in [0, \tfrac{9}{10}].$$

It is easy to verify that $\sum_{i=1}^4 \psi\big(x + \frac{i}{2}\big) = 1$ for any $x \geq 0$ based on the definition of ψ. See Figure 11 for illustrations. It follows that $\sum_{i=1}^4 \psi\big(2Kz + \frac{i}{2}\big) = 1$ for any $z \in [0, \frac{9}{10}]$.

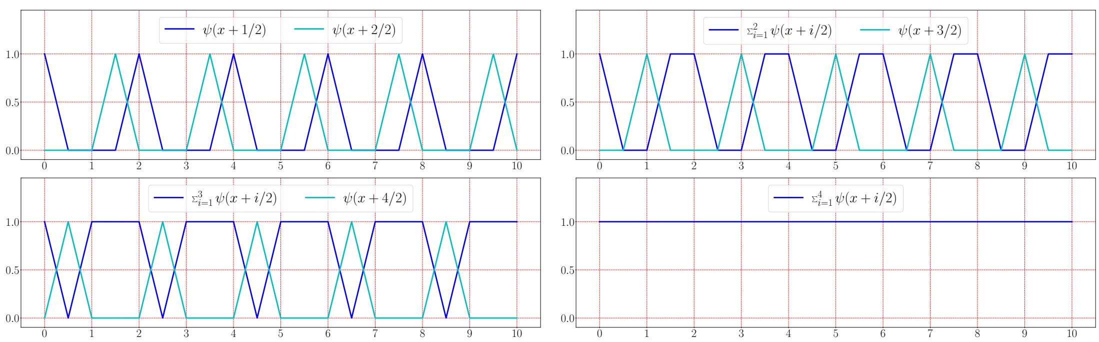


Figure 11: Illustrations of $\sum_{i=1}^4 \psi(x + i/2) = 1$ for any $x \in [0, 10]$.

Hence, by Equation (3.5), we have

$$\begin{aligned} \left|\phi(z) - f(z)\right| &= \left|\sum_{i=1}^{4} \phi_i(z + \tfrac{i}{4K})\psi\big(2Kz + \tfrac{i}{2}\big) - f(z)\sum_{i=1}^{4}\psi\big(2Kz + \tfrac{i}{2}\big)\right| \\ &\leq \sum_{i=1}^{4}\left|\phi_i(z + \tfrac{i}{4K})\psi\big(2Kz + \tfrac{i}{2}\big) - f(z)\psi\big(2Kz + \tfrac{i}{2}\big)\right| < 4\frac{\varepsilon}{4} = \varepsilon. \end{aligned}$$

That is, $|\phi(x)-f(x)|<\varepsilon$ for any $x\in[0,\frac{9}{10}]$ as desired. It remains to show that ϕ, limited on $[0,\frac{9}{10}]$, can be generated by an EUAF network with the desired size.

Note that $x+1=(2K+1)\sigma(\frac{x+1}{2K+1})$ for any $x\in[0,2K]$, which implies

$$\psi(x)=\sigma\big(x+1-\sigma(x+1)\big)=\sigma\Big((2K+1)\sigma(\tfrac{x+1}{2K+1})-\sigma(x+1)\Big).$$

This means ψ, limited on $[0,2K]$, can be generated by an EUAF network with width 2 and depth 2. Since $0\le 2Kx+\frac{i}{2}\le 2K\frac{9}{10}+2=2K(\frac{9}{10}+\frac{1}{K})\le 2K$ for any $x\in[0,\frac{9}{10}]$, $\psi(2K\cdot+\frac{i}{2})$, limited on $[0,\frac{9}{10}]$, can also be generated by an EUAF network with width 2 and depth 2.

Note that ϕ_i, limited on $[0,1]$, can also be generated by an EUAF network with width 2 and depth 3. Clearly, $x+\frac{i}{4K}\in[0,1]$ for any $x\in[0,\frac{9}{10}]$, and, therefore, $\phi_i(\cdot+\frac{i}{4K})$, limited on $[0,\frac{9}{10}]$, can also be generated by an EUAF network with width 2 and depth 3.

Recall that $\|\phi_i\|_{L^\infty([0,1])}\le\|f\|_{L^\infty([-1,1])}+1=:M$. Thus, $|\phi_i(x+\frac{i}{4K})|\le M$ and $|\psi(2Kx+\frac{i}{2})|\le 1\le M$ for any $x\in[0,\frac{9}{10}]$ and $i=1,2,3,4$. By Lemma 3.3, there exists a function Γ generated by an EUAF network with width 9 and depth 2 such that

$$\Gamma(x,y)=xy\quad\text{for any } x,y\in[-M,M].$$

It follows that

$$\Gamma\Big(\phi_i(x+\tfrac{i}{4K}),\psi\big(2Kx+\tfrac{i}{2}\big)\Big)=\phi_i(x+\tfrac{i}{4K})\psi\big(2Kx+\tfrac{i}{2}\big)\quad\text{for } i=1,2,3,4.$$

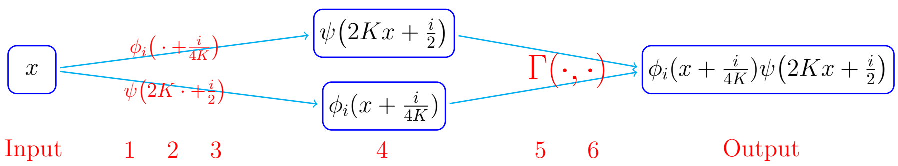


Figure 12: An illustration of the target EUAF network realizing each component of $\phi(x)$, $\phi_i(x+\frac{i}{4K})\psi\big(2Kx+\frac{i}{2}\big)$, for any $x\in[0,\frac{9}{10}]$ and each $i\in\{1,2,3,4\}$. The networks realizing $\phi_i(\cdot+\frac{i}{4K})$ and $\psi\big(2K\cdot+\frac{i}{2}\big)$ can be placed in parallel since we can manually add a hidden layer to ψ since $\sigma\circ\psi\big(2Kx+\frac{i}{2}\big)=\psi\big(2Kx+\frac{i}{2}\big)$ for any $x\in[0,\frac{9}{10}]$.

Therefore, each component of $\phi(x)$, $\phi_i(x+\frac{i}{4K})\psi\big(2Kx+\frac{i}{2}\big)$ for some $i\in\{1,2,3,4\}$, can be generated by the network in Figure 12 for any $x\in[0,\frac{9}{10}]$. Clearly, such a network has width 9 and depth 6. Since the 4-th hidden layer of the network in Figure 12 uses identity as activation function for each neuron in this hidden layer, we can reduce the depth by 1 via composing two adjacent affine linear maps to generate a new one. Thus, the network in Figure 12 can be interpreted as an EUAF network with width 9 and depth 5.

Note that ϕ is the sum of its four components, namely,

$$\phi(x)=\sum_{i=1}^{4}\phi_i(x+\tfrac{i}{4K})\psi\big(2Kx+\tfrac{i}{2}\big)\qquad\text{for any } x\in[0,\tfrac{9}{10}].$$

Therefore, ϕ, limited on $[0, \frac{9}{10}]$, can be generated by an EUAF network with width $9\times4 = 36$ and depth 5 as desired. It is easy to verify that the designed network architecture is independent of the target function f and the desired error ε. That is, we can fix the architecture and only adjust parameters to achieve an arbitrarily desired approximation error. So we finish the proof. □

3.3 Proof of Lemma 3.3

The key idea of proving Lemma 3.3 is the polarization identity $2xy = (x+y)^2 - x^2 - y^2$. Thus, we need to reproduce x^2 locally by an EUAF network as shown in the following lemma.

Lemma 3.4. *There exists a function ϕ generated by an EUAF network with width 3 and depth 2 such that*

$$\phi(x) = x^2 \quad \text{for any } x \in [-1, 1].$$

Proof. Observe that

$$\sigma(y) + 1 = \frac{y}{|y| + 1} + 1 = \frac{y}{-y + 1} + 1 = \frac{1}{-y + 1} \quad \text{for any } y \le 0.$$

For any $x \in [-1, 1]$, we have $-x - 1 \le 0$ and $-x - 2 \le 0$, which implies

$$\begin{aligned}\sigma(-x-1) - \sigma(-x-2) &= \Big(\sigma(-x-1) + 1\Big) - \Big(\sigma(-x-2) + 1\Big) \\ &= \frac{1}{-(-x-1)+1} - \frac{1}{-(-x-2)+1} = \frac{1}{x+2} - \frac{1}{x+3} = \frac{1}{(x+2)(x+3)}.\end{aligned}$$

It follows from $1 - \frac{12}{(x+2)(x+3)} \le 0$ for any $x \in [-1, 1]$ that

$$\sigma\Big(1 - \frac{12}{(x+2)(x+3)}\Big) + 1 = \frac{1}{-\big(1 - \frac{12}{(x+2)(x+3)}\big) + 1} = \frac{x^2 + 5x + 6}{12},$$

implying

$$\begin{aligned}x^2 &= 12\sigma\Big(1 - \frac{12}{(x+2)(x+3)}\Big) + 12 - (5x + 6) \\ &= 12\sigma\Big(1 - 12\big(\sigma(-x-1) - \sigma(-x-2)\big)\Big) + 11\frac{6 - 5x}{11} \\ &= 12\sigma\Big(1 - 12\sigma(-x-1) + 12\sigma(-x-2)\Big) + 11\sigma\Big(\frac{6 - 5x}{11}\Big) \coloneqq \phi(x),\end{aligned}$$

where the equality $\frac{6-5x}{11} = \sigma\big(\frac{6-5x}{11}\big)$ comes from two facts: $\frac{6-5x}{11} \in [0, 1]$ since $x \in [-1, 1]$ and $\sigma(z) = z$ for any $z \in [0, 1]$.

Then, x^2 can be generated by the network shown in Figure 13 for any $x \in [-1, 1]$. The target network has width 3 and depth 2. So we finish the proof. □

With Lemma 3.4 at hand, we are ready to prove Lemma 3.3.

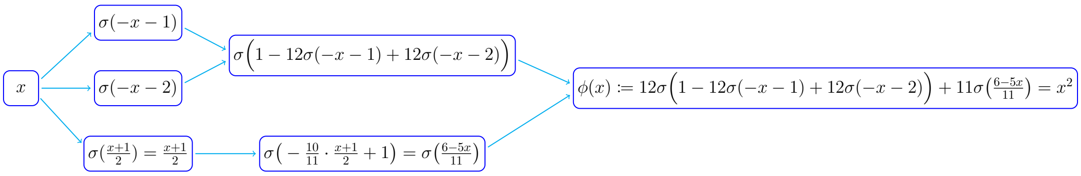


Figure 13: An illustration of the target EUAF network realizing $\phi(x) = x^2$ for $x \in [-1,1]$.

Proof of Lemma 3.3. By Lemma 3.4, there exists a function $\widetilde{\phi}$ generated by an EUAF network such that $\widetilde{\phi}(t) = t^2$ for any $t \in [-1,1]$. Thus, for any $x, y \in [-M, M]$, we have

$$\begin{aligned} xy &= 2M^2\left(\left(\frac{x+y}{2M}\right)^2 - \left(\frac{x}{2M}\right)^2 - \left(\frac{y}{2M}\right)^2\right) \\ &= 2M^2\left(\widetilde{\phi}\left(\frac{x+y}{2M}\right) - \widetilde{\phi}\left(\frac{x}{2M}\right) - \widetilde{\phi}\left(\frac{y}{2M}\right)\right) \coloneqq \phi(x,y). \end{aligned}$$

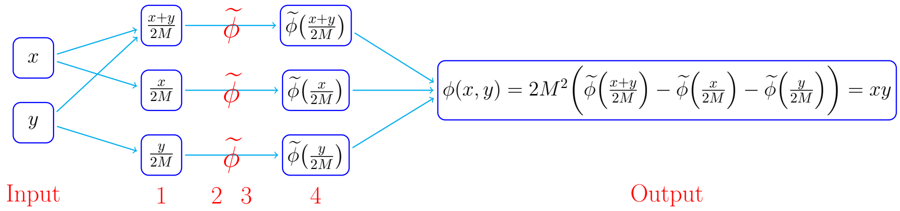


Figure 14: An illustration of the target network realizing $\phi(x) = xy$ for $x, y \in [-M, M]$. "$\xrightarrow{\widetilde{\phi}}$" means the network realizing $\widetilde{\phi}$, i.e., an EUAF network with width 3 and depth 2.

The target network realizing ϕ with width 9 and depth 4 is shown in Figure 14. Note that we can reduce the depth by one if the activation function of each neuron in a hidden layer is identity. In fact, we can eliminate this hidden layer by composing two adjacent affine linear maps to generate a new one. The 1-st and 4-th hidden layers in the network in Figure 14 use identity as an activation function. Thus, the network in Figure 14 can be interpreted as an EUAF network with width 9 and depth 2. So we finish the proof. $\square$

4 Proof of Proposition 2.2

We will prove Proposition 2.2 in this section. The proof includes two main steps. First, we show how to simply generate a set of rationally independent numbers in Lemma 4.1 below. Next, we prove that the target point set via a winding of the generated rationally independent numbers is dense in a hypercube. Such proof relies on the fact that an irrational winding on the torus is dense (e.g., see Lemma 2 of [49]) as shown in Lemma 4.2 below in a hypercube.

