

# Capacity of Structured Multilayer Networks with Shared Weights

Sabine Kröner<sup>1</sup> and Reinhard Moratz<sup>2</sup>

<sup>1</sup> Technische Informatik I, TU Hamburg-Harburg, D-21071 Hamburg

<sup>2</sup> AG Angewandte Informatik, Universität Bielefeld, Postfach 100131,  
D-33501 Bielefeld, Germany

**Abstract.** The capacity or Vapnik-Chervonenkis dimension of a feedforward neural architecture is the maximum number of input patterns that can be mapped correctly to fixed arbitrary outputs. So far it is known that the upper bound for the capacity of two-layer feedforward architectures with independent weights depends on the number of connections in the neural architecture [1].

In this paper we focus on the capacity of multilayer feedforward networks with shared weights, also known as structured architectures. We show that structured architectures can be transformed into equivalent conventional multilayer feedforward architectures. Known estimations for the capacity are extended to achieve upper bounds for the capacity of these general multi-layer feedforward architectures. As a result an upper bound for the capacity of structured architectures is derived that increases with the number of independent network parameters. This means that weight sharing in a fixed neural architecture leads to a significant reduction of the upper bound of the capacity.

## 1 Introduction

Structured multi-layer feedforward networks gain more and more importance in speech- and image processing applications. Their characteristic is that a-priori knowledge about the task to be performed is already built into their architecture by use of nodes with shared weight vectors. Examples are time delay neural networks [10] and networks for invariant pattern recognition [4, 5].

One problem in the training of neural networks is the estimation of the number of training samples needed to achieve good generalization. In [1] is shown that for feedforward neural architectures this number is correlated with the capacity or Vapnik-Chervonenkis dimension of the architecture. So far an upper bound for the capacity has been derived for two-layer feedforward neural architectures with independent weights: it depends with  $\mathcal{O}(\frac{w}{a} \cdot \ln \frac{q}{a})$  on the number  $w$  of connections in the neural architecture with  $q$  nodes and  $a$  output elements.

In this paper we focus on the calculation of upper bounds for the capacity of structured multi-layer feedforward neural architectures. First we give some definitions and introduce a new general terminology for the description of structured neural networks. In section 3 we apply this terminology on structured feedforward architectures first with one layer then with multiple layers. We show that

they can be transformed into equivalent conventional multi-layer feedforward architectures, and derive upper bounds for the capacity of the structured architectures. At the end we comment the results.

## 2 Definitions

A *layered feedforward network architecture*  $\mathcal{N}_{e,a}^r$  is a directed acyclic graph with a sequence of  $e$  input nodes,  $r-1$  ( $r \in \mathbb{N}$ ) intermediate (*hidden*) layers of nodes, and a final output layer with  $a$  nodes. Every node is connected only to nodes in the next layer.

To every node  $k$  with indegree  $n \in \mathbb{N}$  a triplet  $(\mathbf{w}_k, s_k, f_k)$  is assigned, consisting of a weight vector  $\mathbf{w}_k \in \mathbb{R}^n$ , a threshold value  $s_k \in \mathbb{R}$ , and an activation function  $f_k : \mathbb{R} \rightarrow \{0, 1\}$ . The activation  $y_k$  of the node for an input vector  $\mathbf{x} \in \mathbb{R}^n$  is computed in the common way:  $y_k = f_k(\mathbf{w}_k \cdot \mathbf{x})$ . For all input nodes (indegree 1) the weight vectors and the activation functions are fixed:  $\mathbf{w} := (1)$ ,  $f := \text{Id}$ . The activation function for all other nodes is the hard limiter function, and without loss of generality we choose  $s = 0$  for the threshold values of all nodes. An architecture  $\mathcal{N}_{e,a}^r$  with given triplets  $(\mathbf{w}, s, f)$  for all nodes we define as a *net*  $N_{e,a}^r$ . With the net itself a function  $F : \mathbb{R}^e \mapsto \{0, 1\}^a$  is associated.

