Multilayer perceptrons: Approximation order and necessary number of hidden units

Stephan Trenn

Abstract—**This paper considers the approximation of sufficiently smooth multivariable functions with a multilayer perceptron (MLP). For a given approximation order explicit formulas for the necessary number of hidden units and its distributions to the hidden layers of the MLP is derived. These formulas depend only on the number of input variables and on the desired approximation order. The concept of approximation order encompasses Kolmogorov-Gabor polynomials or discrete Volterra series which are widely used in static and dynamic models of nonlinear systems. The results are obtained by considering structural properties of the Taylor polynomials of the function in question and of the MLP function.**

Index Terms—**multilayer perceptron, approximation, necessary number of hidden units**

I. Introduction

The original motivation for artificial neural networks (ANNs) was modelling cognitive processes observed in animals and humans. Many applications of ANNs show that this approach was very useful, although it also became clear that some problems can not be solved with ANNs or could be solved better with other approaches. Nowadays there are lots of different types of ANNs and the connection to biological neural networks is very loose, if there is any at all. For a comprehensive overview over different kinds of neural networks see [1], where also the biological background and historical remarks are given. In this paper only the multilayer perceptron (MLP) is studied, which is very popular in the application area as well as in theoretical research. The reasons for this popularity might be

- its simplicity,
- its scalability,
- its property to be a general function approximator,
- and its adaptivity.

The MLP was primarily used for classification problems, but its capability to approximate functions made it also interesting for other applications. One of these applications is modelling and control, where ANNs, in particular MLPs, are successfully used (see, e.g., [2] and [3]).

When using ANNs in applications, there are two main questions:

- Is it theoretically possible to solve the task with the considered class of ANNs?
- How can one find an ANN which solves the task?

In general, ANNs are scalable, i.e. they can have different sizes, and they are adaptive, i.e. they have parameters which can be changed. In most cases the structure and size of an ANN are chosen a priori and afterwards the ANN “learns” a given task, which is nothing more than adjusting the parameters in a certain way. Therefore, the first question deals with the structure and size of the ANN and the second question targets the change of the parameters, i.e. the learning procedure.

The first question is strongly connected to two other questions:

- What size or structure is *necessary* to solve a given task?
- What size or structure is *sufficient* to solve a given task?

This paper gives an answer to the first of the last two questions for a specific task. It is an important question whether the necessary size is also sufficient, but an answer to this question is not in the scope of this paper. The question how to learn an ANN is also not in the scope of this paper.

The task which is considered here is to approximate any function, which is sufficiently smooth, with a given *approximation order*. One function approximates another function with a specific order if the function value and all derivatives up to the specific order coincide at one fixed point, i.e. the Taylor polynomials are the same. This kind of approximation plays an important role in control theory, where often a steady-state is considered and it is important that in a neighbourhood of this steady-state the function which approximates the system or the controller is very accurate. On the other hand, the accuracy far away from the steady-state does not play an important role. In systems theory it is a widely used method ([4]–[7]) to model nonlinear static and dynamic systems with multivariable polynomials of a certain degree or with truncated discrete Volterra series (these polynomials are also called Kolmogorov-Gabor polynomials). Clearly, this is a special case of the concept of approximation order. In fact, the widely used method of linearization is just an approximation with approximation order one. The question which will be answered in this paper is therefore:

Which size and structure is necessary for an MLP to approximate any sufficiently smooth function with a given approximation order?

There is a wide range of literature on the principle possibility of MLPs to approximate continuous function to any given accuracy, for a good overview see [8] and the references therein. There the focus is on global approximation accuracy, but the results are qualitative in nature, i.e. the formulas include unspecified constants and can therefore not be used directly to calculate the necessary number of hidden units. The same is true for the results in [9], where in addition

S. Trenn is with the Institute of Mathematics, Ilmenau University of Technology, 98684 Ilmenau, Germany (e-mail: stephan.trenn@tu-ilmenau.de)

the constants depend on the specific function which should be approximated. To the authors best knowledge there are no explicit formulas for the number of necessary hidden units available, where no quantities of the specific function are needed. The answer to the above question which is given in this paper (see Theorem 13) only depends on the desired approximation order and the number of inputs. Note that the calculated necessary number of hidden units is a worst case number, i.e. if the number of hidden units is smaller than the calculated number then there exists a function which can not be approximated with the desired approximation order. There are of course special function which can be approximated with less hidden units.

To find an answer to the above question a system of non-linear equation is considered. If this system is to "small" then it is not always solvable and the consequence is that the MLP cannot approximate all functions with a given approximation order. In this case it is not possible to achieve the desired approximation order for some functions, even if one has infinitely many exact data points of the functions.

Finally it should be mentioned that the results in this paper are of theoretical nature. In practical applications one perhaps has more information about the function which should be approximated and therefore better bounds for the size of the network can be calculated or estimated. Furthermore the data points might not be exact and therefore statistical approaches might yield better results. Hence, the results in this paper do not make classical methods like cross-validation or pruning (e.g. as in [10]) superfluous but might be an additional tool for finding the best size of the network. The main contribution of this paper is the theoretical analysis of an MLP as a special parametric function approximator; in particular, the focus is on structural questions and not on questions about practical ways of adjusting the parameters (i.e. training methods, choosing training data, etc.). It is also important to distinguish the results of this paper from results which are based on the analysis of the distribution of the sample data (e.g. as in [11]), because these approaches deal with classification problems and the concept of approximation order makes no sense for classification problems, which can be seen as the search for a global approximator of the corresponding non-smooth indicator function of the class.

This paper is structured as follows. In Section II the MLP model is briefly described and a formal definition is given. Section III deals with the concept of "approximation order" (Definition 3). In this context Taylor polynomials, analyticity, and the approximation accuracy of Taylor polynomial approximation are revisited. In particular a sufficient condition is given for which a high approximation order of an MLP implies good overall approximation accuracy (Proposition 6). Section IV gives a step-by-step derivation for the main results (Theorems 12 and 13) and gives some further interpreting remarks on these results. To improve readability all proofs of the results are put in the Appendix.

This section finishes with some remarks on notation. The natural and real numbers are denoted by $\mathbb{N}$ and $\mathbb{R}$, resp., $K = [-1,1]^{n_0} \subseteq \mathbb{R}^{n_0}$ is the compact n_0-dimensional unit cube for some $n_0 \in \mathbb{N}$, the latter is used to denote the number of relevant inputs. For $N \in \mathbb{N} \cup \{\infty\}$ the space of N-times continuously differentiable functions from some set A to some set B is $\mathcal{C}^N(A \to B)$, the N-th derivative of $f \in \mathcal{C}^N(A \to B)$ is denoted by $f^{(N)}$ or f', f'' for $N = 1, 2$. For some sufficiently smooth function f the Taylor polynomial of degree N is denoted by $\mathcal{T}_N\{f\}$ (see Definition 4 for details). A function $f : A \to B$ is called surjective, if it is "onto", i.e., $f(A) = B$. For two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $n \in \mathbb{N}$, the standard Euclidian inner product is denoted by $\mathbf{x} \cdot \mathbf{y}$, the maximum norm of $\mathbf{x} \in \mathbb{R}^n$ is denoted by $\|\mathbf{x}\| := \max\big\{|x_1|, \ldots, |x_n|\big\}$. For some real number x the value $\lfloor x \rfloor$ is the largest integer not bigger than x, the value $\lceil x \rceil$ is the smallest integer not smaller than x.