Lemma 4.1. *Given any $K \in \mathbb{N}^+$, any transcendental number $\alpha \in \mathbb{R}\backslash\mathbb{A}$, and any pairwise distinct rational numbers $r_1, r_2, \cdots, r_K \in \mathbb{Q}$, the set of numbers*

$$\left\{\frac{1}{\alpha + r_k} : k = 1, 2, \cdots, K\right\}$$

are rationally independent.

Lemma 4.2. *Given any rationally independent numbers $a_1, a_2, \cdots, a_K$ for any $K \in \mathbb{N}^+$ and an arbitrary periodic function $g : \mathbb{R} \to \mathbb{R}$ with period T, i.e., $g(x+T) = g(x)$ for any $x \in \mathbb{R}$, assume there exist $x_1, x_2 \in \mathbb{R}$ with $0 < x_2 - x_1 < T$ such that g is continuous on $[x_1, x_2]$. Then the following set*

$$\left\{ \left[g(wa_1),\, g(wa_2),\, \cdots,\, g(wa_K)\right]^T : w \in \mathbb{R} \right\}$$

is dense in $[M_1, M_2]^K$, where $M_1 = \min_{x\in[x_1,x_2]} g(x)$ and $M_2 = \max_{x\in[x_1,x_2]} g(x)$.

The proofs of these two lemmas can be found in Sections 4.1 and 4.2, respectively. With these two lemmas at hand, the proof of Proposition 2.2 is straightforward. In fact, we can prove a more general result in Proposition 4.3 below, which implies Proposition 2.2 immediately.

Proposition 4.3. *Given an arbitrary periodic function $g : \mathbb{R} \to \mathbb{R}$ with period T, i.e., $g(x+T) = g(x)$ for any $x \in \mathbb{R}$, assume there exist $x_1, x_2 \in \mathbb{R}$ with $0 < x_2 - x_1 < T$ such that g is continuous on $[x_1, x_2]$. Then, for any $K \in \mathbb{N}^+$, any transcendental number $\alpha \in \mathbb{R}\backslash\mathbb{A}$, and any pairwise distinct rational numbers $r_1, r_2, \cdots, r_K \in \mathbb{Q}$, the following set*

$$\left\{ \left[g(\tfrac{w}{\alpha+r_1}),\;\; g(\tfrac{w}{\alpha+r_2}),\;\; \cdots,\;\; g(\tfrac{w}{\alpha+r_K})\right]^T : w \in \mathbb{R} \right\}$$

is dense in $[M_1, M_2]^K$, where $M_1 = \min_{x\in[x_1,x_2]} g(x)$ and $M_2 = \max_{x\in[x_1,x_2]} g(x)$. In the case of $M_1 < M_2$, the following set

$$\left\{ \left[u\cdot g(\tfrac{w}{\alpha+r_1}) + v,\;\; u\cdot g(\tfrac{w}{\alpha+r_2}) + v,\;\; \cdots,\;\; u\cdot g(\tfrac{w}{\alpha+r_K}) + v\right]^T : u, v, w \in \mathbb{R} \right\}$$

is dense in $\mathbb{R}^K$.

Clearly, Proposition 2.2 is a special case of Proposition 4.3 with $g = \sigma_1$, $\alpha = \pi$, $r_k = k$ for $k = 1, 2, \cdots, K$. The transcendence of π is well known (e.g., see the Lindemann–Weierstrass Theorem). By setting $x_1 = 0$ and $x_2 = 1$, we have $[M_1, M_2] = [0, 1]$ and σ_1 is continuous on $[0, 1]$, which means that the following set

$$\left\{ \left[\sigma_1(\tfrac{w}{\pi+1}),\, \sigma_1(\tfrac{w}{\pi+2}),\, \cdots,\, \sigma_1(\tfrac{w}{\alpha+K})\right]^T : w \in \mathbb{R} \right\}$$

is dense in $[0, 1]^K$ as desired.

Finally, let us prove Proposition 4.3 by assuming Lemmas 4.1 and 4.2 are true.

Proof of Proposition 4.3. By Lemma 4.1, the set of numbers

$$\left\{ \tfrac{1}{\alpha+r_k} : k = 1, 2, \cdots, K \right\}$$

are rationally independent. Denote $a_k = \frac{1}{\alpha+r_k}$ for $k = 1, 2, \cdots, K$. Then, by Lemma 4.2,

$$\begin{aligned} &\left\{ \left[g(wa_1),\, g(wa_2),\, \cdots,\, g(wa_K)\right]^T : w \in \mathbb{R} \right\} \\ = &\left\{ \left[g(\tfrac{w}{\alpha+r_1}),\, g(\tfrac{w}{\alpha+r_2}),\, \cdots,\, g(\tfrac{w}{\alpha+r_K})\right]^T : w \in \mathbb{R} \right\} \end{aligned}$$

is dense in $[M_1, M_2]^K$. Now consider the case $M_1 < M_2$ for the latter result. For any $\varepsilon > 0$ and any $\boldsymbol{x} \in \mathbb{R}^K$, by setting $J = \|\boldsymbol{x}\|_\infty + 1 > 0$, we have $\frac{\boldsymbol{x}+J}{2J} \in [0,1]^K$, and hence

$$\boldsymbol{y} \coloneqq \tfrac{\boldsymbol{x}+J}{2J}(M_2 - M_1) + M_1 \in [M_1, M_2]^K.$$

By the former result, there exists $w_0 \in \mathbb{R}$ such that

$$\left\| \boldsymbol{y} - \left[g\big(\tfrac{w_0}{\alpha+r_1}\big),\ g\big(\tfrac{w_0}{\alpha+r_2}\big),\ \cdots,\ g\big(\tfrac{w_0}{\alpha+r_K}\big) \right]^T \right\|_\infty < \tfrac{M_2-M_1}{2J}\varepsilon$$

It follows from $\boldsymbol{y} = \frac{\boldsymbol{x}+J}{2J}(M_2 - M_1) + M_1$ that $\boldsymbol{x} = \frac{2J}{M_2-M_1}\boldsymbol{y} + \frac{J(M_1+M_2)}{M_1-M_2} \eqqcolon u_0\boldsymbol{y} + v_0$, where $u_0 = \frac{2J}{M_2-M_1}$ and $v_0 = \frac{J(M_1+M_2)}{M_1-M_2}$. Therefore,

$$\begin{aligned}
&\left\| \boldsymbol{x} - \left[u_0 g\big(\tfrac{w_0}{\alpha+r_1}\big) + v_0,\ u_0 g\big(\tfrac{w_0}{\alpha+r_2}\big) + v_0,\ \cdots,\ u_0 g\big(\tfrac{w_0}{\alpha+r_K}\big) + v_0 \right]^T \right\|_\infty \\
&= \left\| u_0\boldsymbol{y} + v_0 - \left[u_0 g\big(\tfrac{w_0}{\alpha+r_1}\big) + v_0,\ u_0 g\big(\tfrac{w_0}{\alpha+r_2}\big) + v_0,\ \cdots,\ u_0 g\big(\tfrac{w_0}{\alpha+r_K}\big) + v_0 \right]^T \right\|_\infty < u_0 \tfrac{M_2-M_1}{2J}\varepsilon = \varepsilon.
\end{aligned}$$

Since $\varepsilon > 0$ and $\boldsymbol{x} \in \mathbb{R}^K$ are arbitrary, the following set

$$\left\{ \left[u \cdot g\big(\tfrac{w}{\alpha+r_1}\big) + v,\ u \cdot g\big(\tfrac{w}{\alpha+r_2}\big) + v,\ \cdots,\ u \cdot g\big(\tfrac{w}{\alpha+r_K}\big) + v \right]^T : u, v, w \in \mathbb{R} \right\}$$

is dense in $\mathbb{R}^K$. So we finish the proof. □

4.1 Proof of Lemma 4.1

Before proving Lemma 4.1, let us have a brief discussion on related concepts. Recall that a complex number α is an algebraic number if and only if there exist $\lambda_0, \lambda_1, \cdots, \lambda_J \in \mathbb{Q}$ with $\sum_{j=0}^{J} \lambda_j \alpha^j = 0$. The set of all algebraic numbers is denoted by $\mathbb{A}$. A complex number is called **transcendental** if it is not in $\mathbb{A}$. It is well known that the set $\mathbb{A}$ is **countable**, and, therefore, almost all numbers are transcendental. Therefore, for almost all $\alpha \in \mathbb{R}$, the set of numbers $\left\{\frac{1}{\alpha+k} : k = 1, 2, \cdots, K\right\}$ are rationally independent. The best known transcendental numbers are π (the ratio of a circle's circumference to its diameter) and e (the natural logarithmic base). Thus, both sets of numbers $\left\{\frac{1}{\pi+k} : k = 1, 2, \cdots, K\right\}$ and $\left\{\frac{1}{e+k} : k = 1, 2, \cdots, K\right\}$ are rational independent.

In order to prove Lemma 4.1, we need an auxiliary lemma below, characterizing some properties of coefficients of Lagrange basis polynomials. Recall that, for any given pairwise distinct numbers $x_1, x_2, \cdots, x_K \in \mathbb{R}$, the Lagrange basis polynomials are

$$p_k(x) \coloneqq \prod_{\substack{j \in \{1,2,\cdots,K\} \\ j \neq k}} \frac{x - x_j}{x_k - x_j} = \frac{x - x_1}{x_k - x_1} \cdots \frac{x - x_{k-1}}{x_k - x_{k-1}} \frac{x - x_{k+1}}{x_k - x_{k+1}} \cdots \frac{x - x_K}{x_k - x_K}, \tag{4.1}$$

for $k = 1, 2, \cdots, K$. They are polynomials of degree $\leq K - 1$. Thus, the coefficients of these K Lagrange basis polynomials form a matrix

$$\boldsymbol{A} = (a_{i,j}) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,K} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K,1} & a_{K,2} & \cdots & a_{K,K} \end{bmatrix} \in \mathbb{R}^{K \times K}, \tag{4.2}$$

which satisfies the following equality

$$p_k(x) = \sum_{j=1}^{K} a_{k,j} x^{j-1} = a_{k,1} + a_{k,2}x + \cdots + a_{k,K}x^{K-1} \quad \text{for } k = 1, 2, \cdots, K \text{ and any } x \in \mathbb{R}.$$

The lemma below essentially characterizes the linear independence of Lagrange basis polynomials.

Lemma 4.4. *With the same setting just above, the matrix* $\boldsymbol{A}$ *given in Equation* (4.2) *is invertible.*

Proof. For any $\boldsymbol{y} = [y_1, y_2, \cdots, y_K] \in \mathbb{R}^K$, by the definition of Lagrange basis polynomials $p_k(x)$ for $k = 1, 2, \cdots, K$ in Equation (4.1), $p(x) = \sum_{k=1}^{K} y_k p_k(x)$ is the target interpolation polynomial for sample points $(x_1, y_1), (x_2, y_2), \cdots, (x_K, y_K)$. That is, for any $\ell \in \{1, 2, \cdots, K\}$, we have

$$\begin{aligned}
y_\ell = p(x_\ell) &= \sum_{k=1}^{K} y_k p_k(x_\ell) = \sum_{k=1}^{K} y_k \sum_{j=1}^{K} a_{k,j} x_\ell^{j-1} \\
&= [y_1, y_2, \cdots, y_K] \cdot \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,K} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K,1} & a_{K,2} & \cdots & a_{K,K} \end{bmatrix} \cdot \begin{bmatrix} x_\ell^0 \\ x_\ell^1 \\ \vdots \\ x_\ell^{K-1} \end{bmatrix} = \boldsymbol{y}^T \boldsymbol{A} \begin{bmatrix} x_\ell^0 \\ x_\ell^1 \\ \vdots \\ x_\ell^{K-1} \end{bmatrix}.
\end{aligned}$$

It follows that

$$\boldsymbol{y}^T = [y_1, y_2, \cdots, y_K] = \boldsymbol{y}^T \boldsymbol{A} \begin{bmatrix} x_1^0 & x_2^0 & \cdots & x_K^0 \\ x_1^1 & x_2^1 & \cdots & x_K^1 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{K-1} & x_2^{K-1} & \cdots & x_K^{K-1} \end{bmatrix}.$$

Since $\boldsymbol{y} \in \mathbb{R}^K$ is arbitrary, we have

$$\boldsymbol{A} \begin{bmatrix} x_1^0 & x_2^0 & \cdots & x_K^0 \\ x_1^1 & x_2^1 & \cdots & x_K^1 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{K-1} & x_2^{K-1} & \cdots & x_K^{K-1} \end{bmatrix} = \boldsymbol{I}_K,$$

where $\boldsymbol{I}_K \in \mathbb{R}^{K \times K}$ is the identity matrix. Recall that $x_1, x_2, \cdots, x_K$ are pairwise distinct, which implies the Vandermonde matrix

$$\begin{bmatrix} x_1^0 & x_2^0 & \cdots & x_K^0 \\ x_1^1 & x_2^1 & \cdots & x_K^1 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{K-1} & x_2^{K-1} & \cdots & x_K^{K-1} \end{bmatrix}$$

is invertible. Thus, $\boldsymbol{A}$ is also invertible. So we complete the proof. □

With Lemma 4.4 at hand, we are ready to prove Lemma 4.1.

