

# Bounds on the complexity of neural-network models and comparison with linear methods

Kateřina Hlaváčková-Schindler<sup>1,\*</sup> and Marcello Sanguineti<sup>2,§</sup>

<sup>1</sup>*Institute of Computer-Aided Automation, TU Wien Favoritenstrasse 9, A-1040 Wien, Austria*

<sup>2</sup>*Department of Communications, Computer, and System Sciences (DIST), University of Genoa Via Opera Pia 13, 16145 Genova, Italy*

## SUMMARY

A class of non-linear models having the structure of combinations of simple, parametrized basis functions is investigated; this class includes widespread neural networks in which the basis functions correspond to the computational units of a type of networks. Bounds on the complexity of such models are derived in terms of the number of adjustable parameters necessary for a given modelling accuracy. These bounds guarantee a more advantageous tradeoff than linear methods between modelling accuracy and model complexity: the number of parameters may increase much more slowly, in some cases only polynomially, with the dimensionality of the input space in modelling tasks. Polynomial bounds on complexity allow one to cope with the so-called ‘curse of dimensionality’, which often makes linear methods either inaccurate or computationally unfeasible. The presented results let one gain a deeper theoretical insight into the effectiveness of neural-network architectures, noticed in complex modelling applications.

Copyright © 2003 John Wiley & Sons, Ltd.

**KEY WORDS:** non-linear models; polynomially bounded complexity; curse of dimensionality; neural networks.

## 1. INTRODUCTION

Linear models (see, e.g. References [1, 2]) exhibit limitations in many applications in which strong nonlinearities are present. Recently, learning-based non-linear modelling methodologies have emerged as promising alternatives aimed at implementing identification and control structures for adaptive control of non-linear dynamical systems (see, e.g. References [3–5]).

---

\*Correspondence to: K. Hlaváčková-Schindler, Institute of Computer-Aided Automation, TU Wien Favoritenstrasse 9, A-1040 Wien, Austria

†E-mail: icat@prp.tuwien.ac.at

§E-mail: marcello@dist.unige.it

Contract/grant sponsor: Austrian Science Fund; contract/grant number: FWF P13385-INF.

Contract/grant sponsor: GA CR; contract/grant numbers: 01/99/0092; 201/02/0428.

Contract/grant sponsor: Italian Ministry of University and Research; contract/grant number: MM09198819.

Contract/grant sponsor: National Research Council of Italy – CNR-Agenzia 2000; contract/grant number: CNRC00CAE4.

Parametrized classes of non-linear input–output mappings with powerful approximating capabilities, such as neural networks, have become attractive tools for modelling complex systems on the basis of input–output data (see, e.g. References [6, 7] and the references therein). The *density* property (in the neural-network parlance also called *universal approximation* property), which ensures the possibility of approximating arbitrarily well all ‘reasonable’ functions encountered in applications, has been proven for neural networks with various types of architectures and computational units (e.g. radial-basis-functions and perceptrons; see References [8, 9] and the references therein) and is a necessary condition to guarantee an arbitrary modelling accuracy.

However, density does not ensure feasibility: in complex modelling tasks, classes of models are required to allow a good compromise between the accuracy of approximation and the ‘complexity’ necessary to achieve such an accuracy. For linear and neural-network methods, the complexity can be expressed by the size of a properly defined parameter vector (e.g. the degree of a polynomial or the number of knots of a fixed-knots spline in the linear case, the number of hidden units of a neural network in the non-linear context, etc.). In classical linear methods, the adjustable parameters are the coefficients of the linear combinations of fixed basis functions, whereas in neural-network architectures they also include the weights in computational units. High-dimensional modelling is often unfeasible because of the *curse of dimensionality* [10]: the number of parameters required to obtain a desired modelling accuracy may grow exponentially with the number  $d$  of variables of the non-linear mappings to be modelled. On the other hand, applications have shown the possibility of effectively using neural networks with a moderate number of parameters in approximation and optimization tasks dependent on a large number of variables, for which traditional linear methods are inefficient (see References [11–15] and the references therein). These results have motivated the development of a theoretical framework to investigate the capabilities of neural architectures and to compare them with classical linear methods (see, e.g. References [16, 17] and the references therein).

This paper is written with three objectives. The first is to show how methods from approximation theory can be used to bound the complexity of different models by introducing a comprehensive formulation that encompasses linear techniques and a variety of widespread non-linear ones (neural networks, radial basis networks, wavelets, hinging hyperplanes, fuzzy models, etc.). Toward this end, we focus on the common features of such non-linear techniques: we show that they have the same geometrical structure, and we point out the basic differences between them and linear methods. The second objective is to gain theoretical insights into the effective performances of neural networks in modelling complex non-linear mappings. The third objective is to make some comparisons between linear and neural-network models and to show some advantages of the latter, using *a priori* knowledge to define suitable smoothness classes for the non-linearities to be modelled.

We formalize modelling tasks in terms of approximation theory in functional spaces. A given modelling task determines the choice of an *ambient functional space*, the *a priori* assumptions about the systems to be modelled identify a *hypothesis class*, and the choice of a class of models corresponds to the choice of an *approximation scheme*. Multiple models can be used for the same system in different operating environments, which identify different hypothesis classes.

We give bounds on the complexity of linear and non-linear models of the neural-network type, for different hypothesis classes (e.g. corresponding to different operating environments for the same system) and in different ambient spaces, each associated with a certain modelling task. Bounds on model complexity are determined by estimating the rate of decrease in the modelling

error as a function of the number of adjustable parameters. We provide lower bounds on the worst-case modelling error by linear methods: such bounds are larger than the upper bounds on the same error by non-linear models of the neural-network type. In all such cases, network modelling architectures outperform linear methods. We also exhibit neural-network models characterized by moderate complexity, in the sense that the number of parameters necessary for a desired accuracy grows at most polynomially with the dimension of the input space, in certain modelling tasks for which the use of linear methods is computationally unfeasible. Our estimates give theoretical insights into applicative results, showing that the use of neural-network models instead of linear ones often allows a significant reduction in the number of parameters required for a given modelling accuracy.

The paper is organized as follows. Section 2 deals with basic notations and definitions and introduces a mathematical framework to describe the different structures of linear and neural-network models. Section 3 shows how tools from approximation theory can be used to develop a procedure of complexity-based model selection. Section 4 presents a variety neural-network models with moderate complexity. Section 5 considers linear methods and provides lower bounds that imply a poor efficiency of such methods in certain modelling tasks. As an example of application of the approach developed in previous sections, in Section 6 a comparison between linear and neural-network models (represented by perceptron networks) is made. Section 7 contains some concluding remarks.

## 2. THE STRUCTURES OF LINEAR AND NEURAL-NETWORK MODELS

Both linear and neural-network models can be represented as sets of parametrized mappings in which the parameters have to be updated in such a way that the modelling error is decreased (e.g. by gradient-type descent methods like back-propagation for neural networks). In classical linear models, the adjustable parameters are the coefficients of the linear combinations of fixed basis functions, whereas in neural-network architectures the parameters also include the weights in computational units. In this section, we describe a mathematical framework to represent these two different kinds of parametrizations.

*Linear models* are based on the use of linear combinations of a set of fixed basis functions. Such linear combinations ‘span’ a finite-dimensional subspace, of dimension equal to  $n$ ; the dimension is  $n$  when the fixed basis functions are linearly independent. Thus, the free parameters are the  $n$  coefficients of the linear combinations of the fixed basis functions. For example, the subspace of all polynomials of order at most  $n - 1$  is spanned by the first  $n$  elements of the set  $\{x^{i-1}: i \in \mathcal{N}_+\}$ , where  $\mathcal{N}_+$  denotes the set of positive integers. As this corresponds to the use of linear combinations of a certain number of fixed basis functions, the term *fixed-basis models* is used, too.