II. The multilayer perceptron (MLP)

The multilayer perceptron (MLP) is a very simple model of biological neural networks and is based on the principle of a feed-forward-flow of information, i.e. the network is structured in a hierarchical way. The MLP consists of different layers where the information flows only from one layer to the next layer. Layers between the input and output layer are called hidden layers. From a theoretical point of view, it is not necessary to consider more than one output unit because two or more output units could be realized by considering two or more MLPs in parallel. However, if the outputs are correlated it may be possible to achieve the same approximation results with fewer hidden units. Nevertheless, a correlation analysis of different outputs and its implications to the necessary number of hidden units is beyond the scope of this work.

The input units play no active role in processing the information flow, because they just distribute the signals to the units of the first hidden layer. All hidden units work in an identical way and the output unit is a simpler version of a hidden unit. In an MLP, each hidden unit transforms the signals from the former layer to one output signal, which is distributed to the next layer. Each hidden unit has an, in general nonlinear, activation function. The activation function is modulo a translation via an individual bias the same for all hidden units. The output of a hidden unit is determined by the weighted sum of the signals from the former layer, which is then transformed by the activation function. In the output unit the activation function is the identity function.

The following two definitions give a formal definition of an MLP and the corresponding MLP function. Most of the formalism is not needed in the rest of the paper, but it is necessary to give a precise definition of an MLP, on which the results of the paper are based on.

Definition 1 (Multilayer Perceptron - MLP): A multilayer perceptron (MLP) is a quadtuple

$$(h, \mathbf{n}, \sigma, \mathbf{P}),$$

where $h \in \mathbb{N}$ is the number of hidden layers, $\mathbf{n} = (n_0, n_1, \ldots, n_h) \in \mathbb{N}^{h+1}$ is the number of units per hidden layer (the hidden layer zero is the input layer), $\sigma : \mathbb{R} \to \mathbb{R}$ is the activation function and

$$\mathbf{P} = (\mathbf{W}^1, \ldots, \mathbf{W}^h, \mathbf{w}^y),$$

where, for $i = 1, \ldots, h$, $\mathbf{W}^i = (\mathbf{w}^{i,1}, \ldots, \mathbf{w}^{i,n_i}) \in (\mathbb{R}^{n_{i-1}+1})^{n_i}$ are the parameters (weights and biases) between the $(i-1)$-th and i-th hidden layer and $\mathbf{w}^y \in \mathbb{R}^{n_h}$ is the vector of the parameters between the last hidden layer and the output unit.

Definition 2 (MLP function): For an MLP $(h, \mathbf{n}, \sigma, \mathbf{P})$ as in Definition 1, the MLP-function

$$f_{\text{MLP}} : \mathbb{R}^{n_0} \to \mathbb{R}, \quad \mathbf{x} = (x_1, x_2, \ldots, x_{n_0}) \mapsto y$$

is recursively defined by

$$\begin{aligned}
&y = \mathbf{w}^y \cdot \mathbf{z}^h, \text{ where} \\
&\quad \mathbf{z}^h = \big(\ \sigma(\mathbf{w}^{h,1} \cdot \mathbf{z}^{h-1})\ ,\ \ldots\ ,\ \sigma(\mathbf{w}^{h,n_h} \cdot \mathbf{z}^{h-1})\ \big), \\
&\quad \mathbf{z}^i = \big(\ \sigma(\mathbf{w}^{i,1} \cdot \mathbf{z}^{i-1})\ ,\ \ldots\ ,\ \sigma(\mathbf{w}^{i,n_i} \cdot \mathbf{z}^{i-1})\ ,\ 1\ \big) \\
&\qquad \text{for } i = h-1, \ldots, 1, \\
&\quad \mathbf{z}^0 = (x_1, x_2, \ldots, x_{n_0}, 1).
\end{aligned}$$

Note that for standard MLPs the activation function is given by $\sigma(t) = 1/(1+e^{-t})$. For the results in this paper the specific form of the activation function does not play any role, it is only assumed, that the activation function is smooth (otherwise the concept of approximation order does not make sense). Indeed, it turns out that the above activation function does not fulfill the assumptions of Proposition 7, where conditions are given for which a higher approximation order implies a better overall approximation accuracy; nevertheless the main results of this paper still hold for this activation function.

For practical applications it is not necessary to consider the MLP function as a function on the whole space $\mathbb{R}^{n_0}$, because the input is restricted by physical or other bounds. It is therefore no restriction to assume that the input $\mathbf{x} = (x_1, x_2, \ldots, x_{n_0}) \in \mathbb{R}^{n_0}$ is scaled such that $x_1, x_2, \ldots, x_{n_0} \in [-1, 1]$. Hence, in the rest of the paper the input space is $K = [-1, 1]^{n_0}$.

III. Approximation order

MLPs are used to approximate some function. It is necessary to precisely define what "approximation" should mean, in particular, when one approximation is better then another. One possible measure for approximation accuracy might be the largest error between the function and its approximator. It is well known that MLPs can approximate any continuous function with an arbitrary high approximation accuracy in the above sense (see e.g. [8]), but there are doubts that this result can be practically achieved if the structure of the MLP is fixed, [12]. Often the overall accuracy is less important than a good local approximation; this viewpoint yields the concept of "approximation order".

Definition 3 (Approximation order): A function $f \in \mathcal{C}^N(K \to \mathbb{R})$ approximates $g \in \mathcal{C}^N(K \to \mathbb{R})$ with order $N \in \mathbb{N}$ if, and only if, $f(0) = g(0)$, $f'(0) = g'(0)$, $f''(0) = g''(0)$, $\ldots$, $f^{(N)}(0) = g^{(N)}(0)$.

The concept of approximation order is strongly connected with Taylor polynomials:

Definition 4 (Taylor polynomial): For $N \in \mathbb{N}$ and $f \in \mathcal{C}^N(K \to \mathbb{R})$,

$$\mathcal{T}_N\{f\} : K \to \mathbb{R},\ \mathbf{x} \mapsto \sum_{i=0}^{N} \frac{1}{i!} f^{(i)}(0)\mathbf{x}^i$$

is the Taylor polynomial of degree N. For $N \to \infty$, $\mathcal{T}_\infty\{f\}$ is called the Taylor series, if it exists.