Proof of Lemma 4.1. Let $x_k = -r_k \in \mathbb{Q}$ for $k = 1, 2, \cdots, K$ and define the Lagrange basis polynomials as

$$p_k(x) \coloneqq \prod_{\substack{j \in \{1,2,\cdots,K\} \\ j \neq k}} \frac{x - x_j}{x_k - x_j} = w_k \prod_{\substack{j \in \{1,2,\cdots,K\} \\ j \neq k}} (x - x_j), \quad \text{where } w_k = \prod_{\substack{j \in \{1,2,\cdots,K\} \\ j \neq k}} \frac{1}{x_k - x_j} \neq 0,$$

for $k = 1, 2, \cdots, K$. Note that w_k is rational and nonzero for any k, which is important for later proof. The coefficients of these K Lagrange basis polynomials form a matrix

$$\boldsymbol{A} = (a_{i,j}) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,K} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K,1} & a_{K,2} & \cdots & a_{K,K} \end{bmatrix} \in \mathbb{R}^{K \times K},$$

which satisfies the following equality

$$p_k(x) = \sum_{j=1}^{K} a_{k,j} x^{j-1} = a_{k,1} + a_{k,2} x + \cdots + a_{k,K} x^{K-1} \quad \text{for } k = 1, 2, \cdots, K \text{ and any } x \in \mathbb{R}.$$

Now assume there exist $\lambda_1, \lambda_2, \cdots, \lambda_K \in \mathbb{Q}$ such that $\sum_{k=1}^{K} \lambda_k \cdot \frac{1}{\alpha + r_k} = 0$. Our goal is to prove $\lambda_1 = \lambda_2 = \cdots = \lambda_K = 0$. Clearly, we have

$$\begin{aligned} 0 &= \sum_{k=1}^{K} \lambda_k \cdot \frac{1}{\alpha + r_k} = \underbrace{\sum_{k=1}^{K} \frac{\lambda_k}{\alpha - x_k}}_{=0} = \prod_{j=1}^{K} (\alpha - x_j) \cdot \underbrace{\sum_{k=1}^{K} \frac{\lambda_k}{\alpha - x_k}}_{=0} = \sum_{k=1}^{K} \frac{\lambda_k}{w_k} \cdot w_k \prod_{\substack{j \in \{1,2,\cdots,K\} \\ j \neq k}} (\alpha - x_j) \\ &= \sum_{k=1}^{K} \frac{\lambda_k}{w_k} \cdot p_k(\alpha) = \sum_{k=1}^{K} \frac{\lambda_k}{w_k} \sum_{j=1}^{K} a_{k,j} \alpha^{j-1} = \sum_{j=1}^{K} \Bigg(\underbrace{\sum_{k=1}^{K} \frac{\lambda_k}{w_k} a_{k,j}}_{=0 \text{ since } \alpha \in \mathbb{R} \backslash \mathbb{A}} \Bigg) \cdot \alpha^{j-1}. \end{aligned}$$

Note that $\alpha \in \mathbb{R} \backslash \mathbb{A}$ is not an algebraic number and $\frac{\lambda_k}{w_k} \in \mathbb{Q}$ since $\lambda_k, w_k \in \mathbb{Q}$ for any k. Thus, the coefficients must be 0, namely,

$$\sum_{k=1}^{K} \frac{\lambda_k}{w_k} a_{k,j} = 0 \quad \text{for } j = 1, 2, \cdots, K.$$

It follows that

$$\boldsymbol{0} = \left[\tfrac{\lambda_1}{w_1}, \tfrac{\lambda_2}{w_2}, \cdots, \tfrac{\lambda_K}{w_K}\right] \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,K} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K,1} & a_{K,2} & \cdots & a_{K,K} \end{bmatrix} = \left[\tfrac{\lambda_1}{w_1}, \tfrac{\lambda_2}{w_2}, \cdots, \tfrac{\lambda_K}{w_K}\right] \boldsymbol{A}.$$

By Lemma 4.4, $\boldsymbol{A}$ is invertible. Thus, $\left[\frac{\lambda_1}{w_1}, \frac{\lambda_2}{w_2}, \cdots, \frac{\lambda_K}{w_K}\right] = \boldsymbol{0}$, which implies $\lambda_1 = \lambda_2 = \cdots = \lambda_K = 0$. Hence, the set of numbers $\left\{\frac{1}{\alpha + r_k} : k = 1, 2, \cdots, K\right\}$ are rationally independent, which means we finish the proof. $\square$

4.2 Proof of Lemma 4.2

The proof of Lemma 4.2 is mainly based on the fact that an irrational winding is dense on the torus (e.g., see Lemma 2 of [49]). For completeness, we establish a lemma below and give its detailed proof.

Lemma 4.5. *Given any $K \in \mathbb{N}^+$ and an arbitrary set of rationally independent numbers $\{a_k : k = 1, 2, \cdots, K\} \subseteq \mathbb{R}$, the following set*

$$\Big\{ \big[\tau(wa_1),\ \tau(wa_2),\ \cdots,\ \tau(wa_K)\big]^T : w \in \mathbb{R} \Big\} \subseteq [0,1)^K$$

is dense in $[0,1]^K$, where $\tau(x) \coloneqq x - \lfloor x \rfloor$ for any $x \in \mathbb{R}$.

The proof of Lemma 4.5 can be found later in this section. Now let us first prove Lemma 4.2 by assuming Lemma 4.5 is true.

Proof of Lemma 4.2. Define $\widetilde{g}(x) \coloneqq g(Tx)$ for any $x \in \mathbb{R}$. The continuity of g on $[x_1, x_2]$ implies $\widetilde{g}$ is continuous on $[\frac{x_1}{T}, \frac{x_2}{T}]$, and, therefore, uniformly continuous on $[\frac{x_1}{T}, \frac{x_2}{T}]$. For any $\varepsilon > 0$, there exists $\delta \in (0, \frac{x_2 - x_1}{T})$ such that

$$|\widetilde{g}(u) - \widetilde{g}(v)| < \varepsilon \quad \text{for any } u, v \in [\tfrac{x_1}{T}, \tfrac{x_2}{T}] \text{ with } |u - v| < \delta. \tag{4.3}$$

Given any $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_K] \in [M_1, M_2]^K$, by the intermediate value theorem, there exists $z_1, z_2, \cdots, z_K \in [x_1, x_2]$ such that $g(z_k) = \xi_k$ for any $k = 1, 2, \cdots, K$.

For any $k = 1, 2, \cdots, K$, set $y_k = z_k/T \in [\frac{x_1}{T}, \frac{x_2}{T}]$ and

$$\widetilde{y}_k = y_k + \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \le \frac{x_1}{T} + \frac{\delta}{2}\}} - \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \ge \frac{x_2}{T} - \frac{\delta}{2}\}}.$$

Then, for $k = 1, 2, \cdots, K$, we have

$$\widetilde{y}_k = y_k + \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \le \frac{x_1}{T} + \frac{\delta}{2}\}} - \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \ge \frac{x_2}{T} - \frac{\delta}{2}\}} \in \Big[\tfrac{x_1}{T} + \tfrac{\delta}{2},\ \tfrac{x_2}{T} - \tfrac{\delta}{2}\Big]$$

and

$$|\widetilde{y}_k - y_k| \le \Big| \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \le \frac{x_1}{T} + \frac{\delta}{2}\}} - \tfrac{\delta}{2} \cdot \mathbb{1}_{\{y_k \ge \frac{x_2}{T} - \frac{\delta}{2}\}} \Big| \le \delta/2.$$

Define $\tau(x) = x - \lfloor x \rfloor$ for any $x \in \mathbb{R}$. Clearly, $[\tau(\widetilde{y}_1), \tau(\widetilde{y}_2), \cdots, \tau(\widetilde{y}_K)]^T \in [0,1]^K$. Then by Lemma 4.5, there exists $w_0 \in \mathbb{R}$ such that

$$|\tau(w_0 a_k) - \tau(\widetilde{y}_k)| < \delta/2 \quad \text{for } k = 1, 2, \cdots, K.$$

It follows that

$$\Big|\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor - \widetilde{y}_k\Big| = \Big|\tau(w_0 a_k) - (\widetilde{y}_k - \lfloor \widetilde{y}_k \rfloor)\Big| = \Big|\tau(w_0 a_k) - \tau(\widetilde{y}_k)\Big| < \delta/2,$$

for $k = 1, 2, \cdots, K$. Since $\widetilde{y}_k \in [\frac{x_1}{T} + \frac{\delta}{2}, \frac{x_2}{T} - \frac{\delta}{2}]$, we have $\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor \in [\frac{x_1}{T}, \frac{x_2}{T}]$. Besides,

$$\Big|\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor - y_k\Big| \le \Big|\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor - \widetilde{y}_k\Big| + \Big|\widetilde{y}_k - y_k\Big| < \delta/2 + \delta/2 = \delta,$$

for $k = 1, 2, \cdots, K$. Then, by Equation (4.3), we have

$$\left|\widetilde{g}\big(\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor\big) - \widetilde{g}(y_k)\right| < \varepsilon \quad \text{for } k = 1, 2, \cdots, K.$$

By the definition of $\widetilde{g}$, it is periodic with period 1 since g is periodic with period T. This implies

$$\widetilde{g}\big(\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor\big) = \widetilde{g}\big(w_0 a_k - \lfloor w_0 a_k \rfloor + \lfloor \widetilde{y}_k \rfloor\big) = \widetilde{g}(w_0 a_k) = g(T \cdot w_0 a_k),$$

for $k = 1, 2, \cdots, K$. Also, $\widetilde{g}(y_k) = g(T y_k) = g(z_k) = \xi_k$ for $k = 1, 2, \cdots, K$. It follows that

$$\left|g(T \cdot w_0 a_k) - \xi_k\right| = \left|\widetilde{g}\big(\tau(w_0 a_k) + \lfloor \widetilde{y}_k \rfloor\big) - \widetilde{g}(y_k)\right| < \varepsilon \quad \text{for } k = 1, 2, \cdots, K.$$

That is

$$\left\|\big[g(w_1 a_1),\, g(w_1 a_2),\, \cdots,\, g(w_1 a_K)\big]^T - \boldsymbol{\xi}\right\|_\infty < \varepsilon,$$

where $w_1 = T \cdot w_0 \in \mathbb{R}$. Since $\boldsymbol{\xi} \in [M_1, M_2]^K$ and $\varepsilon > 0$ are arbitrary, the following set

$$\Big\{ [g(w a_1),\, g(w a_2),\, \cdots,\, g(w a_K)]^T : w \in \mathbb{R} \Big\}$$

is dense in $[M_1, M_2]^K$ as desired. So we finish the proof. □

Finally, let us present the detailed proof of Lemma 4.5.

Proof of Lemma 4.5. We prove this lemma by mathematical induction. First, we consider the case $K = 1$. Note that $a_1 \neq 0$ since it is rationally independent. Thus, we have $\{\tau(w a_1) : w \in \mathbb{R}\} = [0, 1)$, which implies $\{\tau(w a_1) : w \in \mathbb{R}\}$ is dense in $[0, 1]$.

Now assume this lemma holds for $K = J - 1 \in \mathbb{N}^+$. Our goal is to prove the case $K = J$. Given any $\varepsilon \in (0, 1/100)$ and arbitrary $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_J]^T \in [0, 1]^J$, our goal is to find a proper $w \in \mathbb{R}$ such that

$$|\tau(w a_j) - \xi_j| < C\varepsilon \quad \text{for } j = 1, 2, \cdots, J, \quad \text{where } C \text{ is an absolute constant.} \tag{4.4}$$

As we shall see later, we need an assumption that the given point is in $[6\varepsilon, 1 - 6\varepsilon]^J$. Thus, we set

$$\widetilde{\xi}_j = \xi_j + 6\varepsilon \cdot \mathbb{1}_{\{\xi_j \le 6\varepsilon\}} - 6\varepsilon \cdot \mathbb{1}_{\{\xi_j \ge 1 - 6\varepsilon\}} \quad \text{for } j = 1, 2, \cdots, J.$$

Then, we have

$$\widetilde{\xi}_j \in [6\varepsilon, 1 - 6\varepsilon] \quad \text{for } j = 1, 2, \cdots, J \tag{4.5}$$

and

$$\left|\xi_j - \widetilde{\xi}_j\right| = \left|6\varepsilon \cdot \mathbb{1}_{\{\xi_j \le 6\varepsilon\}} - 6\varepsilon \cdot \mathbb{1}_{\{\xi_j \ge 1 - 6\varepsilon\}}\right| \le 6\varepsilon \quad \text{for } j = 1, 2, \cdots, J. \tag{4.6}$$

Define

$$\widehat{\xi}_j \coloneqq \tau\big(\widetilde{\xi}_j - \tfrac{\widetilde{\xi}_J}{a_J} a_j\big) \quad \text{for } j = 1, 2, \cdots, J. \tag{4.7}$$

Then $\widehat{\xi}_J = 0$ and $\widehat{\xi}_j \in [0, 1)$ for $j = 1, 2, \cdots, J - 1$. To approximate $[\widehat{\xi}_1, \widehat{\xi}_2, \cdots, \widehat{\xi}_{J-1}]^T \in [0, 1)^{J-1}$, we only need to consider $J - 1$ indices, and, therefore, we can use the induction hypothesis to continue our proof.