Let an architecture  $\mathcal{N}_{e,a}^r$  be given. A set of  $m \in \mathbb{N}$  input vectors  $\mathbf{x}_l \in \mathbb{R}^e$  ( $l = 1, \dots, m$ ) arranged as  $m$  rows in a  $(m \times e)$ -matrix  $S$  is denoted *input matrix* for  $\mathcal{N}_{e,a}^r$ . An input-matrix  $S$  is mapped to an  $(m \times a)$ -*output-matrix*  $T$  by a net  $N_{e,a}^r$ .

Let  $S$  be a fixed input-matrix  $S$  for  $\mathcal{N}_{e,a}^r$ . All nets  $N_{e,a}^r$  that map  $S$  on the same output-matrix  $T$  are grouped in a *net class* of  $\mathcal{N}_{e,a}^r$  related to  $S$ .  $\Delta(S)$  is the number of net classes of  $\mathcal{N}_{e,a}^r$  related to  $S$ . The *growth function*  $g(m)$  of an architecture  $\mathcal{N}_{e,a}^r$  with  $m$  input vectors is the maximum number of net classes over all  $(m \times e)$ -input matrices  $S$ .

Now we consider the nodes of the architecture  $\mathcal{N}_{e,a}^r$  within one layer (except the input layer) with the same indegree  $d \in \mathbb{N}$ . All nodes  $k$  whose components of their weight vectors  $\mathbf{w}_k \in \mathbb{R}^d$  can be permuted through a permutation  $\pi_k : \mathbb{R}^d \rightarrow \mathbb{R}^d$  so that  $\pi_k(\mathbf{w}_k) = \mathbf{w} \forall k$  for some vector  $\mathbf{w} \in \mathbb{R}^d$ , are elements of the same *node class*  $K\mathbf{w}$ . We call an architecture  $\mathcal{N}_{e,a}^r$  *structured* if at least one node class has more than one element. Then the architecture with  $b$  node classes  $K\mathbf{w}_i$  ( $i = 1, \dots, b$ ) is denoted  $\mathcal{N}_{e,a}^r(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$ .

The *Vapnik-Chervonenkis dimension*  $d_{VC}$  [9] of a feedforward architecture is defined by  $d_{VC} := \sup \{m \in \mathbb{N} \mid g(m) = 2^{ma}\}$ . Let  $Q := \left\{ m \in \mathbb{N} \mid \frac{g(m)}{2^{ma}} \geq \frac{1}{2} \right\}$ .

Then  $c := \sup Q$  for  $Q \neq \emptyset$ , or  $c := 0$  for  $Q = \emptyset$ , is an upper bound for the Vapnik-Chervonenkis dimension and also defined as capacity in [2, 7].

## 3 Upper bounds for the capacity

In this section is shown how structured architectures can be transformed into conventional architectures with independent weights. The upper bounds for the

capacity of these conventional architectures then are applied to the structured architectures.

A basic transformation needed in the following derivations is the transformation of structured one-layer architectures  $\mathcal{N}(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with input nodes of outdegree  $\geq 1$  and input vectors  $\mathbf{x}_l$  into structured one-layer architectures  $\mathcal{N}'(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with input nodes of outdegree 1 only and dependent input vectors  $\mathbf{x}_l'$  ( $l = 1, \dots, m$ ): Every input node with outdegree  $z > 1$  is replaced by  $z$  copies of that input node. The outgoing edges of the input node are assigned to the copies in such a way that every copy has outdegree 1. The elements of the input vectors are duplicated in the same way. By permuting the input nodes and the corresponding components of the input vectors we get the architecture  $\mathcal{N}''(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  without any intersecting edges.

### 3.1 Structured one-layer architectures

I) First we focus on structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with a set  $I := \{u_1, \dots, u_e\}$  of  $e$  input nodes and the output layer  $K := \{k_1, \dots, k_a\}$ . Let  $K\mathbf{w} := K$  be the only node class. All nodes in  $K\mathbf{w} = K$  have the same indegree  $d \in \mathbb{N}$ .