Note that the derivatives of f are viewed as multilinear mappings and $f^{(i)}(0)\mathbf{x}^i$ stands short for $f^{(i)}(0)(\mathbf{x}, \mathbf{x}, \ldots, \mathbf{x})$ (see VII.5 in [13]). The evaluation of this multilinear mappings yields multivariable polynomials, e.g.,

$$\begin{aligned}
\mathcal{T}_2\{f\}(x_1, x_2) = {}& \overbrace{a_{01}}^{f(0)} + \overbrace{a_{11}x_1 + a_{12}x_2}^{f'(0)(x_1,x_2)} \\
& + \underbrace{a_{21}{x_1}^2 + a_{22}x_1x_2 + a_{23}{x_2}^2}_{\frac{1}{2}f''(0)(x_1,x_2)^2},
\end{aligned}$$

where $a_{01}, \ldots, a_{23} \in \mathbb{R}$ are the coefficients of the Taylor polynomial $\mathcal{T}_2\{f\}$ which are determined by f.

Clearly, some sufficiently smooth function f approximates another function g with order N if, and only if,

$$\mathcal{T}_N\{f\} = \mathcal{T}_N\{g\}.$$

It is a classical result, that for sufficiently smooth functions f the Taylor polynomial $\mathcal{T}_N\{f\}$ is a local approximation of f, [13, Thm. VII.5.11]. But even if the Taylor series converges it does not necessarily coincide with the function or is a good global approximation. The following proposition gives an answer to the question, when a local approximation is also a good global approximation. For the formulation of the proposition the following definition is necessary first.

Definition 5 (Analyticity): A function $f \in \mathcal{C}^\infty\big(K \to \mathbb{R}\big)$ is called analytical (in zero) if, and only if, there exist $\delta > 0$ such that

$$\mathcal{T}_\infty\{f\}(\mathbf{x}) = f(\mathbf{x}) \quad \forall\, \mathbf{x} \in K \text{ with } \|\mathbf{x}\| < \delta.$$

A function $f \in \mathcal{C}^\infty\big(K \to \mathbb{R}\big)$ is called *nicely analytical* if, and only if,

$$\mathcal{T}_\infty\{f\}(\mathbf{x}) = f(\mathbf{x}) \quad \forall\, \mathbf{x} \in K \tag{1}$$

and

$$\limsup_{k \to \infty} \sqrt[k]{\frac{\|f^{(k)}(0)\|}{k!}} < 1, \tag{2}$$

where $\|f^{(k)}(0)\|$ is the operator norm of the multilinear operator $f^k(0)$ (see e.g. [13, Thm. VII.4.2]).

Proposition 6 (Local vs. global approximation): If f and g are nicely analytical functions then there exists for every $\varepsilon > 0$ an $N_\varepsilon \in \mathbb{N}$ such that the following implication holds

$$\mathcal{T}_{N_\varepsilon}\{f\} = \mathcal{T}_{N_\varepsilon}\{g\} \ \Rightarrow\ \sup_{\mathbf{x} \in K} \big|f(\mathbf{x}) - g(\mathbf{x})\big| < \varepsilon,$$

i.e. for nicely analytical functions better local approximation implies better global approximation.

It is no restriction to consider only nicely analytical functions as target functions, because all polynomials are nicely analytical and the space of polynomials is dense in the space of continuous functions, see, e.g., [14, Kor. V.4.8]. The condition that the MLP function is nicely analytical might restrict the choice of the activation functions. The following proposition give a condition for the activation function which ensures that the corresponding MLP function is nicely analytical.

Proposition 7 (Nicely analytical MLP function): Consider an MLP with an analytical activation function $\sigma : \mathbb{R} \to \mathbb{R}$ for which $\mathcal{T}_\infty\{\sigma\}(t) = \sigma(t)$ for all $t \in \mathbb{R}$, i.e., the Taylor series coincide globally with σ. Then the MLP function $f_{\text{MLP}} : K \to \mathbb{R}$ is nicely analytical.

The two former propositions show that for appropriate activation functions of the MLP an arbitrarily good global approximation can be achieved by increasing the approximation order with which the MLP function approximates the desired nicely analytical function. However, it will not be possible to directly calculate the necessary size of an MLP to achieve a given (global) approximation accuracy, because the necessary approximation order will in general depend on the target function. As already mentioned above, the standard activation function given by $\sigma(t) = 1/(1+e^{-t})$ does not fulfill the condition of Proposition 7. An activation function which fulfills the assumption of Proposition 7 is, for example, the sine function. It should be noted at this point, that although polynomials fulfill the condition of Proposition 7 they can for large N_ε not fulfill the left side of the implication in Proposition 6. In particular, Proposition 6, only makes sense if σ is not a polynomial, compare [8, Thm. 3.1].

IV. Number of necessary hidden units

The main idea for the calculation of the necessary number of hidden units in an MLP to achieve a given approximation order is to ask on the one hand how many independent values *must be* adjusted to achieve a given approximation order for an arbitrarily sufficient smooth function and on the other hand how many independent values *can be* adjusted by varying the network parameters. From an abstract point of view the latter is equivalent to the question, whether some function with n variables can have as function values all values of an m-dimensional space. A necessary condition is given in the next lemma.

Lemma 8 (Surjectivity of differentiable functions): Let $n, m \in \mathbb{N}$, $U \subseteq \mathbb{R}^n$ open and $g \in \mathcal{C}^1(U \to \mathbb{R}^m)$. If $n < m$ then g is not surjective.

Although this result seems intuitively clear its proof is not trivial. One should note, in particular, that there exist *continuous* functions $g \in C(\mathbb{R}^n \to \mathbb{R}^m)$ with $n < m$ which are surjective. For $n = 1$ and $m = 2$ these functions are called space filling curves or Peano curves [15].

Each MLP with fixed network parameters $\mathbf{P}$ induces an MLP function $f_{\text{MLP}} : K \to \mathbb{R}$. The MLP approximates a function $f \in \mathcal{C}^N(K \to \mathbb{R})$ with order $N \in \mathbb{N}$ if, and only if, the Taylor polynomials of degree N of f_{MLP} and f are equal, i.e. $\mathcal{T}_N\{f_{\text{MLP}}\} = \mathcal{T}_N\{f\}$. The latter is equivalent to the condition that all corresponding coefficients of the two Taylor polynomials are equal. Clearly, every parameter set $\mathbf{P}$ induces different MLP functions and in particular different coefficients of the Taylor polynomial $\mathcal{T}_N\{f_{\text{MLP}}\}$. Since the coefficients of the Taylor polynomial $\mathcal{T}_N\{f\}$ can be arbitrary it is for the considered approximation task necessary that the function which maps the network parameters $\mathbf{P}$ to the coefficients of the Taylor polynomial $\mathcal{T}_N\{f_{\text{MLP}}\}$ is surjective. Therefore Lemma 8 yields the next corollary.