In recent years, there has been growing interest in non-linear models such as wavelets, radial basis networks, kernel estimators, B-splines, sigmoidal neural networks, hinging hyperplanes, projection pursuit regression, etc. (see Reference [18]). All these models share a common feature: they are built as combinations of a number of nonlinearities corresponding to parametrized basis functions with a fixed structure in which a certain number of ‘free’ parameters have to be adjusted. Typically, such basis functions are obtained by parametrizing a single ‘mother’ function, the choice of which determines the properties of a model [19]. In the following, we introduce a general class of non-linear models that encompasses such modelling techniques.

We call  $\phi$ -network models, with computational units  $\phi: \mathcal{R}^p \times \mathcal{R}^d \rightarrow \mathcal{R}$  ( $\mathcal{R}$  denotes the set of real numbers) and a single linear output unit, the non-linear mappings  $\sum_{i=1}^n w_i \phi(a_i, \cdot)$ , where  $a_i \in A \subseteq \mathcal{R}^p$ , and  $p$  and  $d$  represent the dimensions of a *parameter space* and an *input space*, respectively. The function  $\phi$  plays the role of the ‘mother’ function mentioned above. As in applications all inputs are bounded, the set  $A$  is a bounded subset of  $\mathcal{R}^d$ ; without loss of generality, we consider  $A$  to be the unit cube  $[0, 1]^d$ . Widespread types of  $\phi$ -network models are perceptron networks and radial basis functions networks.

A perceptron with an activation function  $\psi: \mathcal{R} \rightarrow \mathcal{R}$  computes functions of the form  $\phi((v, b), x) = \psi(v \cdot x + b): \mathcal{R}^{d+1} \times \mathcal{R}^d \rightarrow \mathcal{R}$ , where  $v \in \mathcal{R}^d$  is an *input weight* vector and  $b \in \mathcal{R}$  is a *bias*. The most common activation functions in perceptrons are *sigmoidals*, i.e. bounded measurable functions  $\sigma: \mathcal{R} \rightarrow [0, 1]$  such that  $\lim_{t \rightarrow -\infty} \sigma(t) = 0$  and  $\lim_{t \rightarrow +\infty} \sigma(t) = 1$ . One can use both continuous sigmoidals (like the logistic sigmoid  $1/(1 + e^{-t})$  or the hyperbolic tangent) and the discontinuous Heaviside function  $\mathfrak{H}$ , defined as  $\mathfrak{H}(t) = 0$  for  $t < 0$  and  $\mathfrak{H}(t) = 1$  for  $t \geq 0$ . By  $P_d(\psi) = \{f: [0, 1]^d \rightarrow \mathcal{R}; f(x) = \psi(v \cdot x + b), v \in \mathcal{R}^d, b \in \mathcal{R}\}$  we denote the set of functions on  $[0, 1]^d$  computable by  $\psi$ -perceptrons. A *radial basis function* (RBF) unit with a radial (even) function  $\psi: \mathcal{R} \rightarrow \mathcal{R}_+$  ( $\mathcal{R}_+$  is the set of positive real numbers) computes  $\phi((v, b), x) = \psi(b \|x - v\|)$ , where  $v \in \mathcal{R}^d$  is a *centroid*,  $b \in \mathcal{R}_+$  is a *width* and  $\|\cdot\|$  is a norm on  $\mathcal{R}^d$ . By  $F_d(\psi) = \{f: [0, 1]^d \rightarrow \mathcal{R}; f(x) = \sum_{i=1}^n w_i \psi(b_i \|x - v_i\|), v_i \in \mathcal{R}^d, w_i, b_i \in \mathcal{R}, n \in \mathcal{N}_+\}$  we denote the set of functions on  $[0, 1]^d$  computable by  $\psi$ -RBF networks. The standard choice of a radial function is the Gaussian function.

Models having the structure of  $\phi$ -networks can be mathematically formalized as linear combinations of at most  $n$  elements of the set  $G_\phi = \{\phi(a, \cdot); a \in A \subseteq \mathcal{R}^p\}$ , representing the set of functions computable by hidden network units. The set of input/output mappings computed by a  $\phi$ -network with a single linear output unit and  $n$  hidden units, each computing the function  $\phi$ , is given by  $\text{span}_n G_\phi = \{\sum_{i=1}^n w_i g_i; w_i \in \mathcal{R}, g_i \in G_\phi\}$ . Such a set has the geometrical structure of the union of finite-dimensional subspaces generated by all  $n$ -tuples of elements of  $G_\phi$  [17]. Thus, non-linear models of the neural-network type correspond to the unions of finite-dimensional subspaces generated by hidden unit functions. As such functions play the role of a variable basis, in contrast to the fixed basis corresponding to the linear case, the term *variable-basis models* is used, too. Commonly used RBF and perceptron models can be formalized as follows: the sets  $\text{span}_n P_d(\psi)$  and  $\text{span}_n F_d(\psi)$  represent the sets of functions on  $[0, 1]^d$  computable by  $\psi$ -perceptron models and  $\psi$ -RBF models, respectively, with  $n$  hidden units;  $\text{span} P_d(\psi)$  and  $\text{span} F_d(\psi)$  denote the set of functions on  $[0, 1]^d$  computable by such networks, resp., with any number of hidden units.

Note that variable-basis models include not only neural networks and radial basis functions, but also many other non-linear models (see the references in References [17, 18]), such as trigonometric polynomials with free frequencies, free-node splines, etc. Multilayer networks with a single linear output unit and  $n$  units in the last hidden layer can also be represented by this class of models; they compute functions from  $\text{span}_n G$ , with  $G$  dependent on the numbers of units in the previous hidden layers.

### 3. COMPLEXITY-BASED MODEL SELECTION

Modelling can be represented as an *approximation task* in a functional space (we call it the *ambient space*), which is defined by the application: for example, the space of continuous functions with the supremum norm (when asymptotic properties of the model have to be

fulfilled, like stability or convergence of estimation errors), the space of square-integrable functions with the  $\mathcal{L}_2$  norm, etc. In general, by  $(X, \|\cdot\|)$  we denote a normed linear space corresponding to the ambient space considered (when the norm is clear from the context, we write  $X$  only). We recall that a *Banach space* is a normed linear space that is complete and a *Hilbert space* is a Banach space with a norm induced by an inner product, i.e.  $\|f\| = \sqrt{f \cdot f}$  [20, pp. 126, 201]. Let  $M, Y$  be two subsets of  $(X, \|\cdot\|)$ :  $M$  represents a class of models and  $Y$  is a set of mappings to be modelled. The error made in modelling the mapping  $f \in Y$  by the model  $g \in M$  is measured by the distance  $\|f - g\|$ .

When linear and  $\phi$ -network models are used, *model complexity* can be expressed by the size of a properly defined parameter vector: the degree of a polynomial or the number of knots of a fixed-knots spline in the linear case, the number of computational units of a neural network, etc. One naturally expects that the larger the number of parameters used to describe a model, the more accurate the description, as the model itself is more flexible. However, implementation feasibility requires a tradeoff between approximation accuracy and model complexity (i.e. the number of parameters), particularly for high-dimensional modelling tasks. Classes of models have to be chosen such that they guarantee an arbitrarily high modelling accuracy when one uses a moderate number of parameters, i.e. models that have a *limited complexity*. *Complexity-based multiple models selection* entails the following steps:

- (1) Identifying different *operating environments* (corresponding, for instance, to the failures of one or more subsystems, to the presence or absence of disturbances, to the change of certain parameters).
- (2) Exploiting the *a priori* assumptions to restrict the sets of mappings to be approximated. Various *hypothesis classes* can be defined, one for each operating environment. As different environments are often represented by non-linear mappings with different properties, a single model may not be adequate enough to identify the changes in a dynamic system. In such cases, the use of multiple models is mandatory.