**Theorem 1.** *Let a structured one-layer architecture  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with only one node class  $K\mathbf{w} = K$  be given. Suppose  $d \in \mathbb{N}$  as the indegree of all nodes in  $K\mathbf{w}$ . The number of input nodes is  $e \leq a \cdot d$ . For  $m$  input vectors of length  $e$*

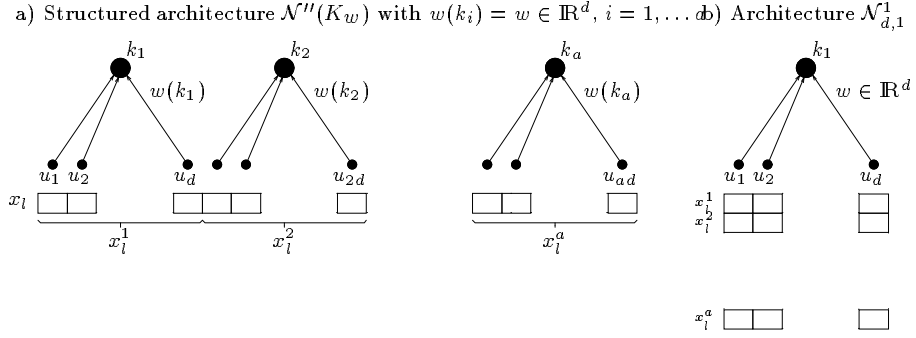
$$C(m \cdot a, d) := 2 \cdot \sum_{i=0}^{d-1} \binom{m \cdot a - 1}{i}$$

*is an upper bound for the growth function  $g(m)$  of the structured one-layer architecture  $\mathcal{N}_{e,a}^1(K\mathbf{w})$ .*

*Proof.* At first we examine structured one-layer architectures with outdegree 1 for every input node, equivalent to architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with  $e = a \cdot d$  input nodes. By permuting the input nodes and the corresponding components of the input vectors we get the architecture  $\mathcal{N}''(K\mathbf{w})$ . Without loss of generality we consider the permutation  $\pi$  of the node class  $K\mathbf{w}$  as the identity function. Thus we have  $\mathbf{w} = \mathbf{w}(k_1) = \dots = \mathbf{w}(k_a) \in \mathbb{R}^d$  for the  $a$  weight vectors (cf. Figure 1 a)). For  $m$  fixed input vectors  $\mathbf{x}_l := (\mathbf{x}_l^1, \dots, \mathbf{x}_l^a) \in \mathbb{R}^{ad}$  ( $\mathbf{x}_l^i \in \mathbb{R}^d$ ,  $l = 1, \dots, m$ ,  $i = 1, \dots, a$ ) let  $S$  be an  $(m \times a \cdot d)$ -input matrix for  $\mathcal{N}_{e,a}^1(K\mathbf{w})$ :

$$S := \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_m \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1^1 & \dots & \mathbf{x}_1^a \\ \vdots & \ddots & \vdots \\ \mathbf{x}_m^1 & \dots & \mathbf{x}_m^a \end{pmatrix}.$$

A given weight vector  $\mathbf{w}_1 \in \mathbb{R}^d$  defines a function  $F_1 : \mathbb{R}^{ad} \rightarrow \{0, 1\}^a$  or a net  $N_1$  respectively. Let  $\mathbf{w}_2$  be a weight vector that defines a function  $F_2$  (a net  $N_2$ ) different to  $F_1$  on the input matrix  $S$ . Thus these two nets are elements of different net classes of  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  related to the input matrix  $S$ .



**Fig. 1.** a) Structured architecture  $\mathcal{N}(K_w)''$  with an input vector  $x_l$  ( $l \in \{1, \dots, m\}$ ).  
b) Architecture  $\mathcal{N}_{d,1}^1$  with the corresponding  $a$  input vectors  $x_l^1, \dots, x_l^a$ .