Corollary 9 (Necessary condition for MLPs): For an MLP with an activation function $\sigma \in \mathcal{C}^\infty(\mathbb{R} \to \mathbb{R})$ which can achieve an approximation order $N \in \mathbb{N}$ for any target function $f \in \mathcal{C}^N(K \to \mathbb{R})$ the number of network parameters can not be smaller than the maximum number of independent coefficients of a Taylor polynomial of degree N.

It remains now to find formulas for the number of network parameters (as function of the number of hidden units) and the number of independent coefficients of a Taylor polynomial. For the latter there is a simple formula:

Lemma 10 (Number of coefficients in polynomials): A multivariable polynomial of degree N with n_0 variables has at most

$$\binom{N+n_0}{n_0} = \frac{(N+n_0)!}{N!\,n_0!}$$

independent coefficients.

The calculation of the number of parameters in an MLP is not so straight forward, because for a given number of hidden units the number of network parameters is not unique. The reason is that the hidden units can be distributed in different hidden layers in many different ways. Since the aim is to find the necessary number of hidden units one has to search for the maximum number of parameters when the number of hidden units is given. The result is given in the next proposition.

Proposition 11 (Maximum number of network parameters): Consider an MLP with n hidden units and n_0 inputs. Let $m \in \{0, 1\}$ be such that $m \equiv n + n_0 \mod 2$. The maximum number of network parameters is then given by

$$(n_0 + 2)n$$

if $n \leq n_0 + 1 + \sqrt{4n_0 + 1 - m}$ and

$$\frac{(n + n_0 + 3)^2 + m - 1}{4} - 2(n_0 + 1)$$

otherwise. In the first case the maximum number is achieved for a single hidden layer MLP, in the second case two hidden layers are necessary to achieve the maximum number, where

$$n_1 = \frac{n + n_0 - m}{2}$$

hidden units are in the first hidden layer and

$$n_2 = \frac{n - n_0 + m}{2}$$

hidden units are in the second hidden layer.

It should be noted that Proposition 11 states, in particular, that more than two hidden layers are not necessary if one wants to maximize the number of parameters for a given number of hidden units. Combining Corollary 9 with the results from Lemma 10 and Proposition 11 it is possible to formulate the two main results of this paper. The first result considers the case where the number of hidden layers in the MLP is restricted to one (this might be relevant in technical applications).

Theorem 12 (Main result for single hidden layer MLPs): An MLP with one hidden layer, $n_0 \in \mathbb{N}$ inputs and smooth activation function can only achieve approximation order $N \in \mathbb{N}$ for all functions $f \in \mathcal{C}^N(K \to \mathbb{R})$, if at least

$$\frac{\binom{N+n_0}{n_0}}{n_0 + 2}$$

hidden units are used.

For the following main result no restriction on the number of hidden layers is assumed. It turns out that more than two layers are not necessary. In some cases one layer suffices but in many cases the necessary hidden units must be distributed to two hidden layers to achieve the necessary number of network parameters.

Theorem 13 (Main result): Consider an MLP with $n_0 \in \mathbb{N}$ input units and smooth activation function. Let $N \in \mathbb{N}$ be the desired approximation order. If

$$\binom{N+n_0}{n_0} \leq (n_0 + 2)(n_0 + 1 + 2\sqrt{n_0})$$

then at least

$$\boxed{n \geq \frac{\binom{N+n_0}{n_0}}{n_0 + 2}}$$

hidden units are necessary to achieve approximation order $N \in \mathbb{N}$ for all functions $f \in \mathcal{C}^N(K \to \mathbb{R})$, otherwise

$$\boxed{n \geq 2\sqrt{\binom{N+n_0}{n_0} + 2(n_0+1)} - n_0 - 3}$$

hidden units are necessary.

In the first case an MLP with one hidden layer achieves the necessary number of parameters. For the second case the necessary number of parameters are obtained for an MLP with two hidden layers with

$$n_1 = \left\lceil \frac{n + n_0 - 1}{2} \right\rceil,$$

units in the first hidden layer and

$$n_2 = n - n_1 = \left\lfloor \frac{n - n_0 + 1}{2} \right\rfloor$$

hidden units in the second hidden layer.

The explicit formulas from Theorem 12 and 13 can be used to calculate the number of necessary hidden units and its distribution to one or two hidden layers if the number of inputs and the desired approximation oder are given. For a range of number of input signals and approximation order these calculated values are displayed in Table I.

Remark 14 (Number of hidden layers):

1) It is never necessary to use more than one hidden layer, as can be seen from Theorem 12, but if one uses only the minimal number of hidden units from the second case of Theorem 13 then one has to use two hidden layers to obtain the necessary number of parameters. The same stays true, if more than the minimal number of hidden units are used; but if the number of hidden units is large enough, then two hidden layers are not necessary any more (although two hidden layers would still be advantageous, because with the same number of hidden units more parameters are available, which in general will lead to better approximation results).
2) From the condition in Theorem 13 it follows that if only linear or quadratic approximation should be achieved, i.e. $N \leq 2$, then only one hidden layer is needed. On the other hand, if the desired approximation order is at least twelve, then two hidden layers are needed (in the sense of 1).

Remark 15 (Growth of the number of hidden units): If n_0 is fixed then the necessary number of hidden units grows polynomially in the approximation order N. Asymptotically (big O notation), it is for the single hidden hidden layer case $n = O(N^{n_0})$ and for the two hidden layer case $n = O(N^{n_0/2})$. Analogously, if the approximation order N is fixed then $n = O(n_0^{N-1})$ for the single hidden layer case and $n = O(n_0^{N/2})$ for the two hidden layers case.