Clearly, the rational independence of $a_1, a_2, \cdots, a_J$ implies none of them is equal to zero. Define

$$\boldsymbol{b}_n := \left[\tau\left(\tfrac{n}{a_J}a_1\right),\ \tau\left(\tfrac{n}{a_J}a_2\right),\ \cdots,\ \tau\left(\tfrac{n}{a_J}a_{J-1}\right)\right]^T \in [0,1)^{J-1}.$$

Then the bounded sequence $(\boldsymbol{b}_n)_{n=1}^{\infty}$ has a convergent subsequence by the Bolzano-Weierstrass Theorem. Thus, there exist $n_1, n_2 \in \mathbb{N}^+$ with $n_1 < n_2$ such that $\|\boldsymbol{b}_{n_2} - \boldsymbol{b}_{n_1}\|_\infty < \varepsilon$. That is,

$$\left|\tau\left(\tfrac{n_2}{a_J}a_j\right) - \tau\left(\tfrac{n_1}{a_J}a_j\right)\right| < \varepsilon \quad \text{for } j = 1, 2, \cdots, J-1.$$

Set $\widehat{n} = n_2 - n_1 \in \mathbb{N}^+$ and $k_j = \lfloor \tfrac{n_1}{a_J}a_j \rfloor - \lfloor \tfrac{n_2}{a_J}a_j \rfloor$ for $j = 1, 2, \cdots, J-1$. Then, by defining

$$\widehat{a}_j := \tfrac{\widehat{n}}{a_J}a_j + k_j \quad \text{for } j = 1, 2, \cdots, J-1,$$

we have

$$\begin{aligned} |\widehat{a}_j| = \left|\tfrac{\widehat{n}}{a_J}a_j + k_j\right| &= \left|\tfrac{n_2}{a_J}a_j - \tfrac{n_1}{a_J}a_j + \lfloor \tfrac{n_1}{a_J}a_j \rfloor - \lfloor \tfrac{n_2}{a_J}a_j \rfloor\right| \\ &= \left|\left(\tfrac{n_2}{a_J}a_j - \lfloor \tfrac{n_2}{a_J}a_j \rfloor\right) - \left(\tfrac{n_1}{a_J}a_j - \lfloor \tfrac{n_1}{a_J}a_j \rfloor\right)\right| = \left|\tau\left(\tfrac{n_2}{a_J}a_j\right) - \tau\left(\tfrac{n_1}{a_J}a_j\right)\right| < \varepsilon. \end{aligned} \tag{4.8}$$

It is easy to verify that $\widehat{a}_1, \widehat{a}_2, \cdots, \widehat{a}_{J-1}$ are rationally independent. To see this, assume there exist $\lambda_1, \lambda_2, \cdots, \lambda_{J-1} \in \mathbb{Q}$ such that

$$0 = \sum_{j=1}^{J-1} \lambda_j \widehat{a}_j = \sum_{j=1}^{J-1} \lambda_j \left(\tfrac{\widehat{n}}{a_J}a_j + k_j\right) = \sum_{j=1}^{J-1} \lambda_j \tfrac{\widehat{n}}{a_J}a_j + \sum_{j=1}^{J-1} \lambda_j k_j,$$

then

$$0 = \sum_{j=1}^{J-1} \lambda_j \widehat{n} a_j + \Big(\sum_{j=1}^{J-1} \lambda_j k_j\Big) a_J.$$

Since $a_1, a_2, \cdots, a_J$ are rationally independent, we have $\lambda_j \widehat{n} = 0$ for $j = 1, 2, \cdots, J-1$. It follows from $\widehat{n} = n_2 - n_1 > 0$ that $\lambda_1 = \lambda_2 = \cdots = \lambda_{J-1} = 0$. Thus, $\widehat{a}_1, \widehat{a}_2, \cdots, \widehat{a}_{J-1}$ are rationally independent as desired.

By the induction hypothesis, the following set

$$\Big\{ \left[\tau(s \cdot \widehat{a}_1),\ \tau(s \cdot \widehat{a}_2),\ \cdots,\ \tau(s \cdot \widehat{a}_{J-1})\right]^T : s \in \mathbb{R} \Big\} \subseteq [0,1)^{J-1}$$

is dense in $[0,1]^{J-1}$. Recall that $\widehat{\xi}_j \in [0,1]$ for $j = 1, \cdots, J-1$, which implies

$$\widehat{\xi}_j + 3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \le 3\varepsilon\}} - 3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \ge 1-3\varepsilon\}} \in [3\varepsilon, 1-3\varepsilon] \quad \text{for } j = 1, \cdots, J-1.$$

Hence, there exists $s_0 \in \mathbb{R}$ such that

$$\left|\tau(s_0 \widehat{a}_j) - \left(\widehat{\xi}_j + 3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \le 3\varepsilon\}} - 3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \ge 1-3\varepsilon\}}\right)\right| < \varepsilon \quad \text{for } j = 1, \cdots, J-1.$$

It follows that

$$\tau(s_0 \widehat{a}_j) \in [2\varepsilon, 1-2\varepsilon] \quad \text{for } j = 1, \cdots, J-1$$

and

$$\left|\tau(s_0 \widehat{a}_j) - \widehat{\xi}_j\right| < \varepsilon + \left|3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \le 3\varepsilon\}} - 3\varepsilon \cdot \mathbb{1}_{\{\widehat{\xi}_j \ge 1-3\varepsilon\}}\right| \le 4\varepsilon \quad \text{for } j = 1, \cdots, J-1. \tag{4.9}$$

To estimate $\tau(\lfloor s_0 \rceil \widehat{a}_j) - \widehat{\xi}_j$, we need to bound $\tau(s_0 \widehat{a}_j) - \tau(\lfloor s_0 \rceil \widehat{a}_j)$. To this end, we need an observation for any $x, y \in \mathbb{R}$ as follows.

$$|x - y| < \varepsilon \quad \text{and} \quad \tau(x) \in [2\varepsilon, 1 - 2\varepsilon] \quad \Longrightarrow \quad |\tau(x) - \tau(y)| < \varepsilon. \tag{4.10}$$

In fact, $\tau(x) \in [2\varepsilon, 1-2\varepsilon]$ implies $\varepsilon \le \tau(x) - \varepsilon \le \tau(x) + \varepsilon \le 1 - \varepsilon$, deducing

$$y \in [x - \varepsilon, x + \varepsilon] = \Big[\lfloor x \rfloor + \underbrace{\tau(x) - \varepsilon}_{\ge \varepsilon}, \lfloor x \rfloor + \underbrace{\tau(x) + \varepsilon}_{\le 1-\varepsilon}\Big] \subseteq \Big[\lfloor x \rfloor + \varepsilon, \lfloor x \rfloor + 1 - \varepsilon\Big] \subseteq \Big[\lfloor x \rfloor, \lfloor x \rfloor + 1\Big).$$

Thus, $\lfloor y \rfloor = \lfloor x \rfloor$, which implies $|\tau(x) - \tau(y)| = \big|\tau(x) - \tau(y) + \lfloor x \rfloor - \lfloor y \rfloor\big| = |x - y| < \varepsilon$ as desired.

By Equation (4.8), we have

$$\Big| s_0 \widehat{a}_j - \lfloor s_0 \rceil \widehat{a}_j \Big| \le \Big| s_0 - \lfloor s_0 \rceil \Big| \cdot |\widehat{a}_j| < \varepsilon \quad \text{for } j = 1, 2, \cdots, J-1.$$

Recall that $\tau(s_0 \widehat{a}_j) \in [2\varepsilon, 1 - 2\varepsilon]$ for $j = 1, \cdots, J-1$. Then, for each $j \in \{1, 2, \cdots, J-1\}$, by the observation above in Equation (4.10) (set $x = s_0 \widehat{a}_j$ and $y = \lfloor s_0 \rceil \widehat{a}_j$ therein), we have $\big|\tau(s_0 \widehat{a}_j) - \tau(\lfloor s_0 \rceil \widehat{a}_j)\big| < \varepsilon$. Therefore, by Equations (4.7) and (4.9), we have

$$\begin{aligned}
\Big|\tau(\lfloor s_0 \rceil \widehat{a}_j) - \tau(\widetilde{\xi}_j - \tfrac{\widetilde{\xi}_J}{a_J} a_j)\Big| &= \Big|\tau(\lfloor s_0 \rceil \widehat{a}_j) - \widehat{\xi}_j\Big| \\
&\le \Big|\tau(\lfloor s_0 \rceil \widehat{a}_j) - \tau(s_0 \widehat{a}_j)\Big| + \Big|\tau(s_0 \widehat{a}_j) - \widehat{\xi}_j\Big| < \varepsilon + 4\varepsilon = 5\varepsilon,
\end{aligned}$$

for $j = 1, 2, \cdots, J-1$. Recall the fact: For any $x, y \in \mathbb{R}$, it holds that $\tau(x) - \tau(y) = x - \lfloor x \rfloor - \big(y - \lfloor y \rfloor\big) = x - y - z$, where $z = \lfloor x \rfloor - \lfloor y \rfloor \in \mathbb{Z}$.

Therefore, for $j = 1, 2, \cdots, J-1$, there exists $z_j \in \mathbb{Z}$ such that

$$\tau(\lfloor s_0 \rceil \widehat{a}_j) - \tau(\widetilde{\xi}_j - \tfrac{\widetilde{\xi}_J}{a_J} a_j) = \lfloor s_0 \rceil \widehat{a}_j - \big(\widetilde{\xi}_j - \tfrac{\widetilde{\xi}_J}{a_J} a_j\big) - z_j = \lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j - (z_j + \widetilde{\xi}_j),$$

which implies

$$\Big| \lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j - (z_j + \widetilde{\xi}_j) \Big| = \Big| \tau(\lfloor s_0 \rceil \widehat{a}_j) - \tau(\widetilde{\xi}_j - \tfrac{\widetilde{\xi}_J}{a_J} a_j) \Big| < 5\varepsilon.$$

It follows that

$$\lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j \in [z_j + \underbrace{\widetilde{\xi}_j - 5\varepsilon}_{\ge \varepsilon}, z_j + \underbrace{\widetilde{\xi}_j + 5\varepsilon}_{\le 1-\varepsilon}] \subseteq [z_j + \varepsilon, z_j + 1 - \varepsilon] \quad \text{for } j = 1, 2, \cdots, J-1,$$

where the fact $\varepsilon \le \widetilde{\xi}_j - 5\varepsilon \le \widetilde{\xi}_j + 5\varepsilon \le 1 - \varepsilon$ comes from Equation (4.5). Therefore,

$$\tau(\lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j) = \big(\lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j\big) - z_j \in [\widetilde{\xi}_j - 5\varepsilon, \widetilde{\xi}_j + 5\varepsilon] \quad \text{for } j = 1, 2, \cdots, J-1.$$

For $j = 1, 2, \cdots, J-1$, we have

$$\lfloor s_0 \rceil \widehat{a}_j + \tfrac{\widetilde{\xi}_J}{a_J} a_j = \lfloor s_0 \rceil \Big(\tfrac{\widehat{n}}{a_J} a_j + k_j \Big) + \tfrac{\widetilde{\xi}_J}{a_J} a_j = \tfrac{\lfloor s_0 \rceil \widehat{n} + \widetilde{\xi}_J}{a_J} a_j + \underbrace{k_j \lfloor s_0 \rceil}_{\in \mathbb{Z}},$$

which implies

$$\tau\Big(\tfrac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}a_j\Big)=\tau\Big(\lfloor s_0\rfloor\widehat{a}_j+\tfrac{\widetilde{\xi}_J}{a_J}a_j\Big)\in[\widetilde{\xi}_j-5\varepsilon,\widetilde{\xi}_j+5\varepsilon]\quad\text{for } j=1,2,\cdots,J-1.$$

By Equation (4.5), we have $\widetilde{\xi}_J\in[6\varepsilon,1-6\varepsilon]$, which implies

$$\tau\Big(\tfrac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}a_J\Big)=\tau\big(\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J\big)=\widetilde{\xi}_J.$$

Thus, for $j=1,2,\cdots,J$, we have

$$\Big|\tau\Big(\tfrac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}a_j\Big)-\widetilde{\xi}_j\Big|\le 5\varepsilon.$$

By Equation (4.6), we have $|\widetilde{\xi}_j-\xi_j|<6\varepsilon$ for $j=1,2,\cdots,J$, which implies

$$\Big|\tau\Big(\tfrac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}a_j\Big)-\xi_j\Big|\le\Big|\tau\Big(\tfrac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}a_j\Big)-\widetilde{\xi}_j\Big|+\big|\widetilde{\xi}_j-\xi_j\big|\le 5\varepsilon+6\varepsilon=11\varepsilon.$$

Therefore, $w_0=\frac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}$ is the desired w in Equation (4.4). That is,

$$\big|\tau(w_0a_j)-\xi_j\big|\le 11\varepsilon\quad\text{for } j=1,2,\cdots,J.$$

Since $\boldsymbol{\xi}=[\xi_1,\xi_2,\cdots,\xi_J]^T\in[0,1]^J$ is arbitrary, the following set

$$\Big\{\big[\tau(wa_1),\ \tau(wa_2),\ \cdots,\ \tau(wa_J)\big]^T: w\in\mathbb{R}\Big\}\subseteq[0,1)^J$$

is dense in $[0,1]^J$ as desired. We finish the process of mathematical induction, and, therefore, finish the proof by the principle of mathematical induction. □

We remark that the target parameter $w_0=\frac{\lfloor s_0\rfloor\widehat{n}+\widetilde{\xi}_J}{a_J}$ designed in the above proof may not be bounded uniformly for any approximation error ε since $\widehat{n}$ can be arbitrarily large depending on ε. Therefore, the network in Theorem 1.1 may require sufficiently large parameters to achieve a target error ε.