Now we consider the one-layer architecture  $\mathcal{N}_{d,1}^1$ , consisting of a single node with indegree  $d$ . By rearranging the  $m$  rows of the input matrix  $S$  for  $\mathcal{N}''(K_w)$  to one column of the  $m \cdot a$  input vectors  $x_l^i \in \mathbb{R}^d$  ( $l = 1, \dots, m$ ,  $i = 1, \dots, a$ ) we derive the  $(m \cdot a \times d)$ -input matrix  $\tilde{S}$  for  $\mathcal{N}_{d,1}^1$  (cf. Figure 1 b)):

$$\tilde{S} := \begin{pmatrix} x_1^1 \\ x_1^2 \\ \vdots \\ x_m^a \end{pmatrix}. \quad (1)$$

On  $\mathcal{N}_{d,1}^1$  the weight vector  $w_1$  ( $w_2$ ) defines a function  $\tilde{F}_1 : \mathbb{R}^d \rightarrow \{0, 1\}$  ( $\tilde{F}_2 : \mathbb{R}^d \rightarrow \{0, 1\}$ ) or a net  $\tilde{N}_1$  ( $\tilde{N}_2$ ) respectively. Because of  $F_1(x_s) \neq F_2(x_s)$  for at least one input vector  $x_s$  ( $s \in \{1, \dots, m\}$ ) and definition (1) the nets  $\tilde{N}_1$  and  $\tilde{N}_2$  are elements of different net classes of  $\mathcal{N}_{d,1}^1$  related to the input matrix  $\tilde{S}$ .

Summarizing we get: if two nets of the architecture  $\mathcal{N}_{e,a}^1(K_w)$  are different related to any input matrix  $S$  we can define an input matrix  $\tilde{S}$  for  $\mathcal{N}_{d,1}^1$  by (1), so that the corresponding nets are different, too. For the number of net classes this yields

$$\Delta(S) \leq \Delta(\tilde{S}). \quad (2)$$

With the results of [7] the growth function of the architecture  $\mathcal{N}_{d,1}^1$  is given by  $C(m \cdot a, d)$ . From (2) also follows that this is an upper bound for the growth function of the structured one-layer architecture  $\mathcal{N}''(K_w)$  or  $\mathcal{N}_{ad,a}^1(K_w)$  respectively:  $g(m) \leq C(m \cdot a, d)$ . The inequation  $g(m) \geq C(m \cdot a, d)$  can easily be verified in a similar way, so it yields  $g(m) = C(m \cdot a, d)$  for the growth function of structured one-layer architectures  $\mathcal{N}_{e,a}^1(K_w)$  with outdegree 1 for every input node.

Now we consider structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with outdegree  $z > 1$  for some input nodes. These architectures can be transformed into structured one-layer architectures  $\mathcal{N}''(K\mathbf{w})$  with  $e = a \cdot d$  input nodes all with outdegree 1. But the input vectors of the input matrix for the transformed architecture  $\mathcal{N}''(K\mathbf{w})$  cannot be chosen totally independent. Thus  $C(m \cdot a, d)$  is an upper bound for the growth function of structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with exactly one node class  $K\mathbf{w} = K$ .  $\square$

*Remark.* With [7] we find  $\frac{2 \cdot d}{a}$  as an upper bound for the capacity of structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w})$  with exactly one node class  $K\mathbf{w}$ .

**II)** Second we focus on structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with  $b$  ( $2 \leq b < a$ ) node classes  $K\mathbf{w}_1, \dots, K\mathbf{w}_b$ . These classes form the set  $K$  of the  $a$  output nodes:  $K = K\mathbf{w}_1 \dot{\cup} \dots \dot{\cup} K\mathbf{w}_b$ .

**Theorem 2.** *Assume a structured one-layer architecture  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with  $e \leq \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes,  $a = \sum_{i=1}^b \alpha_i$  output nodes, and  $b \in \mathbb{N}$  ( $2 \leq b \leq a$ ) node classes  $K\mathbf{w}_i$  ( $i = 1, \dots, b$ ). Let  $\alpha_i := |K\mathbf{w}_i|$  be the sizes of the node classes  $K\mathbf{w}_i$ , and  $d_i$  the indegrees of the nodes in  $K\mathbf{w}_i$  ( $i = 1, \dots, b$ ). For  $m$  input vectors the product*