V. Conclusions

The main contribution of this paper is the explicit formula for the necessary number of hidden units in a multilayer perceptron to achieve a given approximation order. It was also possible to decide how many hidden layers should be used. It turns out that more than two hidden layers are not needed, if one aims to minimize the number of necessary hidden units. Depending on the number of inputs and the desired approximation order one or two hidden layers should be used. For high approximation orders (≥ 12) two hidden layers should be used instead of one hidden layer, the same

order	number of inputs								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1	1 1	1 1	1 1	1 1	1 1	1 1	1 1	1 1	1 1
2	1 1	2 2	2 2	3 3	3 3	4 4	4 4	5 5	5 5
3	2 2	3 3	4 4	6 6	8 8	11 11	14 (10,4)	17 (12,5)	20 (14,5)
4	2 2	4 4	7 7	12 (7,4)	18 (10,6)	27 (13,8)	37 (17,11)	50 (21,14)	65 (26,17)
5	2 2	6 6	12 (6,4)	21 (10,7)	36 (15,10)	58 (20,15)	88 (27,20)	129 (35,27)	182 (43,35)
6	3 3	7 (4,3)	17 (8,6)	35 (13,10)	66 (20,16)	116 (29,24)	191 (40,34)	301 (53,46)	455 (69,61)
7	3 3	9 (5,3)	24 (10,7)	55 (17,13)	114 (27,22)	215 (40,35)	382 (57,51)	644 (79,71)	1040 (106,97)
8	3 3	12 (6,4)	33 (12,9)	83 (21,17)	184 (35,30)	376 (53,48)	715 (79,72)	1287 (112,105)	2210 (154,146)
9	4 4	14 (6,5)	44 (14,11)	120 (25,22)	286 (43,39)	626 (69,64)	1272 (106,99)	2431 (154,147)	4420 (219,211)
10	4 4	17 (7,5)	58 (16,13)	167 (30,27)	429 (53,49)	1001 (88,83)	2161 (138,132)	4376 (208,200)	8398 (302,294)
11	4 4	20 (8,6)	73 (18,15)	228 (36,32)	624 (65,60)	1547 (110,104)	3536 (177,170)	7559 (273,266)	15270 (408,400)
12	5 (3,2)	23 (8,7)	91 (20,18)	304 (41,38)	884 (77,73)	2321 (135,129)	5599 (223,217)	12597 (353,346)	26721 (541,532)
13	5 (3,2)	27 (9,8)	112 (22,20)	397 (47,44)	1224 (91,87)	3392 (163,158)	8614 (277,270)	20349 (450,442)	45220 (704,695)
14	5 (3,2)	30 (10,8)	136 (25,22)	510 (54,50)	1662 (106,102)	4845 (195,190)	12920 (340,333)	31977 (564,556)	74290 (902,894)
15	6 (3,2)	34 (10,9)	164 (27,25)	646 (61,57)	2215 (123,119)	6783 (231,226)	18950 (411,405)	49032 (699,691)	118864 (1142,1133)

TABLE I

Necessary number of hidden units for an MLP with 1 to 9 inputs and desired approximation order 1 to 15. The first entry is the necessary number of hidden units for an MLP with one hidden layer, the second entry is the necessary number of hidden units for a two hidden layers MLP, if the second entry consists only of one value, then a second layer is not needed.

is true for smaller approximation order and a sufficiently high number of inputs, as long as the approximation order is at least three. Interestingly, for linear and quadratic approximation only one hidden layer is needed.

The correlation between approximation order and approximation accuracy was studied. A sufficient condition was given for the activation function for which a high approximation order implies a high approximation accuracy, or in other words when a good local approximations also yields a good global approximation.

Although the important question "How many hidden units are necessary?" was answered in a satisfying manner, there are other important questions which remain open. The next obvious question considers the sufficient number of hidden units and under which conditions the number of necessary hidden units, calculated in this paper, is also sufficient. Another important question is how an MLP must be trained to achieve a good approximation order.

Acknowledgment

This paper has its origin in the diploma thesis [16] written by the author and supervised by PD Dr. P. Otto. The author would like to thank PD Dr. P. Otto for giving him this fruitful topic to work on and for the valuable comments to a first version of this paper. Thanks goes also to the author's PhD supervisor Prof. A. Ilchmann for leaving the freedom to work on a completely different topic.

Appendix

Proof of Proposition 6

From the definition of $f^{(n)}(0)$ (see, e.g., [13]) it follows that

$$\left|f^{(n)}(0)\mathbf{x}^n\right| \leq \left\|f^{(n)}(0)\right\|\|\mathbf{x}\|^n \leq \left\|f^{(n)}(0)\right\|$$

for all $n \in \mathbb{N}$, $\mathbf{x} \in K = [-1,1]^{n_0}$ and $\|\mathbf{x}\| = \max\left\{|x_1|, |x_2|, \dots |x_{n_0}|\right\}$. Let $\varepsilon > 0$, then there exists, because of (2), an $N_\varepsilon \in \mathbb{N}$ such that

$$\sum_{k=N_\varepsilon+1}^{\infty} \frac{1}{k!}\|f^{(k)}(0)\| < \varepsilon.$$

From (1) it follows that for all $\mathbf{x} \in K$ and all $N \in \mathbb{N}$

$$\left|\mathcal{T}_N\{f\}(\mathbf{x}) - f(\mathbf{x})\right| = \left|\sum_{k=N+1}^{\infty} \frac{f^{(k)}(0)}{k!}\mathbf{x}^k\right|,$$

hence

$$\left|\mathcal{T}_N\{f\}(\mathbf{x}) - f(\mathbf{x})\right| \leq \sum_{k=N+1}^{\infty} \frac{1}{k!}\|f^{(k)}(0)\| < \varepsilon$$

for all $N \geq N_\varepsilon$. $\boxed{\text{qed}}$

Proof of Proposition 7

For the proof of this proposition a lemma is needed:

Lemma 16: Let $\sigma : \mathbb{R} \to \mathbb{R}$ be a activation function which fulfills the assumption from Proposition 7 and let $f_1, f_2, \ldots, f_m : K \to \mathbb{R}$ be nicely analytical functions for some $m \in \mathbb{N}$ and $K = [-1, 1]^{n_0}$. The function

$$g : K \to \mathbb{R},\ \mathbf{x} \mapsto \sigma\big(w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \ldots + w_m f_m(\mathbf{x}) + \theta\big),$$

where $w_1, w_2, \ldots, w_m, \theta \in \mathbb{R}$, is then also nicely analytical.

The proof of Lemma 16 is technically and is carried out in the appendix of [16, Lem. 4.5.3].

For the proof of Proposition 7 consider Definition 2 for the MLP function, then it follows inductively by Lemma 16 that the mappings $\mathbf{x} \mapsto z^{i,j}$ for each $1 \leq i \leq h$ and $1 \leq j \leq n_i$ are nicely analytical and hence the MLP function $\mathbf{x} \mapsto y$ is also nicely analytical, because the output activation function is the identity function and fulfills therefore the assumptions of Lemma 16. $\boxed{\text{qed}}$