Then, for each hypothesis class, the following tasks should be accomplished:

- (3) Estimating the *number of parameters* required by different modelling networks to achieve the desired modelling accuracy.
- (4) Choosing the structure (e.g. linear, non-linear of the neural-network type, etc.) that allows the desired modelling accuracy with the smallest number of parameters. In particular, in tasks involving high-dimensional input spaces the curse of dimensionality has to be avoided.
- (5) Tuning the parameters.

In the remainder of this section, we introduce a mathematical framework to formalize steps (3) and (4) and to compare the complexities of linear and neural-network models in terms of approximation theory.

As both linear and neural-network models are obtained by combining elements from a set of (fixed or variable, resp.) basis functions, the number of such functions can be considered as a measure of model complexity. Classes of models with increasing complexity can be represented as the union of a nested sequence  $\{M_n; n \in \mathcal{N}_+\}$  of sets of functions spanned by an increasing number  $n$  of basis functions. In applications, the rate of decrease in the modelling error in dependence of the number of basis functions has to be high enough so that models with a moderate complexity may guarantee a sufficient accuracy.

This can be studied with tools from approximation theory by introducing the concept of *worst-case modelling error* from a class  $M$  of models, used to model elements from a hypothesis class  $Y$ . Such an error is mathematically formalized by the *deviation of  $Y$  from  $M$* , defined as

$$\delta(Y, M) = \sup_{f \in Y} \|f - M\| = \sup_{f \in Y} \inf_{g \in M} \|f - g\|$$

Thus we are interested in the rate of decrease of the deviations  $\delta(Y, M_n)$  for increasing values of  $n$ . If  $\bigcup_{n \in \mathcal{N}} M_n$  is dense in  $X$ , then, for any  $f \in X$ , the sequence  $\{\|f - M_n\|: n \in \mathcal{N}_+\}$  converges to 0. But, in practical applications, this convergence has to be sufficiently fast to guarantee any desired modelling accuracy for  $n$  small enough, to allow models from  $M_n$  to have moderate numbers of parameters. For functions of  $d$  variables, it often happens that deviations are of order  $O(1/n^{1/d})$ . In such a case, to achieve a modelling accuracy within  $\varepsilon$ , approximating functions with complexity of order  $O(1/\varepsilon^d)$  are needed. Such exponential dependence of complexity on the number of variables determines the so-called *curse of dimensionality* [10].

Recalling that linear methods correspond to the use of finite-dimensional subspaces  $X_n$  of the ambient space  $X$  (i.e.  $M_n = X_n$ ), to compare neural networks with linear techniques we shall compare the deviation  $\delta(Y, \text{span}_n G)$  with the deviation of  $Y$  from the ‘best’ approximating linear subspace  $X_n$  of  $X$ . This is formalized by the concept of  *$n$ -width* of  $Y$  in  $X$ , defined as (see, e.g. Reference [21, pp. 1, 2])

$$d_n(Y) = \inf_{X_n} \delta(Y, X_n) = \inf_{X_n} \sup_{f \in Y} \inf_{g \in X_n} \|f - g\|$$

where the left-most infimum is taken over all  $n$ -dimensional subspaces  $X_n$  of  $X$ , spanned by  $n$  linearly independent basis functions.

Estimating deviation and  $n$ -width in dependence of the number  $n$  of basis functions and the dimension  $d$  of the input space (i.e. the number of variables) is essential to the choice of a model with limited complexity. Approximation theory provides lower bounds on  $n$ -width that often result in poor performances of linear methods (see, e.g. References [17, 21]). When the parameters of the basis functions are continuously adjusted to the function to be approximated, it is possible to extend the lower bounds on  $n$ -width from linear methods to non-linear models. This was done in Reference [22] by defining a proper *continuous non-linear  $n$ -width*. However, such lower bounds cannot be applied *a priori* to neural networks, as it has recently been shown [23] that, for most standard types of such networks (e.g. Gaussian radial-basis-functions and Heaviside perceptrons) the best approximation cannot be achieved in a continuous way. Roughly speaking, neural-network models might exploit the non-linearity-in-parameters to achieve the same modelling accuracy as attained by linear methods but with a reduced complexity. Indeed, experimental results have shown that neural-network adaptive architectures with relatively few adjustable parameters can obtain surprisingly good performances.

Next sections give a theoretical insight into the effectiveness of neural networks in complex modelling tasks, in which traditional linear models are often unable to perform satisfactorily. We make comparisons between estimates of  $d_n(Y)$  and  $\delta(Y, \text{span}_n G)$ , i.e. between the best achievable modelling accuracy of mappings from a hypothesis class  $Y$  by linear models with  $n$  basis functions and the best non-linear modelling accuracy of the same mappings by elements of network models represented by  $\text{span}_n G$ .

## 4. UPPER BOUNDS ON THE COMPLEXITY OF NEURAL-NETWORK MODELS

From a qualitative point of view, the main limitation of linear models consists in the use of linear combinations of certain *fixed* functions: being fixed, such functions cannot be adapted to mappings representing different systems to be modelled or different operating environments of the same system. In contrast to this, non-linear approximators of the variable-basis type such as neural networks take advantage of their adjustable parameters (e.g. the centres and the weight matrices in radial basis functions, the frequencies and phases in trigonometric basis functions, the thresholds in sigmoidal basis functions, etc.), which allow one to choose the basis functions adaptively, i.e. depending on the input–output mapping to be approximated.

The description of various sets of  $d$ -variable functions that do not exhibit the curse of dimensionality in variable-basis approximation can be derived by exploiting the following result of non-linear approximation theory, obtained by Maurey, Jones and Barron (see References [24, 25, 16]).

*Theorem 1* (Maurey [24], Jones [25], Barron [16])

Let  $(X, \|\cdot\|)$  be a Hilbert space,  $G$  its bounded subset,  $s_G = \sup_{g \in G} \|g\|$ , and  $f \in cl \text{ conv } G$ . Then, for every positive integer  $n$ ,  $\|f - \text{conv}_n G\| \leq \sqrt{(s_G^2 - \|f\|^2)/n}$ .

This theorem gives an upper bound of order  $O(n^{-1/2})$ , independently of the number  $d$  of variables of functions in  $X$ . As  $\text{conv}_n G \subseteq \text{span}_n G$ , the upper bound from Theorem 1 also applies to rates of approximation by  $\text{span}_n G$ . However, when  $G$  is not closed under multiplication by scalars,  $cl \text{ conv } G$  is a proper subset of  $cl \text{ span } G$ . Thus, the density of  $\text{span } G$  in  $(X, \|\cdot\|)$  does not guarantee that Theorem 1 can be applied to all elements of  $X$ . By replacing  $G$  with  $G(c) = \{wg; w \in \mathcal{R}, |w| \leq c, g \in G\}$ , for any  $c > 0$  one gets  $\text{conv}_n G(c) \subseteq \text{span}_n G(c) = \text{span}_n G$ , and so one can apply Theorem 1 to all elements of  $\bigcup_{c \in \mathcal{R}_+} cl \text{ conv } G(c)$ . This approach was formulated in Reference [26] in terms of a norm ‘tailored’ to a set  $G$ , called  $G$ -variation (variation with respect to  $G$ ) and defined [26] as the Minkowski functional of the set  $cl \text{ conv } (G \cup -G)$ , i.e.  $\|f\|_G = \inf \{c \in \mathcal{R}_+; f/c \in cl \text{ conv } (G \cup -G)\}$ .