5 Other examples of UAFs

This section aims at designing new UAFs with additional properties such as smooth or sigmoidal functions. As discussed in the introduction and shown in the proof of our main theorem, the construction of UAFs mainly relies on three properties: high nonlinearity, periodicity, and the capacity to reproduce step functions. The EUAF σ defined in Equation (1.3) is a simple and typical example of UAFs satisfying these three properties. Indeed, having these properties plays an important role in our proof and is a necessary but not sufficient condition for designing a UAF. In other words, these properties are important, but cannot guarantee the successful construction of UAFs.

Here, we present another idea to design new UAFs, which mainly relies on the following observation: If a UAF ϱ can be approximated by a fixed-size network activated by a new function $\widetilde{\varrho}$ within an arbitrary error on any bounded interval, then $\widetilde{\varrho}$ is also a UAF. Such an observation is a direct result of the lemma below.

Lemma 5.1. *Let $\varrho, \widetilde{\varrho} : \mathbb{R} \to \mathbb{R}$ be two functions with $\varrho \in C(\mathbb{R})$. For an arbitrary given function $f \in [a,b]^d \to \mathbb{R}$ and $\varepsilon > 0$, suppose that the following two conditions hold:*

- *There exists a function ϕ_ϱ realized by a ϱ-activated network with width N and depth L such that*
$$|\phi_\varrho(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon/2 \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

- *For any $M > 0$ and each $\delta \in (0,1)$, there exists a function ϱ_δ realized by a $\widetilde{\varrho}$-activated network with width $\widetilde{N}$ and depth $\widetilde{L}$ such that*
$$\varrho_\delta(t) \rightrightarrows \varrho(t) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } t \in [-M, M],$$
where $\rightrightarrows$ denotes the uniform convergence.

Then, there exists a function $\phi = \phi_{\widetilde{\varrho}}$ generated by a $\widetilde{\varrho}$-activated network with width $N\widetilde{N}$ and depth $L\widetilde{L}$ such that
$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

The proof of Lemma 5.1 is placed in Section 5.3. Based on Lemma 5.1, we will propose two UAFs with better mathematical properties. That is, the idea of designing a C^s UAF is given in Section 5.1 and a sigmoidal UAF is constructed in detail in Section 5.2.

5.1 Smooth UAF

The smoothness of a function is one of the most desired properties in mathematical modeling and computation. The EUAF σ is continuous but not smooth. So we will show how to construct a C^s UAF based on an existing one. The key point is the fact that the integral of a continuous function is continuously differentiable.

Suppose ϱ is a continuous UAF. Define
$$\widetilde{\varrho}(x) := \int_0^x \varrho(t)\mathrm{d}t \quad \text{for any } x \in \mathbb{R}.$$
For any $M > 0$, it holds that
$$\frac{\widetilde{\varrho}(x+\delta) - \widetilde{\varrho}(x)}{\delta} = \frac{1}{\delta}\int_x^{x+\delta} \varrho(t)\mathrm{d}t \rightrightarrows \varrho(x) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } x \in [-M, M].$$
This means ϱ can be approximated by a one-hidden-layer $\widetilde{\varrho}$-activated network with width 2 arbitrarily well on any bounded interval. It follows that $\widetilde{\varrho}$ is also a UAF. By repeated applications of the above idea, one could easily construct a C^s UAF.

In particular, set $\varrho_0 = \sigma$ and define $\varrho_1, \varrho_2, \cdots, \varrho_s$ by induction as follows.
$$\varrho_{i+1}(x) := \int_0^x \varrho_i(t)\mathrm{d}t \quad \text{for any } x \in \mathbb{R} \text{ and } i \in \{0, 1, \cdots, s-1\}. \tag{5.1}$$
Then, ϱ_s is a C^s UAF as shown in the following theorem.

Theorem 5.2. *Let $\varrho_s \in C^s(\mathbb{R})$ be the function defined in Equation (5.1) for any $s \in \mathbb{N}^+$. Then, for any $f \in C([a,b]^d)$ and $\varepsilon > 0$, there exists a function ϕ generated by a ϱ_s-activated network with width $72sd(2d+1)$ and depth 11 such that*

$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Proof. For any $i \in \{0, 1, \cdots, s-1\}$ and any $M > 0$, it is easy to verify that

$$\frac{\varrho_{i+1}(x+\delta) - \varrho_{i+1}(x)}{\delta} = \frac{1}{\delta}\int_x^{x+\delta} \varrho_i(t)\mathrm{d}t \rightrightarrows \varrho_i(x) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } x \in [-M, M].$$

This means ϱ_i can be approximated by a one-hidden-layer ϱ_{i+1}-activated network with width 2 arbitrarily well on any bounded interval. By induction, one could easily prove that $\varrho_0 = \sigma$ can be approximated by a one-hidden-layer ϱ_s-activated network with width $2s$ arbitrarily well on any bounded interval. That is, for each $\delta \in (0,1)$, there exists a function $\sigma_{s,\delta}$ realized by a ϱ_s-activated network with width $2s$ and depth 1 such that

$$\sigma_{s,\delta}(t) \rightrightarrows \sigma(t) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } t \in [-M, M].$$

By Theorem 1.1, there exists a function ϕ_σ generated by a σ-activated network with width $36d(2d+1)$ and depth 11 such that

$$|\phi_\sigma(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon/2 \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Then, by Lemma 5.1, there exists another function $\phi = \phi_{\varrho_s}$ realized by a ϱ_s-activated network with width $2s \times 36d(2d+1) = 72sd(2d+1)$ and depth $1 \times 11 = 11$ such that

$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

So we finish the proof. □

5.2 Sigmoidal UAF

Many activation functions used in real applications are sigmoidal functions. Generally, we say a function $g: \mathbb{R} \to \mathbb{R}$ is sigmoidal (or sigmoid, e.g., see [20]) if it satisfies the following conditions.

- Bounded: $\lim_{x\to\infty} g(x) = 1$ and $\lim_{x\to-\infty} g(x) = -1$ (or 0).
- Differentiable: $g'(x)$ exists and continuous for all $x \in \mathbb{R}$.
- Increasing: $g'(x)$ is non-negative for all $x \in \mathbb{R}$.

Our goal is to construct a sigmoidal UAF. To this end, we need to design a new function $\widetilde{\sigma}$ based on σ such that σ can be reproduced/approximated by a $\widetilde{\sigma}$-activated network with a fixed size. Making $\widetilde{\sigma}$ bounded and increasing is not difficult. The key is to make $\widetilde{\sigma}$ continuously differentiable, which can be true by the fact that the integral of a continuous function is continuously differentiable. To be exact, we can define $\widetilde{\sigma}$ as follows.

- For $x \in (-\infty, 0]$, define $\widetilde{\sigma}(x) \coloneqq \sigma(x) = \frac{x}{-x+1}$.

- For $x \in (0, \infty)$, define

$$\widetilde{\sigma}(x) := \int_0^x \frac{c\sigma(t)+1}{(2t+1)^2} \mathrm{d}t, \quad \text{where} \quad c = \frac{1}{2\int_0^\infty \frac{\sigma(t)}{(2t+1)^2}\mathrm{d}t} \approx 2.554.$$

Remark that there are many possible choices for the integrand in the above definition of $\sigma(x)$ for $x \in (0, \infty)$. Here, we just give a simple example. See an illustration of $\widetilde{\sigma}$ in Figure 15.

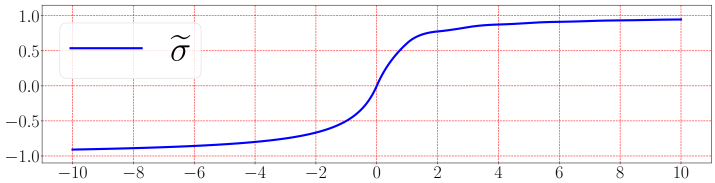


Figure 15: An illustration of $\widetilde{\sigma}$ on $[-10, 10]$.

Then, $\widetilde{\sigma}$ is a sigmoidal function as verified below.

- Clearly, $\lim_{x\to-\infty} \widetilde{\sigma}(x) = \lim_{x\to-\infty} \frac{x}{-x+1} = -1$. Moreover,

$$\lim_{x\to\infty} \widetilde{\sigma}(x) = \int_0^\infty \frac{c\sigma(t)+1}{(2t+1)^2}\mathrm{d}t = \frac{1}{2} + \int_0^\infty \frac{1}{(2t+1)^2}\mathrm{d}t = 1.$$

- Obviously, $\widetilde{\sigma}$ is continuously differentiable on $(-\infty, 0)$ and $(0, \infty)$. Meanwhile, we have $\widetilde{\sigma}'(0) = 1$ and $\lim_{x\to 0} \widetilde{\sigma}'(x) = 1$. Therefore, we have $\widetilde{\sigma} \in C^1(\mathbb{R})$ as desired.

- For $x \in (-\infty, 0)$, $\widetilde{\sigma}'(x) = \frac{1}{(-x+1)^2} > 0$. For $x = 0$, $\widetilde{\sigma}'(x) = 1 > 0$. For $x \in (0, \infty)$, $\widetilde{\sigma}'(x) = \frac{c\sigma(x)+1}{(2x+1)^2} > 0$. That is, $\widetilde{\sigma}'(x) > 0$ for all $x \in \mathbb{R}$.

Based on Theorem 1.1 corresponding to σ, we establish a similar theorem for $\widetilde{\sigma}$, Theorem 5.3 below, showing that fixed-size $\widetilde{\sigma}$-activated networks can also approximate continuous functions within an arbitrary error on a hypercube.

Theorem 5.3. *For any $f \in C([a,b]^d)$ and $\varepsilon > 0$, there exists a function ϕ generated by a $\widetilde{\sigma}$-activated network with width $1044d(2d+1)$ and depth 66 such that*

$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

To prove this theorem based on Theorem 1.1, we only need to show σ can be approximated by a fixed-size $\widetilde{\sigma}$-activated network within an arbitrary error on any pre-specified interval as presented in the following lemma.

Lemma 5.4. *For any $\varepsilon > 0$ and any $M > 0$, there exists a function ϕ realized by a $\widetilde{\sigma}$-activated network with width 29 and depth 6 such that*

$$|\phi(x) - \sigma(x)| < \varepsilon \quad \text{for any } x \in [-M, M].$$

The proof of Lemma 5.4 can be found later. By assuming Lemma 5.4 is true, we can give the proof of Theorem 5.3.

Proof of Theorem 5.3. By Theorem 1.1, there exists a function ϕ_σ generated by a σ-activated network with width $36d(2d+1)$ and depth 11 such that

$$|\phi_\sigma(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon/2 \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

By Lemma 5.4, for any $M > 0$ and each $\delta \in (0,1)$, there exists a function σ_δ realized by a $\widetilde{\sigma}$-activated network with width 29 and depth 6 such that

$$\sigma_\delta(t) \rightrightarrows \sigma(t) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } t \in [-M, M].$$

Then, by Lemma 5.1, there exists another function $\phi = \phi_{\widetilde{\sigma}}$ realized by a $\widetilde{\sigma}$-activated network with width $29 \times 36d(2d+1) = 1044d(2d+1)$ and depth $6 \times 11 = 66$ such that

$$|\phi(\boldsymbol{x}) - f(\boldsymbol{x})| < \varepsilon \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

So we finish the proof. □

Finally, let us present the detailed proof of Lemma 5.4.