Proof of Lemma 8

The main ideas of the proof are based on [17].
The space $\mathbb{R}^m$ is Lindelöf (i.e. every open covering has a countable subcovering, see, e.g., [18]) and hence $U \subseteq \mathbb{R}^m$ is Lindelöf, too. The set

$$\mathcal{R}_g := \left\{ y \in \mathbb{R}^n \;\middle|\; \begin{array}{l} \forall x \in g^{-1}(y) : \\ g'(x) : \mathbb{R}^m \to \mathbb{R}^n \text{ is surjective} \end{array} \right\}$$

is by Sard's Theorem for Manifolds, [18, Thm. 3.6.8], a countable intersection of open dense sets. Note that for $y \notin g(U)$ trivially $y \in \mathcal{R}_g$. On the other hand, for every $y \in g(U)$ and $x \in g^{-1}(y)$ the linear mapping $g'(x)$ is not surjektive, because $m < n$. Hence $g(U) \cap \mathcal{R}_g = \emptyset$, i.e. $g(U) \subseteq \mathbb{R}^n \backslash \mathcal{R}_g$. The complement of a countable intersection of open dense sets is a countable union of closed sets with empty interior, hence $g(U) \subseteq M := \bigcup_{i \in \mathbb{N}} C_i$, where $C_i \subseteq \mathbb{R}^n$ are closed sets with empty interior.
Seeking a contradiction assume $g(U) = \mathbb{R}^n$, which implies that $M = \mathbb{R}^n$.
The space $\mathbb{R}^n$ is locally compact and the Baire Category Theorem, [18, Thm. 1.7.3], yields that $\mathbb{R}^n$ is a Baire space, i.e. every countable intersection of open and dense subsets is dense. For $i \in \mathbb{N}$ the subsets $O_i := \mathbb{R}^n \backslash C_i$ are open and dense and hence $\bigcap_{i \in \mathbb{N}} O_i$ is dense in $\mathbb{R}^n$. This yields the contradiction

$$\emptyset = \mathbb{R}^n \backslash M = \mathbb{R}^n \backslash \bigcup_{i \in \mathbb{N}} C_i = \bigcap_{i \in \mathbb{N}} O_i.$$

$\boxed{\text{qed}}$

Proof of Corollary 9

Consider an MLP with parameters $\mathbf{P}$ (weights and biases), then the MLP function $f_{\text{MLP}} : K \to \mathbb{R}$ depends on the specific chosen parameters $\mathbf{P}$, in particular the Taylor polynomial $\mathcal{T}_N\{f_{\text{MLP}}\}$ for some $N \in \mathbb{N}$ depends on $\mathbf{P}$. Denote with $N_{\mathbf{P}}$ the dimension of $\mathbf{P}$, i.e., the number of parameters in the considered MLP. Denote furthermore the maximum number of coefficients of the Taylor polynomial $\mathcal{T}_N\{f_{\text{MLP}}\}$ with $N_{\mathcal{T}}$. Let $\mathcal{C}_{\mathcal{T}} : \mathbb{R}^{N_{\mathbf{P}}} \to \mathbb{R}^{N_{\mathcal{T}}}$ be the function which maps the parameters $\mathbf{P}$ to the coefficient vector of the Taylor polynomial $\mathcal{T}_N\{f_{\text{MLP}}\}$, i.e. $\mathcal{C}_{\mathcal{T}}(\mathbf{P}) = (b_1, b_2, \ldots, b_{N_{\mathcal{T}}})$, where $b_1, \ldots, b_{N_{\mathcal{T}}}$ are the $N_{\mathcal{T}}$ coefficients of $\mathcal{T}_N\{f_{\text{MLP}}\}$. The condition that the considered MLP can approximate any sufficiently smooth function with order N is then equivalent to the condition that the function $\mathcal{C}_{\mathcal{T}}$ is surjective.

Therefore, the assertion of Corollary 9 is $N_{\mathbf{P}} \geq N_{\mathcal{T}}$. In view of Lemma 8, it only remains to show that the function $\mathbb{R}^{N_{\mathbf{P}}} \to \mathbb{R}^{N_{\mathcal{T}}}$, $\mathbf{P} \mapsto \mathcal{C}_{\mathcal{T}}(\mathbf{P})$ is differentiable, because then surjectivity of $\mathbf{P} \mapsto \mathcal{C}_{\mathcal{T}}(\mathbf{P})$ implies $N_{\mathbf{P}} \geq N_{\mathcal{T}}$.

Write $\mathcal{T}_N\{f_{\text{MLP}}\}(\mathbf{x}) = \sum_{|I| \leq N} b_I(\mathbf{P}) \mathbf{x}^I$, where $I = (i_1, i_2, \ldots, i_{n_0}) \in \mathbb{N}^{n_0}$, $|I| = i_1 + i_2 + \ldots + i_{n_0}$ and $\mathbf{x}^I = {x_1}^{i_1} {x_2}^{i_2} \cdots {x_{n_0}}^{i_{n_0}}$, then the Taylor coefficient function $\mathcal{C}_{\mathcal{T}}$ consists of the scalar valued functions $\mathbf{P} \mapsto b_I(\mathbf{P}) \in \mathbb{R}$ for indexes $I \in \mathbb{N}^{n_0}$ with $|I| \leq N$. It suffices now to show that for each $I \in \mathbb{N}^{n_0}$ with $|I| \leq N$ the function $\mathbf{P} \mapsto b_I(\mathbf{P})$ is differentiable.

To highlight that f_{MLP} depends on $\mathbf{P}$ write $f_{\text{MLP}}(\mathbf{P}, \mathbf{x})$ instead of $f_{\text{MLP}}(\mathbf{x})$. It is

$$b_I(\mathbf{P}) = c_I \frac{\partial^{|I|} f_{\text{MLP}}(\mathbf{P}, \cdot\,)}{(\partial \mathbf{x})^I}(0),$$

where c_I is some multiple which results from the symmetries of the partial derivatives and

$$(\partial \mathbf{x})^I = \partial^{i_1} x_1 \partial^{i_2} x_2 \cdots \partial^{i_{n_0}} x_{n_0},$$

if $I = (i_1, i_2, \ldots, i_{n_0})$. From the definition of the MLP function (see Definition 2) and the assumption that $\sigma \in \mathcal{C}^\infty(\mathbb{R} \to \mathbb{R})$ it follows that the MLP function $(\mathbf{P}, \mathbf{x}) \mapsto f_{\text{MLP}}(\mathbf{P}, \mathbf{x})$ is not only arbitrarily often continuously differentiable with respect to $\mathbf{x}$, but also with respect to $\mathbf{P}$. This implies that the function $(\mathbf{P}, \mathbf{x}) \mapsto f_{\text{MLP}}(\mathbf{P}, \mathbf{x})$ is arbitrarily often differentiable. In particular, there exists for every partial derivative a further partial derivative. The derivative of $\mathbf{P} \mapsto b_I(\mathbf{P})$ is simply the partial derivative (with respect to $\mathbf{P}$) of $c_I \frac{\partial^{|I|} f_{\text{MLP}}(\mathbf{P}, \cdot\,)}{(\partial \mathbf{x})^I}(0)$, which itself is a partial derivative of the MLP function $(\mathbf{P}, \mathbf{x}) \mapsto f_{\text{MLP}}(\mathbf{P}, \mathbf{x})$ with respect to $\mathbf{x}$. Hence $\mathbf{P} \mapsto b_I(\mathbf{P})$ is differentiable, which implies differentiability of $\mathcal{C}_{\mathcal{T}}$. $\boxed{\text{qed}}$