$G$ -variation is a norm on  $\{f \in X; \|f\|_G < \infty\} \subseteq X$ . To simplify the notation we write only  $\|\cdot\|_G$  whenever it is clear with respect to which norm  $G$ -variation is defined. Intuitively,  $\|f\|_G$  shows us how much the set  $G$  should be dilated, so that  $f$  may be in the closure of the convex symmetric hull of the dilated set.  $G$ -variation is a generalization of two concepts: the notion of total variation used in integration theory (for functions of one variable, it coincides up to a constant with total variation) and  $l_1$  norm (see Reference [17] for details).

The following upper bound is a reformulation of Theorem 1 in terms of  $G$ -variation [14].

*Theorem 2* (Kůrková)

Let  $(X, \|\cdot\|)$  be a Hilbert space,  $G$  its bounded subset,  $s_G = \sup_{g \in G} \|g\|$ . Then, for every  $f \in X$  and every positive integer  $n$ ,  $\|f - \text{span}_n G\| \leq \sqrt{((s_G \|f\|_G)^2 - \|f\|^2)/n}$ .

Let  $B_r(\|\cdot\|')$  denote the ball of radius  $r > 0$  with respect to a norm  $\|\cdot\|'$ , i.e.  $B_r(\|\cdot\|') = \{f \in (X, \|\cdot\|); \|f\|' \leq r\}$ . As  $\text{conv}(G \cup -G) = \text{conv } G(1)$ , we have  $\|f\|_G = \inf \{c \in \mathcal{R}_+; f \in cl \text{ conv } G(c)\}$  and the unit ball  $B_1(\|\cdot\|_G)$  in  $G$ -variation is equal to  $cl \text{ conv}(G \cup -G)$ . Thus Theorem 2

implies the following upper bound on deviation of balls in  $G$ -variation. For a subset  $G$  of  $X$ ,  $G^0 = \{g^0 = g/\|g\| : 0 \neq g \in G\}$  denotes the set of its normalized elements.

*Corollary 1* (Kůrková and Sanguinetti [17])

Let  $(X, \|\cdot\|)$  be a Hilbert space,  $G$  its bounded subset, and  $s_G = \sup_{g \in G} \|g\|$ . Then, for every positive integer  $n$  and every  $r > 0$ ,  $\delta(B_r(\|\cdot\|_G), \text{span}_n G) \leq rs_G/\sqrt{n}$ , and  $\delta(B_r(\|\cdot\|_{G^0}), \text{span}_n G) \leq r/\sqrt{n}$ .

Corollary 1 can be used as explained in the following to bound the complexity of variable-basis models. Suppose, for example, that a system-theoretic analysis guarantees the existence of a non-linear input–output mapping or of a mapping between measurements and controls. If one can prove that such non-linearities belong to balls of fixed radius in  $G_\phi$ -variation, for a certain  $\phi$ -network, then any desired modelling accuracy  $\varepsilon$  can be obtained by using  $\phi$ -network models with  $n \leq r^2/\varepsilon^2$  basis functions (i.e. at most  $n$  adjustable network units), independently of the number  $d$  of variables of the mappings to be modelled. This is particularly interesting in high-dimensional problems, in which the use of linear methods is often made computationally unfeasible as too many parameters are required, and assumes major importance when the input space is of large dimension, such as in the case of high-order plants [7, p. 43].

When  $G$  corresponds to the set  $G_\phi$  of functions computable by hidden units, Corollary 1 applies to neural-network models. The bound of order  $O(n^{-1/2})$ , when combined with the availability of gradient-like training algorithms (based on the output error of the network and well-suited to parallel implementation) for adjusting parameters, makes  $\phi$ -network models very attractive. Further theoretical results on upper bounds on the complexity of  $\phi$ -network models can be found in Reference [27].

The practical importance of Corollary 1 relies on the possibility of embedding a ball of some radius in  $G$ -variation in the set representing the hypothesis class of the modelling problem at hand. In other words, Corollary 1 becomes useful only if sets  $B_r(\|\cdot\|_{G_\phi})$ , where  $G_\phi$  corresponds to a  $\phi$ -network model, contain functions of interest in applications. In the following we give some examples, for various hypothesis classes defined by smoothness conditions in terms of the Fourier transform and for  $\phi$ -networks with various types of basis functions.

*Example 1 (Trigonometric polynomials with free frequencies)*

Using the results from Reference [25], one can see that functions  $f : \mathcal{R}^d \rightarrow \mathcal{R}$  satisfying  $\int_{\mathcal{R}^d} |\hat{f}(\omega)| d\omega \leq c$  for some  $c > 0$ , where  $\hat{f}$  is the Fourier transform of  $f$ , are contained in a ball of a suitable radius in variation with respect to  $P_d(\sin)$ , where  $P_d(\sin)$ -variation is defined in  $L_2(\Omega)$ ,  $\Omega \subset \mathcal{R}^d$ , with the  $\|\cdot\|_2$  norm. Note that models of the form  $P_d(\sin)$  correspond to trigonometric polynomials with free frequencies.

*Example 2 (Ramp perceptron models)*

Similarly, on the basis of Reference [28] one concludes that the sets of functions  $f : \mathcal{R}^d \rightarrow \mathcal{R}$  satisfying  $\int_{\mathcal{R}^d} |\omega|^2 |\hat{f}(\omega)| d\omega \leq c$  for some  $c > 0$ , can be embedded into balls in variation with respect to ramp perceptrons, i.e. perceptrons using as a computational unit the *ramp* function  $\kappa$ , defined as  $\kappa(t) = t\mathcal{I}(t)$ , i.e.  $\kappa(t) = 0$  for  $t < 0$  and  $\kappa(t) = t$  for  $t \geq 0$ .

*Example 3 (Sigmoidal perceptron models)*

In Reference [16], sets of functions  $\Gamma_c^d = \{f : \mathcal{R}^d \rightarrow \mathcal{R} \mid \int_{\mathcal{R}^d} |\omega| |\hat{f}(\omega)| d\omega \leq c\}$ ,  $c > 0$ , were investigated, where  $\hat{f}$  is the Fourier transform of  $f$ , and  $|\omega| = \sqrt{\omega \cdot \omega}$  denotes the  $l_2$  norm of

the frequency  $\omega$ . Reformulation of the results from [16, pp. 935, 941] implies that  $\hat{\Gamma}_c^d = \{f|_{B_1^d}: f \in \Gamma_c^d\}$ , where  $B_1^d$  denotes the unit ball in  $\mathcal{R}^d$  with the  $l_2$  norm, is contained in the ball  $B_{2c}(\|\cdot\|_{P_d(\sigma)})$ , where  $P_d(\sigma)$ -variation is defined in  $L_2(B^d)$  with the  $\|\cdot\|_2$  norm. Multi-variable sets of functions that are of interest in modelling tasks and are contained in  $\Gamma_c^d$ , for some  $c > 0$ , were described in Reference [16].

The above-mentioned results are summarized in Table I; other  $\phi$ -network models with the same bound of order  $1/\varepsilon^2$  on complexity are discussed in Reference [29]. In general, to guarantee a limited model complexity, different approximating families have to be used in different operating environments that can result from various events (sudden changes in parameter values, external disturbances, faults such as failures of sensors, etc.). When an input/output representation is available, such environments are mathematically represented by hypothesis classes having different smoothness characteristics. Thus Table I gives an insight into the intuition that the most convenient model depends on each hypothesis class.