Proof of Lemma 5.4. Since $1 = \widetilde{\sigma}'(0) = \lim_{x\to 0} \frac{\widetilde{\sigma}(x)}{x}$, it is easy to show: For any $\mathscr{E} > 0$ and any $R > 0$, there exists a sufficiently small $w > 0$ such that

$$\left\|\widetilde{\sigma}(wx)/w - x\right\|_{L^\infty([-R,R])} < \mathscr{E}.$$

Thus, we may assume the identity map is allowed to be the activation function in $\widetilde{\sigma}$-activated networks. Without loss of generality, we may assume $M \geq 2$ because $\widehat{M} = \max\{2, M\}$ implies $[-M, M] \subseteq [-\widehat{M}, \widehat{M}]$.

For simplicity, we denote $\mathscr{H}_{\widetilde{\sigma}}(N, L)$ as the (hypothesis) space of functions generated by $\widetilde{\sigma}$-activated networks with width N and depth L. Then the proof can be roughly divided into three steps as follows.

(1) Design $\Gamma \in \mathscr{H}_{\widetilde{\sigma}}(9,2)$ to reproduce xy on $[-4\widetilde{M}, 4\widetilde{M}]^2$, where $\widetilde{M} = (M+1)^2$.

(2) Design $\psi_\delta \in \mathscr{H}_{\widetilde{\sigma}}(20,4)$ based on the first step to approximate σ well on $[0, M]$.

(3) Design $\phi \in \mathscr{H}_{\widetilde{\sigma}}(29,6)$ based on the previous two steps to approximate σ well on $[-M, M]$.

The details of the three steps can be found below.

Step 1: Design $\Gamma \in \mathscr{H}_{\widetilde{\sigma}}(9,2)$ to reproduce xy on $[-4\widetilde{M}, 4\widetilde{M}]^2$.

Observe that

$$\widetilde{\sigma}(y) + 1 = \frac{y}{|y|+1} + 1 = \frac{y}{-y+1} + 1 = \frac{1}{-y+1} \quad \text{for any } y \leq 0.$$

For any $x \in [-4, 4]$, we have $-x-4 \leq 0$ and $-x-5 \leq 0$, implying

$$\begin{aligned}\widetilde{\sigma}(-x-4) - \widetilde{\sigma}(-x-5) &= \Big(\widetilde{\sigma}(-x-4) + 1\Big) - \Big(\widetilde{\sigma}(-x-5) + 1\Big) \\ &= \frac{1}{-(-x-4)+1} - \frac{1}{-(-x-5)+1} = \frac{1}{x+5} - \frac{1}{x+6} = \frac{1}{(x+5)(x+6)}.\end{aligned}$$

It follows from $1-\frac{90}{(x+5)(x+6)}\le 0$ for any $x\in[-4,4]$ that

$$\widetilde{\sigma}\Big(1-\frac{90}{(x+5)(x+6)}\Big)+1=\frac{1}{-\big(1-\frac{90}{(x+5)(x+6)}\big)+1}=\frac{x^2+11x+30}{90},$$

implying

$$\begin{aligned} x^2 &= 90\widetilde{\sigma}\Big(1-\frac{90}{(x+5)(x+6)}\Big)+90-(11x+30)\\ &= 90\widetilde{\sigma}\Big(1-90\big(\widetilde{\sigma}(-x-4)-\widetilde{\sigma}(-x-5)\big)\Big)-11x+60 \\ &= 90\widetilde{\sigma}\Big(1-90\widetilde{\sigma}(-x-4)+90\widetilde{\sigma}(-x-5)\Big)-11x+60. \end{aligned} \tag{5.2}$$

Thus, x^2 can be realized by a $\widetilde{\sigma}$-activated network with width 3 and depth 2 on $[-4,4]$. Set $\widetilde{M}=(M+1)^2$. Then, for any $x,y\in[-4\widetilde{M},4\widetilde{M}]$, we have $\frac{x}{2\widetilde{M}},\frac{y}{2\widetilde{M}},\frac{x+y}{2\widetilde{M}}\in[-4,4]$. Recall the fact

$$xy=2\widetilde{M}^2\Big((\tfrac{x+y}{2\widetilde{M}})^2-(\tfrac{x}{2\widetilde{M}})^2-(\tfrac{y}{2\widetilde{M}})^2\Big).$$

Thus, xy can be realized by a $\widetilde{\sigma}$-activated network with width 9 and depth 2 for any $x,y\in[-4\widetilde{M},4\widetilde{M}]$. That is, there exists $\Gamma\in\mathscr{H}_{\widetilde{\sigma}}(9,2)$ such that $\Gamma(x,y)=xy$ on $[-4\widetilde{M},4\widetilde{M}]^2$.

Step 2: Design $\psi_\delta\in\mathscr{H}_{\widetilde{\sigma}}(9,4)$ to approximate σ well on $[0,M]$.

Recall that x^2 can be realized by a $\widetilde{\sigma}$-activated network with width 3 and depth 2 on $[-4,4]$. There exists $\psi_1\in\mathscr{H}_{\widetilde{\sigma}}(3,2)$ such that

$$\psi_1(x)=\frac{(2x+1)^2}{(2M+1)^2}\quad\text{for any } x\in[-M,M].$$

Define

$$\psi_{2,\delta}(x)\coloneqq\frac{\widetilde{\sigma}(x+\delta)-\widetilde{\sigma}(x)}{\delta}\quad\text{for any } x\in\mathbb{R}.$$

Then, we have $\psi_{2,\delta}\in\mathscr{H}_{\widetilde{\sigma}}(2,1)$ and

$$\psi_{2,\delta}(x)\coloneqq\frac{\widetilde{\sigma}(x+\delta)-\widetilde{\sigma}(x)}{\delta}\rightrightarrows\frac{\mathrm{d}}{\mathrm{d}x}\widetilde{\sigma}(x)=\frac{c\sigma(x)+1}{(2x+1)^2}\quad\text{as}\quad\delta\to0^+,$$

for any $x\in[0,M]$ and

$$c=\frac{1}{2\int_0^\infty\frac{\sigma(t)}{(2t+1)^2}\mathrm{d}t}\approx 2.554.$$

Define

$$\psi_\delta(x)\coloneqq\tfrac{(2M+1)^2}{c}\Gamma\Big(\psi_1(x),\psi_{2,\delta}(x)\Big)-\tfrac{1}{c}\quad\text{for any } x\in\mathbb{R}.$$

Since $\Gamma\in\mathscr{H}_{\widetilde{\sigma}}(9,2)$, $\psi_1\in\mathscr{H}_{\widetilde{\sigma}}(3,2)$, and $\psi_{2,\delta}\in\mathscr{H}_{\widetilde{\sigma}}(2,1)$, we have $\psi_\delta\in\mathscr{H}_{\widetilde{\sigma}}(9,4)$.

Clearly, for any $x\in[0,M]$, we have $\psi_1(x)=\frac{(x+1)^2}{(2M+1)^2}\in[0,1]$ and $\psi_{2,\delta}(x)\approx\frac{c\sigma(x)+1}{(2x+1)^2}\in[0,3.6]$, implying $\psi_1(x),\psi_{2,\delta}(x)\in[-4,4]\subseteq[-4\widetilde{M},4\widetilde{M}]^2$ for any small $\delta>0$. Thus, for any $x\in[0,M]$, as δ goes to 0^+, we get

$$\begin{aligned}\psi_\delta(x)&=\tfrac{(2M+1)^2}{c}\Gamma\Big(\psi_1(x),\psi_{2,\delta}(x)\Big)-\tfrac{1}{c}=\tfrac{(2M+1)^2}{c}\cdot\psi_1(x)\cdot\psi_{2,\delta}(x)-\tfrac{1}{c}\\ &\rightrightarrows\tfrac{(2M+1)^2}{c}\cdot\tfrac{(2x+1)^2}{(2M+1)^2}\cdot\tfrac{c\sigma(x)+1}{(2x+1)^2}-\tfrac{1}{c}=\sigma(x).\end{aligned}$$

That is, for any $x \in [0, M]$,

$$\psi_\delta(x) \rightrightarrows \sigma(x) \quad \text{as} \quad \delta \to 0^+.$$

Step 3: Design $\phi \in \mathscr{H}_{\widetilde{\sigma}}(29, 6)$ to approximate σ well on $[-M, M]$.

Note that $\widetilde{\sigma}(x) = \sigma(x)$ for all $x \in [-M, 0)$ and $\psi_\delta(x)$ approximates $\sigma(x)$ well for all $x \in [0, M]$. Then, $\widetilde{\sigma}(x) \cdot \mathbb{1}_{\{x \in [-M,0)\}} + \psi_\delta(x) \cdot \mathbb{1}_{\{x \in [0,M]\}}$ approximates $\sigma(x)$ well for all $x \in [-M, M]$. To design ϕ approximating σ well on $[-M, M]$, we need to design a $\widetilde{\sigma}$-activated network to approximate an indicator function $\mathbb{1}_{\{x \in [0,M]\}}$ well.

It is impossible to approximate $\mathbb{1}_{\{x \in [0,M]\}}$ well by a $\widetilde{\sigma}$-activated network due to the continuity of $\widetilde{\sigma}$. However, we define a continuous function g to replace $\mathbb{1}_{\{x \in [0,M]\}}$. By the continuity of $\widetilde{\sigma}$ and σ, there exists $\eta_0 \in (0, 1)$ such that

$$|\widetilde{\sigma}(x)| < \varepsilon/6 \quad \text{and} \quad |\sigma(x)| < \varepsilon/6 \quad \text{for any } x \in [0, \eta_0]. \tag{5.3}$$

Then we define

$$g(x) \coloneqq \frac{\mathrm{ReLU}(x) - \mathrm{ReLU}(x - \eta_0)}{\eta_0}, \quad \text{where } \mathrm{ReLU}(x) = \max\{0, x\} \quad \text{for any } x \in \mathbb{R}.$$

See Figure 16 for an illustration of g.

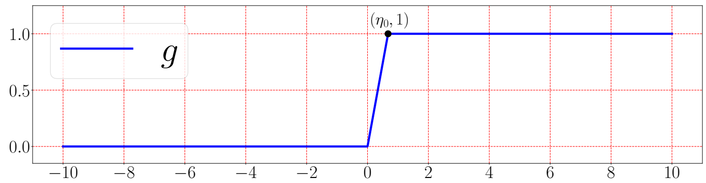


Figure 16: An illustration of g on $[-10, 10]$.

We will construct a $\widetilde{\sigma}$-activated network to approximate g well. To this end, we first design a $\widetilde{\sigma}$-activated network to approximate the ReLU function well. For $x \in [-M-1, M+1]$, we have $\frac{x}{M+1} + 1 \in [0, 2] \subseteq [0, M]$, implying

$$1 - \psi_\delta\big(\tfrac{x}{M+1} + 1\big) \rightrightarrows 1 - \sigma\big(\tfrac{x}{M+1} + 1\big) = \big|\tfrac{x}{M+1}\big| \quad \text{as} \quad \delta \to 0^+,$$

where the last equality comes from $1 - \sigma(y) = |y - 1|$ for any $y \in [0, 2]$. Note that $\mathrm{ReLU}(x) = \frac{x}{2} + \frac{|x|}{2} = \frac{x}{2} + \frac{M+1}{2} \cdot \big|\frac{x}{M+1}\big|$ for any $x \in [-M-1, M+1]$. Define

$$\widetilde{g}_\delta(x) \coloneqq \tfrac{x}{2} + \tfrac{M+1}{2}\Big(1 - \psi_\delta\big(\tfrac{x}{M+1} + 1\big)\Big) \quad \text{for any } x \in \mathbb{R}.$$

Then, $\psi_\delta \in \mathscr{H}_{\widetilde{\sigma}}(9, 4)$ implies $\widetilde{g}_\delta \in \mathscr{H}_{\widetilde{\sigma}}(10, 4)$. Moreover, for any $x \in [-M-1, M+1]$,

$$\widetilde{g}_\delta(x) \rightrightarrows \tfrac{x}{2} + \tfrac{M+1}{2} \cdot \big|\tfrac{x}{M+1}\big| = \mathrm{ReLU}(x) \quad \text{as} \quad \delta \to 0^+.$$

Define

$$g_\delta(x) \coloneqq \frac{\widetilde{g}_\delta(x) - \widetilde{g}_\delta(x - \eta_0)}{\eta_0} \quad \text{for any } x \in \mathbb{R}.$$

Clearly, $\widetilde{g}_\delta \in \mathscr{H}_{\widetilde{\sigma}}(10,4)$ implies $g_\delta \in \mathscr{H}_{\widetilde{\sigma}}(20,4)$. For any $x \in [-M, M]$, we have $x, x-\eta_0 \in [-M-1, M+1]$, implying

$$g_\delta(x) = \frac{\widetilde{g}_\delta(x) - \widetilde{g}_\delta(x-\eta_0)}{\eta_0} \rightrightarrows \frac{\mathrm{ReLU}(x) - \mathrm{ReLU}(x-\eta_0)}{\eta_0} = g(x) \quad \text{as} \quad \delta \to 0^+.$$

Next, define

$$\phi_\delta(x) \coloneqq \Gamma\Big(\psi_\delta(x), g_\delta(x)\Big) + \Gamma\Big(\widetilde{\sigma}(x), 1 - g_\delta(x)\Big) \quad \text{for any } x \in \mathbb{R}.$$

Since $\Gamma \in \mathscr{H}_{\widetilde{\sigma}}(9,2)$, $\psi_\delta \in \mathscr{H}_{\widetilde{\sigma}}(9,4)$, and $g_\delta \in \mathscr{H}_{\widetilde{\sigma}}(20,4)$, we have $\phi_\delta \in \mathscr{H}_{\widetilde{\sigma}}(29,6)$.