Proof of Lemma 10

Consider some multivariable polynomial $(x_1, x_2, \ldots, x_n) \mapsto p(x_1, x_2, \ldots, x_n)$ with $n \in \mathbb{N}$ variables and degree $N \in \mathbb{N}$ and write

$$\begin{aligned} &p(x_1, x_2, \ldots, x_n) \\ &\qquad = p_1(x_1, x_2, \ldots, x_{n-1}) + x_n p_2(x_1, x_2, \ldots, x_n), \end{aligned}$$

where p_1 consists of all monomials of p without the variable x_n and p_2 consists of all remaining monomials divided by x_n. Let $\mathcal{C}(n, N)$ be the number of coefficients of a polynomial with n variables and degree N then, by the above splitting, the following recursive formula holds

$$\mathcal{C}(n, N) = \mathcal{C}(n-1, N) + \mathcal{C}(n, N-1)$$

with initial conditions $\mathcal{C}(n, 0) = 1$ and $\mathcal{C}(0, N) = 1$. The term $\begin{pmatrix} N+n \\ n \end{pmatrix}$ fulfills the recursive formula and its initial conditions and hence the lemma is shown. $\boxed{\text{qed}}$

Proof of Proposition 11

It follows from Definition 1 that the number of parameters $N_{\mathbf{P}}(h,\mathbf{n})$ of an MLP with h hidden layers and $\mathbf{n} = (n_0, n_1, \dots, n_h)$ units is

$$N_{\mathbf{P}}(h,\mathbf{n}) = \sum_{i=1}^{h}(n_{i-1}+1)n_i + n_h. \tag{3}$$

Let

$$N^*_{\mathbf{P}}(n_0,n) := \max_{h\in\mathbb{N}, |\mathbf{n}|=n} N_{\mathbf{P}}(h,\mathbf{n}),$$

where $|\mathbf{n}| = n_1 + n_2 + \ldots + n_h$ and the maximum is only taken over $\mathbf{n} = (n_0, n_1, \dots, n_h)$, where $n_1, ..., n_h > 0$ and $n_0 > 0$ is fixed. The number $N^*_{\mathbf{P}}(n_0,n)$ is the maximum number of parameters which an MLP with n hidden units can have, regardless how the hidden units are distributed in the different hidden layers. For the calculation of $N^*_{\mathbf{P}}(n_0,n)$ consider first the case that the number of hidden layers is fixed:

$$N^h_{\mathbf{P}}(n_0,n) := \max_{|\mathbf{n}|=n} N_{\mathbf{P}}(h,\mathbf{n}).$$

Clearly, by (3),

$$N^1_{\mathbf{P}}(n_0,n) = (n_0+1)n + n.$$

For $h = 2$ the n hidden units can be distributed to the two hidden layers such that n_2 units are in the second and $n_1 = n - n_2$ units are in the first hidden layer. Therefore, by (3),

$$N^2_{\mathbf{P}}(n_0,n) = \max_{1\le n_2\le n-1}\big((n_0+1)(n-n_2)+(n-n_2+1)n_2+n_2\big).$$

To calculate $N^2_{\mathbf{P}}(n_0,n)$ consider for fixed $n, n_0 \in \mathbb{N}$ the function

$$m : \mathbb{R} \to \mathbb{R}, \quad x \mapsto -x^2 + x(n-n_0+1) + (n_0+1)n,$$

then $N^2_{\mathbf{P}}(n_0,n) = \max_{1\le n_2\le n-1} m(n_2)$. The real valued function m has a unique maximum at

$$x_{\max} = \frac{n-n_0+1}{2}.$$

Since m is a parabola, $N^2_{\mathbf{P}}(n_0,n)$ is maximal for $n_2 = \left\lfloor \frac{n-n_0+1}{2} \right\rfloor$ and $n_2 = \left\lceil \frac{n-n_0+1}{2} \right\rceil$. Note that the function value is the same for both points. The optimal value n_2 is only valid, if $1 \le n_2 \le n-1$ otherwise one of the hidden layers would be empty. Since $n \ge 1$ and $n_0 \ge 0$ this yields for the optimal number n_2^* of hidden units in the second layer:

$$n_2^* = \max\left\{ \left\lfloor \frac{n-n_0+1}{2} \right\rfloor \ , \ 1 \right\}.$$

Note that with $m \equiv n + n_0 \mod 2$ it is $\left\lfloor \frac{n-n_0+1}{2} \right\rfloor = \frac{n-n_0+m}{2}$. Hence

$$\begin{aligned} &N^2_{\mathbf{P}}(n_0,n) \\ &= \begin{cases} \frac{(n+n_0+3)^2+m-1}{4} - 2(n_0+1) & \text{if } n \ge n_0+2-m, \\ (n_0+1)(n-1)+n-1 & \text{otherwise.} \end{cases} \end{aligned}$$

For the first case the maximum is obtained for $n_2 = \left\lfloor \frac{n-n_0+1}{2} \right\rfloor$ and in the second case for $n_2 = 1$. For the latter case it is clearly better to take only one hidden layer, because with one hidden layer more parameters can be obtained. Evaluating the inequality $N^2_{\mathbf{P}}(n_0,n) > N^1_{\mathbf{P}}(n_0,n)$ yields

$$N^2_{\mathbf{P}}(n_0,n) > N^1_{\mathbf{P}}(n_0,n) \ \Leftrightarrow \ n > n_0 + 1 + \sqrt{4n_0+1-m},$$

which is the statement of the proposition (note that $n > n_0 + 1 + \sqrt{4n_0+1-m}$ implies $n \ge n_0 + 2 - m$), if also it is shown that more than two hidden layers are not needed.