*Example 4 (Sobolev spaces as hypothesis classes)*

Let us consider the Sobolev spaces  $W_p^s(\Omega)$ ,  $1 \leq p < \infty$ ,  $\Omega \subset \mathcal{R}^d$ , of measurable functions that, together with their partial derivatives up to order  $s$ , have an integrable  $p$ -th power. It is known [19] that the best approximation of functions belonging to  $W_2^s(\Omega)$ , when the error is measured in  $L_2$  norm, is an orthogonal projection on a linear subspace. However, it can be shown [21, pp. 232–233] that, for the hypothesis class  $W_p^s(\Omega)$  with  $p < 2$ , the optimal projection on an  $n$ -dimensional subspace converges at a rate of order  $O(n^{-s'/d})$ , where  $s' = s - 1/p + 1/2$ , whereas there exist non-linear approximators exhibiting a higher convergence rate of order  $O(n^{-s/d})$  [19]. The smaller  $p$ , the more significant the difference between these two rates. From a practical point of view, functions in  $W_p^s(\Omega)$  with small  $p$  values have sparse singularities: the rate of linear models for mappings with local jumps and spikes is slower than for uniformly smooth functions. Roughly speaking, variable-basis models are preferable in these cases [19].

It is worth noting that the upper bound on the modelling error by  $\text{span}_n G_\phi$  given by Theorems 1 and 2 and Corollary 1 can be applied not only to neural-network models but also to various other classes of widespread non-linear models, such as free-node splines, trigonometric polynomials with free frequencies, etc. [17].

Table I. Examples of neural-network models with complexity bounded by  $r/\varepsilon^2$  in the ambient space  $L_2(\Omega)$ ,  $\Omega \subset \mathcal{R}^d$  compact, for some hypothesis classes.

Neural-network model $\text{span}_n G_\phi$	Hypothesis class (subset of $B_r(\ \cdot\ _G)$ )	Bound on complexity
$G_{\sin} = \{f(\mathbf{x}) = \sin(\mathbf{x} \cdot \mathbf{w} + \theta), \mathbf{w} \in \mathcal{R}^d, \theta \in \mathcal{R}\}$	$\{f : \int_{\mathcal{R}^d}  \hat{f}(\omega)  d\omega < r\}$	$\frac{r^2}{\varepsilon^2}$
$G_\sigma = \{f(\mathbf{x}) = \sigma(\mathbf{x} \cdot \mathbf{w} + \theta), \mathbf{w} \in \mathcal{R}^d, \theta \in \mathcal{R}\}$	$\{f : \int_{\mathcal{R}^d}  \omega   \hat{f}(\omega)  d\omega < r\}$	$\frac{r^2}{\varepsilon^2}$
$G_\kappa = \{f(\mathbf{x}) = \kappa(\mathbf{x} \cdot \mathbf{w} + \theta), \mathbf{w} \in \mathcal{R}^d, \theta \in \mathcal{R}\}$	$\{f : \int_{\mathcal{R}^d}  \omega ^2  \hat{f}(\omega)  d\omega < r\}$	$\frac{r^2}{\varepsilon^2}$

## 5. LOWER BOUNDS ON THE COMPLEXITY OF LINEAR MODELS

We start this section with an example.

*Example 5 (Comparison between linear and neural-network models of sigmoidal perceptron type)*

The first result pointing out the advantages of neural-network approximation over linear methods is the theoretical comparison in Reference [16] of the worst-case errors in linear and neural-network approximations. Applying Theorem 1, classes of multi-variable functions were described in Reference [16]; for such classes the  $L_2$  approximation error by one-hidden-layer sigmoidal perceptron networks is bounded from above by a quantity of order  $O(1/\sqrt{n})$ , where  $n$  is the number of network hidden units. It was also proven that the  $L_2$  error of the best linear approximator is bounded from below by a quantity of order  $O(1/(d\sqrt[n]{n}))$ , where  $n$  is the dimension of the linear approximating subspace and  $d$  is the number of variables of the functions to be approximated. More precisely, in  $(L_2([0, 1]^d), \|\cdot\|_2)$ ,

$$d_n(\bar{\Gamma}_c^d) \geq \frac{ca}{d\sqrt[n]{n}} \quad (1)$$

where  $a \geq 1/(8\pi e^{\pi-1})$  and  $\bar{\Gamma}_c^d = \{f|_{[0,1]^d}: f \in \Gamma_c^d\}$ . Instead, in  $(L_2(B_1^d), \|\cdot\|_2)$

$$\delta(\tilde{\Gamma}_c^d, \text{span}_n P_d(\sigma)) \leq \frac{2c}{\sqrt[n]{n}} \quad (2)$$

where  $\tilde{\Gamma}_c^d = \{f|_{B_1^d}: f \in \Gamma_c^d\}$  and  $B_1^d$  denotes the unit ball in  $\mathscr{R}^d$  with the  $l_2$  norm.

The asymptotic rates of bounds (1) and (2) are influenced by the dependence of  $c$  on the dimension  $d$ . Hypothesis classes for which such dependence is linear or, more generally, polynomial, are described in Reference [16]. For such classes, which include various sets of functions of interest in applications, the lower bound (1) implies the curse of dimensionality, whereas the upper bound (2) grows only polynomially with increasing dimension  $d$ . Thus, when  $c$  grows polynomially with  $d$ , linear models with a number of basis functions growing exponentially with  $d$  are necessary to guarantee an accuracy  $\varepsilon$  (as (1) implies  $n \geq ((ca)/(d\varepsilon))^d$ ). Instead, the number of network computational units necessary for the same accuracy has to grow at most polynomially with  $d$  (as (2) implies  $n \leq (2c/\varepsilon)^2$ ). In these cases, the advantageous behaviour of sigmoidal neural-network models contrasts with the unfeasibility of linear methods in high-dimensional modelling tasks.

It was shown in Reference [17] that, for linear models, lower bounds of the same order as (1) can be derived for more general classes, for which the upper bound (2) on modelling accuracy by neural networks holds. Moreover, a general framework was developed in Reference [17] for the description of sets of functions the approximation of which by linear methods may exhibit the curse of dimensionality, whereas the rate of approximation by  $\text{span}_n G_\phi$  depends polynomially on  $d$  for suitable classes of  $\phi$ -networks. For such sets of functions, linear methods are outperformed by  $\phi$ -network models.

In the following, we use tools from metric entropy theory and derive lower bounds on the worst-case error in modelling by linear methods various classes of mappings. We show that linear models are outperformed by neural-network models for such hypothesis classes.

Recall that, for  $\varepsilon > 0$ , the  $\varepsilon$ -covering number of a subset  $K$  of a normed linear space  $(X, \|\cdot\|)$  is defined as  $\text{cov}_\varepsilon(K, \|\cdot\|) = \min\{m \in \mathcal{N}_+: K \subseteq \bigcup_{i=1}^m B_\varepsilon(f_i, \|\cdot\|), f_i \in X, i = 1, \dots, m\}$  if the set over

which the minimum is taken is nonempty, otherwise  $\text{cov}_\varepsilon(K, \|\cdot\|) = +\infty$ . The  $\varepsilon$ -metric entropy of  $K$  is defined as  $H_\varepsilon(K, \|\cdot\|) = \log_2 \text{cov}_\varepsilon(K, \|\cdot\|)$ . When it is clear from the context which norm is considered, we will simply write  $\text{cov}_\varepsilon(K)$  and  $H_\varepsilon(K)$  instead of  $\text{cov}_\varepsilon(K, \|\cdot\|)$  and  $H_\varepsilon(K, \|\cdot\|)$ , respectively. As pointed out by Lorentz [30, pp. 150, 163], both the  $n$ -width of a set and its metric entropy measure the ‘size’ of that set, although in different ways.