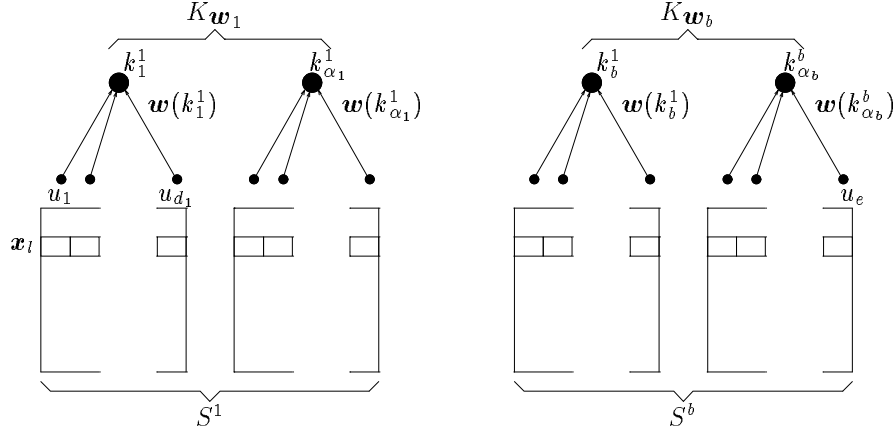
$$\prod_{i=1}^b C(m \cdot \alpha_i, d_i)$$

*is an upper bound for the growth function of  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$ .*

*Proof.* First we examine structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with  $e = \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes (all with outdegree 1) and an  $(m \times e)$ -input matrix  $S$ . The  $a$  nodes in the output layer  $K$  are permuted so that we get the ordered sequence  $K = \{K\mathbf{w}_1, \dots, K\mathbf{w}_b\}$  with  $K\mathbf{w}_i := \{k_1^i, \dots, k_{\alpha_i}^i\}$  ( $i = 1, \dots, b$ ). The input nodes and the corresponding components of the input vectors are permuted as in the Proof of Theorem 1: so no edges are intersecting and  $\mathbf{w}(k_1^i) = \mathbf{w}(k_2^i) = \dots = \mathbf{w}(k_{\alpha_i}^i)$  for  $i = 1, \dots, b$ . These permutations generate  $b$  structured one-layer sub architectures  $\mathcal{N}_{e_i, \alpha_i}^1(K\mathbf{w}_i)$  with  $e_i := \alpha_i \cdot d_i$  input nodes,  $\alpha_i$  output nodes and  $(m \times e_i)$ -sub input matrices  $S^i$  ( $i = 1, \dots, b$ ) (cf. Figure 2).

With Theorem 1 we get  $g_i(m) \leq C(m \cdot \alpha_i, d_i)$  for the growth functions  $g_i(m)$  of the sub architectures  $\mathcal{N}_{e_i, \alpha_i}^1(K\mathbf{w}_i)$ . For the determination of the growth function  $g(m)$  of  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  the  $b$  input matrices  $S^i$  for the sub architectures can be chosen independently. Thus we get  $g(m) = \prod_{i=1}^b g_i(m) \leq \prod_{i=1}^b C(m \cdot \alpha_i, d_i)$ .

Now we examine structured one-layer architectures  $\mathcal{N}_{e,a}^1(K\mathbf{w}_1, \dots, K\mathbf{w}_b)$  with outdegree  $\geq 1$  for some input nodes, equivalent to  $e < \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes. These architectures are transformed into structured one-layer architectures with  $e = \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes, each with outdegree 1 (cf. proof of Theorem 1) which then are transformed as in the beginning of this proof. As we have



**Fig. 2.**  $b$  sub architectures  $\mathcal{N}_{e_i, \alpha_i}^1(Kw_i)$  and the corresponding sub input matrices  $S^i$  ( $i = 1, \dots, b$ ) of a structured one-layer architecture  $\mathcal{N}_{e, a}^1(Kw_1, \dots, Kw_b)$  with  $b$  node classes  $Kw_i$  and  $e = \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes.

seen in the proof of Theorem 1 there are dependences between some elements of the input vectors since some components of the input vectors are identical. So  $\prod_{i=1}^b C(m \cdot \alpha_i, d_i)$  is an upper bound for the growth function of structured one-layer architectures  $\mathcal{N}_{e, a}^1(Kw_1, \dots, Kw_b)$  with  $e < \sum_{i=1}^b \alpha_i \cdot d_i$  input nodes, too.  $\square$