It is

$$N^3_{\mathbf{P}}(n_0,n) = \max_{n_2,n_3} N_{\mathbf{P}}\big(3, (n_0, n-n_2-n_3, n_2, n_3)\big)$$

and by (3)

$$\begin{aligned} &N_{\mathbf{P}}\big(3, (n_0, n-n_2-n_3, n_2, n_3)\big) \\ &\quad = (n_0+1)n + n_2(n-n_0-n_2-1) - n_3(n_0-1) \end{aligned}$$

Clearly, the value of n_3 must be chosen minimal to maximise $N_{\mathbf{P}}\big(3, (n_0, n-n_2-n_3, n_2, n_3)\big)$, because $n_0 \ge 1$ (if $n_0 = 1$ then the value of the maximum does not depend on n_3 and it can also be chosen to be minimal to obtain the maximal value). Hence the optimal value n_3^* is

$$n_3^* = 1$$

and

$$\begin{aligned} &N^3_{\mathbf{P}}(n_0,n) \\ &\quad = \max_{n_2} N^3_{\mathbf{P}}\big(3, (n_0, n-n_2-1, n_2, 1)\big) \\ &\quad = \max_{n_2} (n_0+1)(n-n_2) + (n-n_2)n_2 - n_0 + 1 \\ &\quad \le \max_{n_2} (n_0+1)(n-n_2) + (n-n_2+1)n_2 + n_2 \\ &\quad = N^2_{\mathbf{P}}(n_0,n) \end{aligned}$$

Hence a two hidden layers MLP with the same number of hidden units as a three hidden layers MLP has always at least the same number of parameters and therefore three hidden layers are not needed if one aims for maximizing the number of parameters with respect to the number of hidden units. Clearly, more then three hidden layers will yield an analogous result, i.e. to achieve a maximum number of parameters for a given number of hidden units only MLPs with one or two hidden layers must be considered. $\boxed{\text{qed}}$

Proof of Theorem 12

By Corollary 9, Lemma 10 and Proposition 11 for the necessary number n of hidden units the following inequality is necessary

$$\binom{N+n_0}{n_0} \le n(n_0+2),$$

which implies the result of the theorem. $\boxed{\text{qed}}$

Proof of Theorem 13

From Corollary 9, Lemma 10 and Proposition 11 it follows that the following inequality must be fulfilled for the necessary number n of hidden units:

$$N^*_{\mathbf{P}}(n_0,n) \ge \binom{N+n_0}{n_0},$$

where

$$N^*_{\mathbf{P}}(n_0,n) = \begin{cases} (n_0+2)n, & \text{if } n \le n_0+1+\sqrt{4n_0+1-m}, \\ \frac{(n+n_0+3)^2+m-1}{4} - 2(n_0+1), & \text{otherwise,} \end{cases}$$

is the maximum number of network parameters and $m = n_0 + n \mod 2$. Consider the function $H : \mathbb{R} \to \mathbb{R}$ defined by

$$H(x) = \begin{cases} (n_0+2)x, & \text{if } x \le n_0+1+2\sqrt{n_0}, \\ \frac{(x+n_0+3)^2}{4} - 2(n_0+1), & \text{otherwise,} \end{cases}$$

which is continuous, strictly increasing and fulfills $H(n) \ge N^*_{\mathbf{P}}(n_0,n)$ and the necessary number n of hidden units must therefore fulfill $H(n) \ge \binom{N+n_0}{n_0}$. Since H is unbounded there exists a unique $x^* \in \mathbb{R}$ with $H(x^*) = \binom{N+n_0}{n_0}$ and the inequality $H(n) \ge \binom{N+n_0}{n_0}$ is equivalent with $n \ge x^*$. Simple calculations show, that if

$$\big(N+n_0\big)\, n_0 \le (n_0+2)(n_0+1+2\sqrt{n_0})$$

then $x^* = \binom{N+n_0}{n_0} \Big/ (n_0+2)$, otherwise x^* is the solution of

$$\binom{N+n_0}{n_0} = \frac{(x+n_0+3)^2}{4} - 2(n_0+1)$$

which is

$$x^* = 2\sqrt{\binom{N+n_0}{n_0} + 2(n_0+1)} - n_0 - 3.$$

This is the result of the theorem. $\boxed{\text{qed}}$

References

[1] S. Haykin, *Neural Networks*, J. Griffin, Ed. New York: Macmillan College Publishing Company, 1994.

[2] M. Nørgaard, O. Ravn, N. Poulsen, and L. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems*, ser. Advanced Textbooks in Control and Signal Processing, M. Grimble and M. Johnson, Eds. London: Springer-Verlag, 2000.

[3] F. Lewis, J. Huang, T. Parisini, D. Prokhorov, and D. Wunsch, "Guest editorial: Special issue on neural networks for feedback control systems," *IEEE Trans. Neural Networks*, vol. 18, no. 4, pp. 969–972, July 2007.

[4] H. Strobel, *Experimentelle Systemanalyse*. Berlin: Akademie Verlag, 1975.

[5] J. Wernstedt, *Experimentelle Prozessanalyse*. Berlin: Verlag Technik / Oldenbourg Verlag, 1989.

[6] S. Billings, H. Jamaluddin, and S. Chen, "Properties of neural networks with applications to modelling non-linear dynamics," *Int. J. Control*, vol. 55, no. 1, pp. 193–224, 1992.

[7] P. Otto, "Identifikation nichtlinearer Systeme mit künstlichen neuronalen Netzen," *Automatisierungstechnik*, vol. 43, no. 2, pp. 62–68, 1995.

[8] A. Pinkus, "Approximation theory of the MLP model in neural networks," *Acta Numerica*, vol. 9, pp. 143–195, 1999.

[9] A. Barron, "Approximation and estimation bounds for artificial neural networks," *Machine Learning*, vol. 14, pp. 115–133, 1994.

[10] T. Ludermir, A. Yamazaki, and C. Zanchettin, "An optimization methodology for neural network weights and architectures," *IEEE Trans. Neural Networks*, vol. 17, no. 6, pp. 1452–1459, November 2006.

[11] E. Teoh, K. Tan, and C. Xiang, "Estimating the number of hidden neurons in a feedforward network using the singular value decomposition," *IEEE Trans. Neural Networks*, vol. 17, no. 6, pp. 1623–1629, November 2006.

[12] A. Pinkus, Private communication, 2006.

[13] H. Amann and J. Escher, *Analysis II*. Basel - Boston - Berlin: Birkhäuser Verlag, 2001.

[14] ——, *Analysis I*. Basel - Boston - Berlin: Birkhäuser Verlag, 2001.

[15] H. Sagan, *Space-filling curves*, J. Ewing, F. Gehring, and P. Halmos, Eds. New York: Springer-Verlag, 1994.

[16] S. Trenn, "Quantitative analysis of neural networks as universal function approximators," Diplomarbeit, Fakultät für Informatik und Automatisierung, Technische Universität Ilmenau, September 2006.

[17] J. Merker, Private communication, 2006.

[18] R. Abraham, J. Marsden, and T. Ratiu, *Manifolds, Tensor Analysis, and Applications*, 2nd ed., ser. Applied Mathematical Sciences, F. John, J. Marsden, and L. Sirovich, Eds. New York: Springer-Verlag, 1988, no. 75.

Stephan Trenn (born 1980) received his diploma degree in mathematics (Dipl.-Math.) in 2004 and a second diploma degree in Computer Science (Dipl.-Inf.) in 2006, both at Ilmenau University of Technology (TU Ilmenau), Germany. After a six-month research visit to the University of Southampton, UK, he returned to TU Ilmenau in 2005. Since than he has been a graduate assistant at the Institute for Mathematics and is currently working on his Ph.D. thesis in differential algebraic systems and distribution theory.