In general, in infinite-dimensional spaces there exist sets whose  $\varepsilon$ -entropies increase arbitrarily fast for  $\varepsilon \rightarrow 0$  [31, 6.8.3]. However, when a compact set of functions to be approximated is characterized by suitable bounds on the rate of increase of its entropy, some lower bounds on the  $n$ -width can be derived. The following proposition is a reformulation in terms of  $n$ -width of Reference [31, Theorems 6.8.31 and 6.8.33]. When suitable estimates of the  $\varepsilon$ -entropy are available, linear methods imply a limited accuracy.

### Proposition 1

Let  $G$  be a compact subset of an infinite dimensional Banach space  $(X, \|\cdot\|)$ .

- (i) If  $a \log_2^\beta(1/\varepsilon) \leq H_\varepsilon(G) \leq b \log_2^\beta(1/\varepsilon)$  for some constants  $a, b > 0$  and some  $\beta > 1$ , then there exists an integer  $n_1$  such that, for all  $n \geq n_1$ ,  $d_n(G) \geq \rho^{n^{1/(\beta-1)}}$ , where  $\rho$  is a constant independent of  $n$  and  $0 < \rho < 1$ .
- (ii) If  $a \log_2^\beta(1/\varepsilon) \leq \log_2 H_\varepsilon(G) \leq b \log_2^\beta(1/\varepsilon)$  for some constants  $a, b > 0$  and some  $\beta > 1$ , then there exists an integer  $n_2$  such that, for all  $n \geq n_2$ ,  $d_n(G) \geq \rho^{(lnn)^{1/\beta}}$ , where  $0 < \rho < 1$  is a constant independent of  $n$ .

We exploit Proposition 1 to prove limitations of linear methods for certain sets of analytic functions, in terms of lower bounds on their  $n$ -width. The use of linear models based on various types of orthogonal basis functions was studied, for example, in Reference [32] in the case where the systems to be modelled are analytic. It was proven in Reference [32] that Laguerre basis functions and Kautz basis functions are convenient for well-damped and lightly damped systems, resp. However, this requires information about the dominating poles of a plant, which might not be available. We denote by  $A^d(S)$  the set of all real-valued analytic functions of  $d$  variables defined on a bounded set  $S$  in the  $d$ -dimensional complex space.  $A^d(S)$  is a  $d$ -dimensional linear space over the complex numbers. If  $\mathcal{F}$  is a compact subset of  $S$ , we denote by  $A = A^d(\mathcal{F}, S, M)$  the metric space of all functions  $f \in A^d(S)$  that satisfy  $|f(z)| \leq M$  for  $z \in S$ ;  $A$  is metrized by the supremum norm on  $\mathcal{F}$ .

### Proposition 2

Let  $d$  be a positive integer,  $M > 0$ ,  $S$  a bounded set in the  $d$ -dimensional complex space,  $\mathcal{F}$  its compact subset with nonempty interior. Then there exists an integer  $n_1$  such that in the supremum norm for all  $n \geq n_1$ ,

$$d_n(A^d(\mathcal{F}, S, M)) \geq \rho^{n^{1/d}}$$

where  $\rho$  is a constant independent of  $n$  and  $0 < \rho < 1$ .

*Proof*

By [30, Theorem 6, p. 161], there exist  $a, b > 0$  such that  $a \log_2^{d+1}(1/\varepsilon) \leq H_\varepsilon(A^d(\mathcal{F}, S, M)) \leq b \log_2^{d+1}(1/\varepsilon)$ . From basic facts about analytic functions it follows that all  $f \in A^d(\mathcal{F}, S, M)$  have uniformly bounded derivatives [30, p. 156] on  $\mathcal{F}$ . Therefore, by Ascoli-Arzelà’s theorem [20,

p. 113], the sets  $A^d(\mathcal{F}, S, M)$  are compact. So we conclude the statement by Proposition 1 (i) with  $G = A^d(\mathcal{F}, S; M)$  and  $\beta = d + 1$ .

We now turn our attention to the lower bounds on the  $n$ -width of balls in  $G_\phi$ -variation, for which we know from Corollary 1 that non-linear models of the  $\phi$ -network type achieve a modelling accuracy of order  $O(1/\sqrt{n})$ . We use the following reformulation of a result from [33, p. 79], also reported in a slightly simplified form in Reference [30, Theorem 8, p. 164], which gives a relationship between ‘inverse’  $n$ -width and covering numbers.

*Theorem 3* (Mityagin [33])

Let  $(X, \|\cdot\|)$  be a Banach space,  $K$  be its subset,  $s_K = \sup_{f \in K} \|f\|$ ,  $\varepsilon > 0$  and  $n_\varepsilon(K) = \sup\{n \in \mathcal{N}_+ : d_n(K) \geq \varepsilon\}$ . If  $n_\varepsilon$  is finite, then

$$\text{cov}_\varepsilon(K) \leq \left[ 8 \left( 1 + \frac{s_K}{\varepsilon} \right) \right]^{n_{\varepsilon/2}}. \tag{3}$$

Theorem 3 gives a relationship between the rate of increase in  $\text{cov}_\varepsilon(K)$  with  $\varepsilon$  and the rate of decrease in  $d_n(K)$  with  $n$ . Indeed, by (3), monotonicity of  $n$ -width [21, p. 10], and definition of  $n_\varepsilon$ , for every  $n \in \mathcal{N}_+$  we have

$$d_n(K) < \varepsilon \quad \text{implies} \quad n_\varepsilon(K) \geq \frac{\log_2 \text{cov}_{2\varepsilon}(K)}{3 + \log_2(1 + (s_K/2\varepsilon))} \tag{4}$$

Thus, if all elements of  $K$  have to be approximated within  $\varepsilon$  by an  $n$ -dimensional subspace, then  $n$  has to be at least proportional to the logarithm of the number of balls of radius  $2\varepsilon$  necessary to cover  $K$ . In the following, we use Theorem 3 together with estimates of covering numbers to derive a lower bound on the worst-case modelling error by linear methods of certain sets of  $d$ -variable functions.

In some cases of interest in applications (see Section 6), the balls in  $G$ -variation contain an orthogonal subset having ‘sufficiently many’ elements with norms larger than or equal to  $1/k$ , for each positive integer  $k$ . In Reference [17], the concept of a set *not quickly vanishing with respect to a positive integer  $d$* , was introduced and defined as a subset  $A$  of a normed linear space  $(X, \|\cdot\|)$  such that  $A = \bigcup_{k \in \mathcal{N}_+} A_k$ , where, for each  $k \in \mathcal{N}_+$ ,  $\text{card } A_k \geq k^d$  and for each  $h \in A_k$ ,  $\|h\| \geq 1/k$ . The following proposition gives a lower bound on the  $n$ -width  $d_n(B_1(\|\cdot\|_G))$ , which will be applied in Section 6 together with Corollary 1 to compare the performances of linear methods and  $\phi$ -network models of the perceptron type.  $\mathcal{H}(p) = -p \log_2(p) - (1-p) \log_2(1-p)$ , for  $0 < p < 1$ , denotes the binary entropy function, and  $\lfloor x \rfloor$  denotes the largest integer smaller than or equal to a real number  $x$ .

*Proposition 3*

Let  $(X, \|\cdot\|)$  be a Hilbert space,  $G$  be its subset containing an orthogonal set not quickly vanishing with respect to  $d$ , and  $\varepsilon > 0$ . Then there exists  $\varepsilon_0 > 0$  such that for  $0 \leq \varepsilon \leq \varepsilon_0$  and every positive integer  $n$ ,

$$d_n(B_1(\|\cdot\|_G)) < \varepsilon \quad \text{implies} \quad n \geq \frac{b(1/4\varepsilon)^{2d/(d+2)} - 1}{3 + \log_2(1 + (s_G/2\varepsilon))} \tag{5}$$

where  $s_G = \sup_{g \in G} \|g\|$  and  $b = 1 - \mathcal{H}(\frac{1}{4})$ .