Clearly, $\widetilde{\sigma}(x)$, $g_\delta(x)$, and $1-g_\delta(x)$ are all in $[-4\widetilde{M}, 4\widetilde{M}]$ for any small $\delta > 0$ and all $x \in [-M, M]$. We will show $\psi_\delta(x) \in [-4\widetilde{M}, 4\widetilde{M}]$ for any small $\delta > 0$ and all $x \in [-M, M]$ via two cases as follows.

- For $x \in [0, M]$, $\psi_\delta(x) \rightrightarrows \sigma(x)$ implies $\psi_\delta(x) \in [-4\widetilde{M}, 4\widetilde{M}]$ for any small $\delta > 0$.
- For $x \in [-M, 0)$, we have $\psi_1(x) = \frac{(x+1)^2}{(2M+1)^2} \in [0,1]$ and

  $$\psi_{2,\delta}(x) = \tfrac{\widetilde{\sigma}(x+\delta) - \widetilde{\sigma}(x)}{\delta} \rightrightarrows \tfrac{\mathrm{d}}{\mathrm{d}x}\widetilde{\sigma}(x) = \tfrac{1}{(-x+1)^2} \quad \text{as} \quad \delta \to 0^+.$$

  Thus, for any $x \in [-M, 0)$, as δ goes to 0^+, we get

  $$\begin{aligned}\psi_\delta(x) &= \tfrac{(2M+1)^2}{c}\Gamma\Big(\psi_1(x), \psi_{2,\delta}(x)\Big) - \tfrac{1}{c} = \tfrac{(2M+1)^2}{c} \cdot \psi_1(x) \cdot \psi_{2,\delta}(x) - \tfrac{1}{c} \\ &\rightrightarrows \tfrac{(2M+1)^2}{c} \cdot \tfrac{(2x+1)^2}{(2M+1)^2} \cdot \tfrac{1}{(-x+1)^2} - \tfrac{1}{c} = \tfrac{(2x+1)^2 - 1}{c(-x+1)^2}.\end{aligned}$$

  Since $\widetilde{M} = (M+1)^2$, we have $\frac{(2x+1)^2-1}{c(-x+1)^2} \in [0, 4\widetilde{M} - 1]$ for all $x \in [-M, 0)$, implying $\psi_\delta(x) \in [-4\widetilde{M}, 4\widetilde{M}]$ for any small $\delta > 0$.

Thus, for any $x \in [\eta_0, M]$, we have $1 - g(x) = 0$, implying

$$\phi_\delta(x) = \psi_\delta(x) \cdot g_\delta(x) + \widetilde{\sigma}(x) \cdot \big(1 - g_\delta(x)\big) \rightrightarrows \sigma(x) \cdot g(x) + 0 = \sigma(x) \quad \text{as} \quad \delta \to 0^+.$$

Similarly, for any $x \in [-M, 0]$, we have $g(x) = 0$, implying

$$\phi_\delta(x) = \psi_\delta(x) \cdot g_\delta(x) + \widetilde{\sigma}(x) \cdot \big(1 - g_\delta(x)\big) \rightrightarrows 0 + \widetilde{\sigma}(x) \cdot \big(1 - g(x)\big) = \sigma(x) \quad \text{as} \quad \delta \to 0^+.$$

Therefore, there exists a small $\delta_0 > 0$ such that

$$|\phi_{\delta_0}(x) - \sigma(x)| < \varepsilon \quad \text{for any } x \in [-M, 0] \bigcup [\eta_0, M],$$

$\|g_{\delta_0}\|_{L^\infty([0,\eta_0])} \le 2, \quad \|1 - g_{\delta_0}\|_{L^\infty([0,\eta_0])} \le 2, \quad$ and

$$\|\psi_{\delta_0}\|_{L^\infty([0,\eta_0])} \le \|\sigma\|_{L^\infty([0,\eta_0])} + \varepsilon/12,$$

where the above inequality comes from $\psi_\delta(x)$ uniformly converges to $\sigma(x)$ for any $x \in [0, \eta_0] \subseteq [0, M]$.

Clearly, for $x \in [0, \eta_0]$, by Equation (5.3), we have

$$\begin{aligned}
|\phi_{\delta_0}(x) - \sigma(x)| \le |\phi_{\delta_0}(x)| + |\sigma(x)| &< \left|\psi_{\delta_0}(x) \cdot g_{\delta_0}(x) + \widetilde{\sigma}(x) \cdot \big(1 - g_{\delta_0}(x)\big)\right| + \varepsilon/6 \\
&\le \left|\psi_{\delta_0}(x)\right| \cdot \left|g_{\delta_0}(x)\right| + \left|\widetilde{\sigma}(x)\right| \cdot \left|1 - g_{\delta_0}(x)\right| + \varepsilon/6 \\
&\le \Big(\|\sigma\|_{L^\infty([0,\eta_0])} + \frac{\varepsilon}{12}\Big) \cdot 2 + \frac{\varepsilon}{6} \cdot 2 + \frac{\varepsilon}{6} \\
&\le \Big(\frac{\varepsilon}{6} + \frac{\varepsilon}{12}\Big) \cdot 2 + \frac{\varepsilon}{6} \cdot 2 + \frac{\varepsilon}{6} = \varepsilon.
\end{aligned}$$

By setting $\phi = \phi_{\delta_0}$, we have $\phi = \phi_{\delta_0} \in \mathscr{H}_{\widetilde{\sigma}}(29, 6)$ and

$$|\phi(x) - \sigma(x)| = |\phi_{\delta_0}(x) - \sigma(x)| < \varepsilon \quad \text{for any } x \in [-M, M].$$

So we finish the proof. □

5.3 Proof of Lemma 5.1

Let the activation function be applied to a vector elementwisely. Then, ϕ_ϱ can be represented in a form of function compositions as follows:

$$\phi_\varrho(\boldsymbol{x}) = \mathcal{L}_L \circ \varrho \circ \mathcal{L}_{L-1} \circ \varrho \circ \ \cdots \ \circ \varrho \circ \mathcal{L}_1 \circ \varrho \circ \mathcal{L}_0(\boldsymbol{x}) \quad \text{for any } \boldsymbol{x} \in \mathbb{R}^d,$$

where $N_0 = d$, $N_1, N_2, \cdots, N_L \in \mathbb{N}^+$, $N_{L+1} = 1$, $\boldsymbol{A}_\ell \in \mathbb{R}^{N_{\ell+1} \times N_\ell}$ and $\boldsymbol{b}_\ell \in \mathbb{R}^{N_{\ell+1}}$ are the weight matrix and the bias vector in the ℓ-th affine linear transform $\mathcal{L}_\ell : \boldsymbol{y} \mapsto \boldsymbol{A}_\ell \boldsymbol{y} + \boldsymbol{b}_\ell$ for each $\ell \in \{0, 1, \cdots, L\}$. Define

$$\phi_{\varrho_\delta}(\boldsymbol{x}) \coloneqq \mathcal{L}_L \circ \varrho_\delta \circ \mathcal{L}_{L-1} \circ \varrho_\delta \circ \ \cdots \ \circ \varrho_\delta \circ \mathcal{L}_1 \circ \varrho_\delta \circ \mathcal{L}_0(\boldsymbol{x}) \quad \text{for any } \boldsymbol{x} \in \mathbb{R}^d.$$

Recall that ϱ_δ can be realized by a $\widetilde{\varrho}$-activated network with width $\widetilde{N}$ and depth $\widetilde{L}$. Thus, ϕ_{ϱ_δ} can be realized by a $\widetilde{\varrho}$-activated network with width $N\widetilde{N}$ and depth $L\widetilde{L}$.

We will prove

$$\phi_{\varrho_\delta}(\boldsymbol{x}) \rightrightarrows \phi_\varrho(\boldsymbol{x}) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } \boldsymbol{x} \in [a, b]^d.$$

For any $\boldsymbol{x} \in \mathbb{R}^d$ and each $\ell \in \{1, 2, \cdots, L+1\}$, define

$$\boldsymbol{h}_\ell(\boldsymbol{x}) \coloneqq \mathcal{L}_{\ell-1} \circ \varrho \circ \mathcal{L}_{\ell-2} \circ \varrho \circ \ \cdots \ \circ \varrho \circ \mathcal{L}_1 \circ \varrho \circ \mathcal{L}_0(\boldsymbol{x})$$

and

$$\boldsymbol{h}_{\ell,\delta}(\boldsymbol{x}) \coloneqq \mathcal{L}_{\ell-1} \circ \varrho_\delta \circ \mathcal{L}_{\ell-2} \circ \varrho_\delta \circ \ \cdots \ \circ \varrho_\delta \circ \mathcal{L}_1 \circ \varrho_\delta \circ \mathcal{L}_0(\boldsymbol{x}).$$

Note that $\boldsymbol{h}_\ell$ and $\boldsymbol{h}_{\ell,\delta}$ are two maps from $\mathbb{R}^d$ to $\mathbb{R}^{N_\ell}$ for each ℓ.

We will prove by induction that

$$\boldsymbol{h}_{\ell,\delta}(\boldsymbol{x}) \rightrightarrows \boldsymbol{h}_\ell(\boldsymbol{x}) \quad \text{as} \quad \delta \to 0^+ \tag{5.4}$$

for any $\boldsymbol{x} \in [a, b]^d$ and each $\ell \in \{1, 2, \cdots, L+1\}$.

First, we consider the case $\ell = 1$. Clearly,

$$\boldsymbol{h}_{1,\delta}(\boldsymbol{x}) = \mathcal{L}_0(\boldsymbol{x}) = \boldsymbol{h}_1(\boldsymbol{x}) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } \boldsymbol{x} \in [a, b]^d.$$

This means Equation (5.4) holds for $\ell = 1$.

Next, suppose Equation (5.4) holds for $\ell = i \in \{1, 2, \cdots, L\}$. Our goal is to prove that it also holds for $\ell = i + 1$. Define

$$M \coloneqq \sup\Big\{\|\boldsymbol{h}_j(\boldsymbol{x})\|_{\ell^\infty} + 1 : \boldsymbol{x} \in [a,b]^d, \quad j = 1, 2, \cdots, L+1\Big\},$$

where the continuity of ϱ guarantees the above supremum is finite. By the induction hypothesis, we have

$$\boldsymbol{h}_{i,\delta}(\boldsymbol{x}) \rightrightarrows \boldsymbol{h}_i(\boldsymbol{x}) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Clearly, for any $\boldsymbol{x} \in [a,b]^d$, we have $\|\boldsymbol{h}_i(\boldsymbol{x})\|_{\ell^\infty} \le M$ and $\|\boldsymbol{h}_{i,\delta}(\boldsymbol{x})\|_{\ell^\infty} \le \|\boldsymbol{h}_i(\boldsymbol{x})\|_{\ell^\infty} + 1 \le M$ for any small $\delta > 0$.

Recall the fact $\varrho_\delta(t) \rightrightarrows \varrho(t)$ as $\delta \to 0^+$ for any $t \in [-M, M]$. Then

$$\varrho_\delta \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) - \varrho \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) \rightrightarrows \boldsymbol{0} \quad \text{as} \quad \delta \to 0^+.$$

The continuity of ϱ implies the uniform continuity of ϱ on $[-M, M]$, deducing

$$\varrho \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) - \varrho \circ \boldsymbol{h}_i(\boldsymbol{x}) \rightrightarrows \boldsymbol{0} \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

Therefore, for any $\boldsymbol{x} \in [a,b]^d$, as $\delta \to 0^+$, we have

$$\varrho_\delta \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) - \varrho \circ \boldsymbol{h}_i(\boldsymbol{x}) = \underbrace{\varrho_\delta \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) - \varrho \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x})}_{\rightrightarrows \boldsymbol{0}} + \underbrace{\varrho \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) - \varrho \circ \boldsymbol{h}_i(\boldsymbol{x})}_{\rightrightarrows \boldsymbol{0}} \rightrightarrows \boldsymbol{0},$$

implying

$$\boldsymbol{h}_{i+1,\delta}(\boldsymbol{x}) = \mathcal{L}_i \circ \varrho_\delta \circ \boldsymbol{h}_{i,\delta}(\boldsymbol{x}) \rightrightarrows \mathcal{L}_i \circ \varrho \circ \boldsymbol{h}_i(\boldsymbol{x}) = \boldsymbol{h}_{i+1}(\boldsymbol{x}).$$

This means Equation (5.4) holds for $\ell = i + 1$. So we complete the inductive step.