**Theorem 3.** *Given a structured one-layer architecture  $\mathcal{N}_{e, a}^1(Kw_1, \dots, Kw_b)$  with  $b \in \mathbb{N}$  ( $2 \leq b \leq a$ ) node classes  $Kw_i$  ( $i = 1, \dots, b$ ), maximum indegree  $\hat{d} \geq 2$  for all nodes, maximum size  $\hat{\alpha} := |Kw_i|$  of the node classes, and  $t := \frac{\hat{\alpha} \cdot b}{a} \geq 2$ , then for the capacity we get*

$$c = \mathcal{O} \left( \frac{b \cdot \hat{d}}{a} \cdot \ln t \right).$$

*Proof Sketch.* With the above definitions and Theorem 2 we get for the growth function  $g(m)$  of the architecture  $\mathcal{N}_{e, a}^1(Kw_1, \dots, Kw_b)$ :

$$g(m) \leq \prod_{i=1}^b C(m \cdot \alpha_i, d_i) \leq \prod_{i=1}^b C(m \cdot \hat{\alpha}, \hat{d}) = C(m \cdot \hat{\alpha}, \hat{d})^b.$$

This yields an upper bound for the capacity:  $c \leq \sup \left\{ m \in \mathbb{N} \mid \frac{C(m \cdot \hat{\alpha}, \hat{d})^b}{2^{m \cdot a}} \geq \frac{1}{2} \right\}$ .

With some estimations and  $const := \frac{2 + 2 \cdot \ln(2)}{(\ln(2))^2}$  it follows:

$$m \leq const \cdot \frac{t \cdot \hat{d}}{\hat{\alpha}} \cdot \ln(t).$$

For details and further information see [8].  $\square$

### 3.2 Structured multi-layer architectures

Consider a structured  $r$ -layer architecture with  $e$  input nodes,  $a_j$  nodes in the hidden layers  $H^j$  ( $j = 1, \dots, r-1$ ) and  $a$  nodes in the output layer  $K$ . Let the layers  $H^j$  be the disjoint union of the  $b_j \leq a_j$  node classes  $H_{\mathbf{w}_1^j}, \dots, H_{\mathbf{w}_{b_j}^j}$  and the output layer the disjoint union of the node classes  $K_{\mathbf{w}_i}$  ( $i = 1, \dots, b$ ). The number of node classes is  $\sum_{j=1}^{r-1} b_j + b =: \beta$ . The structured architecture is denoted by  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$ .

A structured  $r$ -layer feedforward architecture  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$  can be regarded as a combination of  $r$  structured 1-layer feedforward architectures since the output matrices of each layer are the input matrices for the following layer. Thus we get an upper bound for the growth function  $g(m)$  of  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$  by multiplying the growth functions of the  $r$  structured 1-layer architectures (refer to Theorem 2):

$$g(m) \leq \prod_{j=1}^{r-1} \left( \prod_{i=1}^{b_j} C(m \cdot \alpha_i^j, d_i^j) \right) \cdot \prod_{i=1}^b C(m \cdot \alpha_i, d_i). \quad (3)$$

With the maximum size  $\hat{\alpha} := \max \{ \alpha_1, \dots, \alpha_b, \alpha_1^1, \dots, \alpha_{b_{r-1}}^{r-1} \}$  of the  $\beta$  node classes, and the maximum indegree  $\hat{d}$  of all nodes of the architecture  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$ ,  $C(m \cdot \hat{\alpha}, \hat{d})^\beta$  is an upper bound for (3).