*Proof*

As  $G$  contains an orthogonal set  $A$  not quickly vanishing with respect to  $d$ , for any integer  $k$  we have  $\text{cov}_\varepsilon(B_1(\|\cdot\|_G)) \geq \text{cov}_\varepsilon(B_1(\|\cdot\|_{A_k}))$ , where  $\text{card } A_k \geq k^d$ ,  $A = \bigcup_{k \in \mathcal{N}_+} A_k$ , and  $\min_{h \in A_k} \|h\| \geq 1/k$ . By [34, Lemma 3.7], for  $k \geq 3$  one has the following lower bound

$$\text{cov} \frac{1}{2k\sqrt{k^d}}(B_1(\|\cdot\|_{A_k})) \geq 2^{bk^d-1} \tag{6}$$

where  $b = \mathcal{H}(\frac{1}{4})$ . Taking  $k \in \mathcal{N}_+$  such that  $k^d = \lfloor (\frac{1}{4\varepsilon})^{2d/(d+2)} \rfloor$  and  $k \geq 3$  (which implies  $0 \leq \varepsilon \leq \varepsilon_0$  for a suitable  $\varepsilon_0$ ), from (6) we get

$$\text{cov}_{2\varepsilon}(B_1(\|\cdot\|_G)) \geq 2^{b\lfloor (1/4\varepsilon)^{2d/(d+2)} \rfloor - 1}$$

From the definition it follows  $s_G = s_{B_1(\|\cdot\|_G)}$ , so by Theorem 3 we conclude that  $d_n(B_1(\|\cdot\|_G)) < \varepsilon$  implies

$$\begin{aligned} n \geq n_\varepsilon(B_1(\|\cdot\|_G)) &\geq \frac{\log_2(\text{cov}_{2\varepsilon}(B_1(\|\cdot\|_G)))}{\log_2 8(1 + s_G/2\varepsilon)} \geq \frac{b\lfloor (1/4\varepsilon)^{2d/(d+2)} \rfloor - 1}{\log_2 8(1 + (s_G/2\varepsilon))} \\ &= \frac{b\lfloor (1/4\varepsilon)^{2d/(d+2)} \rfloor - 1}{3 + \log_2(1 + (s_G/2\varepsilon))}. \end{aligned}$$

The lower bound from Proposition 3 implies a ‘slow’ decrease with respect to  $n$  in the  $n$ -width of balls in  $G$ -variation, for  $G$  containing an orthogonal set not quickly vanishing with respect to  $d$ . Such a slow decrease implies a poor modelling accuracy by linear methods of functions from balls in  $G$ -variation.

## 6. APPLICATION TO PERCEPTRON-NETWORK MODELS

This section describes an application of Proposition 3 to a widespread class of  $\phi$ -network models, namely neural networks with perceptrons as hidden units. As recalled in Section 2, a perceptron with an activation function  $\psi : \mathcal{R} \rightarrow \mathcal{R}$  computes functions of the form  $\phi((v, b), x) = \psi(v \cdot x + b) : \mathcal{R}^{d+1} \times \mathcal{R}^d \rightarrow \mathcal{R}$ , where  $v \in \mathcal{R}^d$  is an *input weight* vector and  $b \in \mathcal{R}$  is a *bias*. Widely used activation functions are *sigmoidals*, i.e. bounded measurable functions  $\sigma : \mathcal{R} \rightarrow [0, 1]$  such that  $\lim_{t \rightarrow -\infty} \sigma(t) = 0$  and  $\lim_{t \rightarrow +\infty} \sigma(t) = 1$ . One can use both continuous sigmoidals (like the logistic sigmoid  $1/(1 + e^{-t})$  or the hyperbolic tangent) and the discontinuous Heaviside function  $\mathfrak{H}$ , defined as  $\mathfrak{H}(t) = 0$  for  $t < 0$  and  $\mathfrak{H}(t) = 1$  for  $t \geq 0$ .  $P_d(\sigma) = \{f : [0, 1]^d \rightarrow \mathcal{R} : f(x) = \sigma(v \cdot x + b), v \in \mathcal{R}^d, b \in \mathcal{R}\}$  denotes the set of functions on  $[0, 1]^d$  computable by  $\sigma$ -perceptrons,  $\|\cdot\|_{P_d(\sigma)}$  denotes variation with respect to sigmoidal perceptrons with  $d$  inputs, and  $\text{span } P_d(\sigma)$  is the set of functions on  $[0, 1]^d$  computable by sigmoidal perceptron-network models with any number of hidden units.

Next corollary provides a lower bound on the complexity of sigmoidal perceptron-network models.

*Corollary 2*

Let  $d$  and  $n$  be positive integers and  $\sigma : \mathcal{R} \rightarrow \mathcal{R}$  be an increasing sigmoidal function. Then in  $(L^2([0, 1]^d), \|\cdot\|_2)$ , for every positive integer  $n$  and every  $r > 0$

$$\delta(B_1(\|\cdot\|_{P_d(\sigma)}, \text{span}_n G)) \leq \frac{1}{\sqrt{n}} \quad (7)$$

*Proof*

It is easy to check that in  $(L^2([0, 1]^d), \|\cdot\|_2)$ , we have  $s_{P_d(\sigma)} = 1$ . Thus, the result follows from Corollary 1 applied to  $(X, \|\cdot\|) = (L^2([0, 1]^d), \|\cdot\|_2)$ .  $\square$

According to the bound from Corollary 2, to guarantee a modelling accuracy  $\varepsilon$  in  $(L_2([0, 1]^d), \|\cdot\|_2)$  the number of sigmoidal perceptron units has to grow *at most* as  $1/\varepsilon^2$ , independently of the number  $d$  of functions in the mappings to be modelled. On the other hand, Proposition 3 gives the following lower bound on the complexity of linear methods.

*Corollary 3*

Let  $d$  and  $n$  be positive integers and  $\sigma : \mathcal{R} \rightarrow \mathcal{R}$  be an increasing sigmoidal function. Then there exists  $\varepsilon_0 > 0$  such that for  $0 \leq \varepsilon \leq \varepsilon_0$  and every positive integer  $n$ , in  $(L^2([0, 1]^d), \|\cdot\|_2)$  we have

$$d_n(B_1(\|\cdot\|_{P_d(\sigma)})) < \varepsilon \quad \text{implies} \quad n > \frac{b(1/(4\varepsilon))^{2d/(d+2)} - 1}{3 + \log_2(1 + (1/2\varepsilon))} \quad (8)$$

where  $b = 1 - \mathcal{H}(\frac{1}{4})$ .

*Proof*

By Reference [34, Proposition 5.1], for every positive integers  $d$  and  $n$  and every sigmoidal function  $\sigma : \mathcal{R} \rightarrow \mathcal{R}$ , in  $(L_2([0, 1]^d), \|\cdot\|_2)$  the ball  $B_1(\|\cdot\|_{P_d(\sigma)})$  contains an orthogonal subset not quickly vanishing with respect to  $d$ . Hence Corollary 3 follows from Proposition 3 and from the fact that  $s_{P_d(\sigma)} = 1$ .

As discussed in Section 4, many functional classes that are important in applications are contained in balls  $B_r(\|\cdot\|_{P_d(\sigma)})$  (see also Reference [16]); in all such cases, one can apply the upper bound from Corollary 2 on the number of basis functions in sigmoidal perceptron network models and the lower bound from Corollary 3 on the number of fixed basis functions in linear methods.