By the principle of induction, we have

$$\phi_{\varrho_\delta}(\boldsymbol{x}) = \boldsymbol{h}_{L+1,\delta}(\boldsymbol{x}) \rightrightarrows \boldsymbol{h}_{L+1}(\boldsymbol{x}) = \phi_\varrho(\boldsymbol{x}) \quad \text{as} \quad \delta \to 0^+ \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

There exists a small $\delta_0 > 0$ such that

$$\big|\phi_{\varrho_{\delta_0}}(\boldsymbol{x}) - \phi_\varrho(\boldsymbol{x})\big| < \varepsilon/2 \quad \text{for any } \boldsymbol{x} \in [a,b]^d.$$

By setting $\phi = \phi_{\varrho_{\delta_0}}$, we have

$$\big|\phi(\boldsymbol{x}) - f(\boldsymbol{x})\big| \le \big|\phi_{\varrho_{\delta_0}}(\boldsymbol{x}) - \phi_\varrho(\boldsymbol{x})\big| + \big|\phi_\varrho(\boldsymbol{x}) - f(\boldsymbol{x})\big| < \varepsilon/2 + \varepsilon/2 = \varepsilon$$

for any $\boldsymbol{x} \in [a,b]^d$. Moreover, $\phi = \phi_{\varrho_{\delta_0}}$ can be generated by a $\widetilde{\varrho}$-activated network with width $N\widetilde{N}$ and depth $L\widetilde{L}$. So we finish the proof.

6 Conclusion

This paper studies the super approximation power of deep feed-forward neural networks with a fixed size. It is proved by construction that there exists an EUAF network architecture with d input neurons, a maximum width $36d(2d+1)$, 11 hidden layers, and at most $5437(d+1)(2d+1)$ nonzero parameters, achieving the universal approximation property by only adjusting its finitely many parameters. That is, without changing the network size, our EUAF network can approximate any continuous function $f:[a,b]^d \to \mathbb{R}$ within an arbitrarily small error $\varepsilon > 0$ with appropriate parameters depending on f, ε, d, a, and b. Moreover, augmenting this EUAF network using one more layer with 2 neurons can exactly realize a classification function $\sum_{j=1}^{J} r_j \cdot \mathbb{1}_{E_j}$ in $\bigcup_{j=1}^{J} E_j$ for any $J \in \mathbb{N}^+$, where $r_1, r_2, \cdots, r_J$ are distinct rational numbers, $\mathbb{1}_{E_j}$ is the indicator function of E_j for each j, and $E_1, E_2, \cdots, E_J$ are arbitrary pairwise disjoint closed bounded subsets of $\mathbb{R}^d$. While we are interested in the theoretical analysis here, it is interesting to explore the numerical implementation in various applications of the proposed EUAF neural network. Furthermore, it would be very interesting to investigate the generalization and optimization errors of the EUAF networks in deep learning.

Acknowledgments

Z. Shen is supported by Tan Chin Tuan Centennial Professorship. H. Yang was partially supported by the US National Science Foundation under award DMS-1945029. S. Zhang is supported by a Postdoctoral Fellowship under NUS ENDOWMENT FUND (EXP WBS) (01 651).

References

[1] A. R. Barron, *Universal approximation bounds for superpositions of a sigmoidal function*, IEEE Transactions on Information Theory, 39 (1993), pp. 930–945.

[2] A. D. Beck, J. Zeifang, A. Schwarz, and D. G. Flad, *A neural network based shock detection and localization approach for discontinuous galerkin methods*, Journal of Computational Physics, 423 (2020), p. 109824.

[3] C. Beck, M. Hutzenthaler, A. Jentzen, and B. Kuckuck, *An overview on deep learning-based approximation methods for partial differential equations*, arxiv:2012.12348, (2021).

[4] D. E. Bernholdt, M. R. Cianciosa, D. L. Green, J. M. Park, K. J. H. Law, and C. Etienam, *Cluster, classify, regress: A general method for learning discontinuous functions*, Foundations of Data Science, 1 (2019), pp. 491–506.

[5] H. Bölcskei, P. Grohs, G. Kutyniok, and P. Petersen, *Optimal approximation with sparsely connected deep neural networks*, SIAM Journal on Mathematics of Data Science, 1 (2019), pp. 8–45.

[6] A. Bonito, R. DeVore, P. J. Diane Guignard, and G. Petrova, *Polynomial approximation of anisotropic analytic functions of several variables*, Constructive Approximation, 53 (2021), pp. 319–348.

[7] L. Chen and C. Wu, *A note on the expressive power of deep rectified linear unit networks in high-dimensional spaces*, Mathematical Methods in the Applied Sciences, 42 (2019), pp. 3400–3404.

[8] A. Cohen, R. DeVore, G. Petrova, and P. Wojtaszczyk, *Optimal stable nonlinear approximation*, arXiv e-prints, (2020), p. arXiv:2009.09907.

[9] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, MCSS, 2 (1989), pp. 303–314.

[10] I. Daubechies, R. DeVore, S. Foucart, B. Hanin, and G. Petrova, *Nonlinear approximation and (deep)* ReLU *networks*, Constructive Approximation, (2021).

[11] R. A. DeVore, *Nonlinear approximation*, Acta Numerica, 7 (1998), pp. 51–150.

[12] W. E, C. Ma, and Q. Wang, *A priori estimates of the population risk for residual networks*, arXiv e-prints, (2019), p. arXiv:1903.02154.

[13] W. E, C. Ma, and L. Wu, *A priori estimates of the population risk for two-layer neural networks*, Communications in Mathematical Sciences, 17 (2019), pp. 1407–1425.

[14] W. E and Q. Wang, *Exponential convergence of the deep neural network approximation for analytic functions*, CoRR, abs/1807.00297 (2018).

[15] W. E and S. Wojtowytsch, *A priori estimates for classification problems using neural networks*, arXiv e-prints, (2020), p. arXiv:2009.13500.

[16] ——, *Representation formulas and pointwise properties for Barron functions*, arXiv e-prints, (2020), p. arXiv:2006.05982.

[17] D. Elbrächter, P. Grohs, A. Jentzen, and C. Schwab, *Dnn expression rate analysis of high-dimensional pdes: Application to option pricing*, Constructive Approximation, (2021).

[18] J. Gedeon, J. Schmidt, M. J. Hodgson, J. Wetherell, C. L. Benavides-Riveros, and M. A. L. Marques, *Machine learning the derivative discontinuity of density-functional theory*, Machine Learning: Science and Technology, (2021).

[19] V. Gupta, S. K. Gupta, and J. Kim, *Automated discontinuity detection and reconstruction in subsurface environment of mars using deep learning: A case study of sharad observation*, Applied Sciences, 10 (2020).

[20] J. Han and C. Moraga, *The influence of the sigmoid function parameters on the speed of backpropagation learning*, in From Natural to Artificial Neural Computation, J. Mira and F. Sandoval, eds., Berlin, Heidelberg, 1995, Springer Berlin Heidelberg, pp. 195–201.

[21] J. He, X. Jia, J. Xu, L. Zhang, and L. Zhao, *Make ℓ_1 regularization effective in training sparse CNN*, Computational Optimization and Applications, 77 (2020), pp. 163–182.

[22] S. Hon and H. Yang, *Simultaneous neural network approximations in Sobolev spaces*, arXiv e-prints, (2021), p. arXiv:2109.00161.

[23] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, Neural Networks, 4 (1991), pp. 251–257.

[24] K. Hornik, M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks, 2 (1989), pp. 359–366.

[25] W.-F. Hu, T.-S. Lin, and M.-C. Lai, *A Discontinuity Capturing Shallow Neural Network for Elliptic Interface Problems*, arXiv e-prints, (2021), p. arXiv:2106.05587.

[26] Y. Jiao, Y. Lai, X. Lu, and J. Z. Yang, *Deep neural networks with relu-sine-exponential activations break curse of dimensionality on hölder class*, CoRR, abs/2103.00542 (2021).

[27] K. Kawaguchi, *Deep learning without poor local minima*, in Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds., Curran Associates, Inc., 2016, pp. 586–594.

[28] K. Kawaguchi and Y. Bengio, *Depth with nonlinearity creates no bad local minima in resnets*, Neural Networks, 118 (2019), pp. 167–174.

[29] A. N. Kolmogorov, *On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition*, Doklady Akademii Nauk SSSR, 114 (1957), pp. 953–956.

[30] P. Le and W. Zuidema, *Compositional distributional semantics with long short term memory*, in Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics, Denver, Colorado, June 2015, Association for Computational Linguistics, pp. 10–19.

[31] Q. Li, T. Lin, and Z. Shen, *Deep learning via dynamical systems: An approximation perspective*, Journal of European Mathematical Society, (to appear).

[32] Q. Li, C. Tai, and W. E, *Stochastic modified equations and dynamics of stochastic gradient algorithms I: Mathematical foundations*, Journal of Machine Learning Research, 20 (2019), pp. 1–47.

[33] H. Lin and S. Jegelka, *Resnet with one-neuron hidden layers is a universal approximator*, in Advances in Neural Information Processing Systems, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds., vol. 31, Curran Associates, Inc., 2018.

[34] J. Lu, Z. Shen, H. Yang, and S. Zhang, *Deep network approximation for smooth functions*, SIAM Journal on Mathematical Analysis, 53 (2021), pp. 5465–5506.

[35] V. Maiorov and A. Pinkus, *Lower bounds for approximation by MLP neural networks*, Neurocomputing, 25 (1999), pp. 81–91.

[36] H. Montanelli, H. Yang, and Q. Du, *Deep ReLU networks overcome the curse of dimensionality for bandlimited functions*, Journal of Computational Mathematics, (to appear).

[37] B. Neyshabur, Z. Li, S. Bhojanapalli, Y. LeCun, and N. Srebro, *The role of over-parametrization in generalization of neural networks*, in International Conference on Learning Representations, 2019.

[38] Q. N. Nguyen and M. Hein, *The loss surface of deep and wide neural networks*, CoRR, abs/1704.08045 (2017).

[39] P. Petersen and F. Voigtlaender, *Optimal approximation of piecewise smooth functions using deep ReLU neural networks*, Neural Networks, 108 (2018), pp. 296–330.

[40] Z. Shen, H. Yang, and S. Zhang, *Deep network approximation characterized by number of neurons*, Communications in Computational Physics, 28 (2020), pp. 1768–1811.

[41] ——, *Deep network with approximation error being reciprocal of width to power of square root of depth*, Neural Computation, 33 (2021), pp. 1005–1036.

[42] ——, *Neural network approximation: Three hidden layers are enough*, Neural Networks, 141 (2021), pp. 160–173.

[43] ——, *Optimal approximation rate of ReLU networks in terms of width and depth*, Journal de Mathématiques Pures et Appliquées, (to appear).

[44] J. W. Siegel and J. Xu, *Optimal approximation rates and metric entropy of ReLU^k and cosine networks*, arXiv e-prints, (2021), p. arXiv:2101.12365.

[45] E. M. Stein and R. Shakarchi, *Real Analysis: Measure Theory, Integration, and Hilbert Spaces*, Princeton University Press, Princeton, New Jersey, USA, 2005.

[46] J. Turian, J. Bergstra, and Y. Bengio, *Quadratic features and deep architectures for chunking*, in Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers, NAACL-Short '09, USA, 2009, Association for Computational Linguistics, pp. 245–248.

[47] Y. Yang and Y. Wang, *Approximation in shift-invariant spaces with deep ReLU neural networks*, arXiv e-prints, (2020), p. arXiv:2005.11949.

[48] D. Yarotsky, *Optimal approximation of continuous functions by very deep ReLU networks*, in Proceedings of the 31st Conference On Learning Theory, S. Bubeck, V. Perchet, and P. Rigollet, eds., vol. 75 of Proceedings of Machine Learning Research, PMLR, 06–09 Jul 2018, pp. 639–649.

[49] D. Yarotsky, *Elementary superexpressive activations*, in Proceedings of the 38th International Conference on Machine Learning, M. Meila and T. Zhang, eds., vol. 139 of Proceedings of Machine Learning Research, PMLR, 18–24 Jul 2021, pp. 11932–11940.

[50] D. Yarotsky and A. Zhevnerchuk, *The phase diagram of approximation rates for deep neural networks*, in Advances in Neural Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, eds., vol. 33, Curran Associates, Inc., 2020, pp. 13005–13015.

[51] S. Zhang, *Deep neural network approximation via function compositions*, PhD Thesis, National University of Singapore, (2020). URL https://scholarbank.nus.edu.sg/handle/10635/186064.