**Theorem 4.** *Let  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$  be a structured  $r$ -layer feedforward architecture with  $\beta \geq 2$  node classes  $H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b}$ ,  $\hat{d} \geq 2$  the maximum indegree of all nodes,  $\hat{\alpha}$  the maximum size of all  $\beta$  node classes, and  $\hat{t} := \frac{\hat{\alpha} \cdot \beta}{a} \geq 2$ . For the capacity of  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$  we get*

$$c = \mathcal{O} \left( \frac{\beta \cdot \hat{d}}{a} \cdot \ln \hat{t} \right).$$

*Proof.* Analogous to the proof of Theorem 3. □

An architecture  $\mathcal{N}_{e,a}^r(H_{\mathbf{w}_1^1}, \dots, K_{\mathbf{w}_b})$  with  $\sum_{j=1}^{r-1} b_j + b = \sum_{j=1}^{r-1} a_j + a$  node classes is equivalent to an architecture  $\mathcal{N}_{e,a}^r$  in which every node class has size 1. Thus the above upper bounds for the capacity hold good for conventional  $r$ -layer feedforward architectures with  $e$  input and  $a$  output nodes, too.

## 4 Conclusion

In this paper we determined upper bounds for the capacity of structured multi-layer feedforward neural architectures. By transforming architectures with shared weight vectors into equivalent conventional feedforward architectures and the extension of the definitions of the growth function and the capacity to multi-layer

feedforward architectures we could give estimations for the upper bounds of the capacity of structured multi-layer architectures. These upper bounds depend with  $O(\frac{p}{a} \cdot \ln \hat{t})$  on the number  $p$  of free parameters in the structured neural architecture with maximum size  $\hat{a}$  of the  $\beta$  node classes,  $\hat{t} := \frac{\hat{a} \cdot \beta}{a} \geq 2$ , and  $a$  nodes in the output layer. So weight sharing in a fixed neural architecture leads to a reduction of the upper bound of the capacity. The amount of the reduction increases with the extent of the weight sharing. With  $\hat{a} = 1$  the upper bounds hold good for conventional feedforward networks with independent weights, too.

It is known that the generalization ability of a feedforward neural architecture improves within certain limits with a reduction of the capacity for a fixed number of training samples. As a consequence of our results a better generalization ability can be derived for structured neural architectures compared to the same unstructured ones. This is a theoretic justification for the generalization ability of structured neural architectures observed in experiments [5].

Further investigations will focus on an improvement of the upper bounds for the capacity, on the determination of capacity bounds for special structured architectures, and on the derivation of capacity bounds for structured architectures of nodes with continuous transfer functions [3, 6].

## References

1. E. B. Baum, D. Haussler: *What Size Net gives Valid Generalization?*, Advances in Neural Information Processing Systems, D. Touretzky, Ed., Morgan Kaufmann, (1989).
2. T. M. Cover: *Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition*, IEEE Transactions on Electronic Computers, Vol. 14, 326-334, (1965).
3. P. Koiran, E.D. Sontag: *Neural Networks with Quadratic VC Dimension*, Neuro-COLT Technical Report Series, NC-TR-95-044, London, (1995)
4. S. Kröner, R. Moratz, H. Burkhardt: *An adaptive invariant transform using neural network techniques*, in Proceedings of EUSIPCO 94, 7th European Signal Processing Conf., Holt et al. (Ed.), Vol. III, 1489-1491, Edinburgh, (1994).
5. Y. le Cun: *Generalization and Network Design Strategies*, Connectionism in Perspective, R. Pfeiffer, Z. Schreter, F. Fogelman-Soulié, L. Steels (Eds.), Elsevier Science Publishers B.V. 143-155, North-Holland, (1989).
6. W. Maass: *Vapnik-Chervonenkis Dimension of Neural Nets*, Preprint, Technische Universität Graz, (1994).
7. G. J. Mitchison, R. M. Durbin: *Bounds on the Learning Capacity of Some Multi-Layer Networks*, Biological Cybernetics, Vol.60, No. 5, 345-356, (1989).
8. P. Rieper: *Zur Speicherfähigkeit vorwärtsgerichteter Architekturen künstlicher neuronaler Netze mit gekoppelten Knoten*, Diplomarbeit, Universität Hamburg, (1994).
9. V. Vapnik: *Estimation of Dependences Based on Empirical Data*, Springer-Verlag, Berlin, (1982).
10. A. Waibel: *Modular Construction of Time-Delay Neural Networks for Speech Recognition*, Neural Computation, Vol.1, 39-46, (1989).