## 7. CONCLUDING REMARKS

We have investigated the properties of a class of non-linear models including neural networks, in terms of approximation rates of non-linear schemes having the structure of linear combinations of parametrized basis functions, corresponding to a type of network computational unit. The mappings to be modelled have been represented as hypothesis classes defined by smoothness conditions (corresponding to different operating environments) in functional spaces dependent

on the specific modelling task to be accomplished. Exploiting tools from approximation theory, we have derived estimates of the complexity of linear and neural-network models; such estimates are expressed as bounds on the number of adjustable parameters necessary for a given modelling accuracy.

We have discussed classes of non-linear models with desirable computational capabilities and bounded complexity, guaranteeing that the number of adjustable parameters does not increase too fast (e.g. only polynomially) with the dimensionalities of certain modelling tasks and is often much smaller than the number of parameters in linear methods. Our theoretical analysis supports the intuition that the outperformances of non-linear methods, such as neural-networks, as compared with linear ones, are justified by a ‘smarter’ way of using adjustable parameters and of exploiting smoothness hypotheses on the sets of multivariable mappings to be modelled.

Besides their powerful approximation capabilities, another feature making neural-network models particularly attractive is their intrinsic parallelism. Although until now such networks have mostly been simulated on classical serial computers, the availability of parallel VLSI realizations will make such models still more interesting in complex tasks.

From an experimental point of view, the use of non-linear models such as neural networks is enforced by their efficiency in complex adaptive tasks; from a theoretical perspective, their use is made particularly interesting by the possibility of achieving a desired modelling accuracy by means of much fewer parameters than in the case of linear methods. Thus, experience is supplemented with the design criteria based on a mathematical analysis: theoretical results explain why neural-network models should be preferred, and allow a deeper insight into the choice of a network architecture for a given task.

#### ACKNOWLEDGEMENTS

The authors are grateful to Prof. Věra Kůrková (Academy of Sciences of the Czech Republic) for valuable discussions and remarks.

#### REFERENCES

1. Kanellakopoulos I, Kokotovic PV, Morse AS. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transactions on Automatic Control* 1991; **11**:1241–1253.
2. Sastry SS, Isidori A. Adaptive control of linearizable systems. *IEEE Transactions on Automatic Control* 1989; **34**:1123–1131.
3. Chen FC, Liu CC. Adaptively controlling non-linear continuous-time systems using multilayer neural networks. *IEEE Transactions on Automatic Control* 1994; **39**:1306–1310.
4. Chen FC, Khalil H. Adaptive control of a class of non-linear discrete-time systems using multilayer neural networks. *IEEE Transactions on Automatic Control* 1995; **40**:791–801.
5. Narendra KS, Parthasarathi K. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks* 1990; **1**:4–26.
6. Narendra KS, Mukhopadhyay S. Adaptive control using neural networks and approximate models. *IEEE Transactions on Neural Networks* 1997; **8**:475–485.
7. Narendra KS, Balakrishnan J, Ciliz KM. Adaptation and learning using multiple models, switching, and tuning. *IEEE Control Systems Magazine* 1995; **15**:37–51.
8. Pinkus A. Approximation theory of the MLP model in neural networks. *Acta Numerica* 1999; **8**:143–196.
9. Kůrková V. Universality and complexity of approximation of multivariable functions by feedforward networks. In *Softcomputing and Industry: Recent Applications*, Roy R, Koepfen M, Ovaska S, Furuhashi T, Hoffmann F (eds). Springer-Verlag: London, 2002; 13–24.
10. Bellman R. *Dynamic Programming*. Princeton University Press: Princeton, NJ, 1957.

11. Burr DJ. Experiments on neural net recognition of spoken and written text. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 1988; **36**:1162–1168.
12. Parisini T, Zoppoli R. Neural networks for feedback feedforward non-linear control systems. *IEEE Transactions on Neural Networks* 1994; **5**:436–449.
13. Sejnowski TJ, Rosenberg C. Parallel networks that learn to pronounce English text. *Complex Systems* 1987; **1**:145–168.
14. Zoppoli R, Parisini T. Learning techniques and neural networks for the solution of N-stage non-linear nonquadratic optimal control problems. In *Systems, Models and Feedback: Theory and Applications*, Isidori A, Tarn TJ (eds). Birkhäuser: Basel, 1992; 193–210.
15. Zoppoli R, Sanguineti M, Parisini T. Approximating networks and extended Ritz method for the solution of functional optimization problems. *Journal of Optimization Theory and Applications* 2002; **112**:403–440.
16. Barron AR. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory* 1993; **39**:930–945.
17. Kůrková V, Sanguineti M. Comparison of worst case errors in linear and neural network approximation. *IEEE Transactions on Information Theory* 2002; **48**:264–275.
18. Sjöberg J, Zhang Q, Ljung L, Benveniste A, Delyon B, Glorennec P-Y, Hjalmarsson H, Juditsky A. Nonlinear black-box models in system identification: a unified overview. *Automatica* 1995; **31**:1691–1724.
19. Juditsky A, Hjalmarsson H, Benveniste A, Delyon B, Ljung L, Sjöberg J, Zhang Q. Nonlinear black-box models in system identification: mathematical foundations. *Automatica* 1995; **31**:1725–1750.
20. Friedman A. *Foundations of Modern Analysis*. Dover: New York, 1982.
21. Pinkus A. *N – Widths in Approximation Theory*. Springer: New York, 1986.
22. DeVore R, Howard R, Micchelli C. Optimal non-linear approximation. *Manuscripta Mathematica* 1989; **63**:469–478.
23. Kainen PC, Kůrková V, Vogt A. Approximation by neural networks is not continuous. *Neurocomputing* 1999; **29**:47–56.
24. Pisier G. Remarques sur un resultat non publié de B. Maurey. *Seminaire d'Analyse Fonctionnelle* 1980–81; **1**(12) École Polytechnique, Centre de Mathématiques, Palaiseau.
25. Jones LK. A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression and neural network training. *Annals of Statistics* 1992; **20**:608–613.
26. Kůrková V. Dimension-independent rates of approximation by neural networks. In *Computer-Intensive Methods in Control and Signal Processing. The Curse of Dimensionality*, Warwick K, Kárný M (eds). Birkhäuser: Basel, 1997; 261–270.
27. Kůrková V, Savický P, Hlaváčková K. Representations and rates of approximation of real-valued Boolean functions by neural networks. *Neural Networks* 1998; **11**:651–659.
28. Breiman L. Hinging hyperplanes for regression, classification, and function approximation. *IEEE Transactions on Information Theory* 1993; **39**:993–1013.
29. Giulini S, Sanguineti M. On dimension-independent approximation by neural networks and linear approximators. *Proceedings of the International Joint Conference on Neural Networks*, Como, Italy, 2000; **1**:283–288.
30. Lorentz GG. *Approximation of Functions* (2nd edn). Chelsea Publ. Company: New York, 1986.
31. Timan AF. *Theory of Approximation of Functions of a Real Variable*. Dover Publications Inc.: New York, 1993.
32. Wahlberg B. Laguerre and Kautz models. *Proceedings of IFAC Symposium on System Identification*; Copenhagen, Denmark, 1994; 965–976.
33. Mityagin BS. The approximate dimension and bases in nuclear spaces. *Russian Mathematical Surveys* 1961; **16**:59–127 (Original Russian in *Uspekhi Matematicheskikh Nauk*. 1961, Tom XVI, 4:63–132).
34. Kůrková V, Sanguineti M. Tight bounds on rates of variable-basis approximation via estimates of covering numbers. *Research Report ICS-02-865 - Institute of Computer Science - Academy of Sciences of the Czech Republic*, 2002.
35. Kůrková V, Sanguineti M. Bounds on rates of variable-basis and neural-network approximation. *IEEE Transactions on Information Theory* 2001; **47**:2659–2665.