An Analytical Framework for Local Feedforward Networks*

Scott Weaver[1,2], **Leemon Baird**[3], **Marios Polycarpou**[1]

[1]Department of Electrical and Computer Engineering
University of Cincinnati
Cincinnati, Ohio 45221-0030
Email: scott.weaver@uc.edu

[2]Wright-Patterson Air Force Base
WL/AACF
2241 Avionics Circle
WPAFB, Ohio 45433-7318

[3]Computer Science Department
5000 Forbes Avenue
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213-3891

IEEE Transactions on Neural Networks
Submitted: September 1996; Revised: June 1997.

Abstract

Interference in neural networks occurs when learning in one area of the input space causes unlearning in another area. Networks that are less susceptible to interference are referred to as spatially *local* networks. To obtain a better understanding of these properties, a theoretical framework, consisting of a measure of interference and a measure of network localization, is developed. These measures incorporate not only the network weights and architecture but also the learning algorithm. Using this framework to analyze sigmoidal, multi-layer perceptron (MLP) networks that employ the back-propagation learning algorithm on the quadratic cost function, we address a familiar misconception that single-hidden-layer, sigmoidal networks are inherently non-local by demonstrating that given a sufficiently large number of adjustable weights, single-hidden-layer, sigmoidal MLPs exist that are arbitrarily local and retain the ability to approximate any continuous function on a compact domain.

*Partially supported under Task 2312 R1 by the United States Air Force Office of Scientific Research.

1 Introduction

Although the concept of local learning has become familiar in the neural network literature, there is no consensus on the description of local learning. The only common element found in characterizations of local learning is the general understanding that there are links or associations between regions of the input space and sets of adjustable parameters. Typically, a description of a local learning system is simply based on the characteristics of the particular network structure, rather than some fundamental definition of localization [1, 2, 3, 4]. The literature does not provide a universally accepted description of local learning systems nor does it provide any method for measuring the localization properties of a learning system.

Although no rigorous description of local learning is available, the ability of a local learning system to alleviate interference problems during learning is well accepted. In the case of incremental supervised learning, where weights are updated after each presentation of a training sample, *interference* occurs when training at one point of the input space affects the input/output (I/O) map in an undesirable way in other areas of the input space. The general problem of interference has been uncovered in various forms by researchers in many areas [5, 1]. For example, consider a dynamical system after it settles into a desired trajectory (where only a small portion of the input space is reached). Suppose that without noise, a network function approximator learns the system dynamics, reduces the approximation error, and then ceases learning. In the presence of noise, however, the learning algorithm remains active and continually memorizes the system dynamics along the trajectory (because the error never goes to zero) even though there is no need to do so. "Global Network Collapse" results, as the other areas of the input space (those areas not on the trajectory) gradually unlearn due to interference [1]. Another variant of the interference problem is in the classification literature: "Catastrophic Interference" occurs when the training of a new pattern causes the unlearning of originally trained patterns [5]. These and other interference problems may appear different when embedded in their particular applications but the root of these problems is the same; learning tends to interfere with previous learning elsewhere in the input space.

Although local networks, in general, lessen the problem of interference, there are trade-offs to consider (see Barto [6] for a nice summary). For example, look-up tables can be thought of as the most

local of approximation structures because there is a one-to-one relationship between a point in the input space and an adjustable parameter. However, look-up tables are obviously inappropriate when the dimension of the problem grows large because the *curse of dimensionality* [7] causes memory requirements to become prohibitive; furthermore, look-up tables provide no generalization of untrained points. Finding the correct balance between avoiding interference problems, reducing memory requirements, and enhancing generalization, that is, finding a balance between local versus non-local networks, is a key problem in network learning.

The trade-offs involved in local learning systems are closely related to the well-known *stability-plasticity dilemma* [8], namely how to design a learning system that is "plastic" enough to learn new patterns, and yet is stable enough to remember old learned patterns. Carpenter and Grossberg [9] developed an architectural solution to this question using their adaptive resonance theory (ART), which overcomes the stability-plasticity dilemma by adapting the stored pattern of a category only when the input is sufficiently similar to it.

This paper develops analytical tools necessary to measure the localization properties of a network. Networks that are less prone to interference are called *local* because learning in one region of the input space causes changes in the I/O map in only a small region local to the point of training. This link between interference during the learning process and the localization properties of a network is made explicit by the incorporation of the learning algorithm into the proposed measures of interference and localization. The measure of interference is defined by answering the question, "How much does learning at x affect the I/O map at $x' \neq x$?" A second measure proposed in this paper quantifies the degree of localization of a network, by taking the inverse of the mean squared interference over an input domain. According to this definition, the I/O map of a local network in one region of the input space is less likely to be unlearned when learning moves to another area of the input space. This measure of localization incorporates the learning algorithm, which is appropriate because localization is based on interference, and interference is a side effect of learning, which, in turn, is controlled by the learning algorithm.

We use the localization measure on a variety of network structures, assuming a gradient descent learning algorithm, and show that it agrees with the intuitive interpretation of local networks. A main

contribution of this paper is a theorem showing that given an arbitrarily large number of weights, a single-hidden-layer, multi-layer perceptron (MLP) network with a sigmoidal activation function exists that is as local as desired, that is, on average, learning at one point will affect another point to as small a degree as desired. It is also shown that this property does not affect the universal-approximation ability of such a network. Specifically it is shown that a single-hidden-layer, sigmoidal MLP can be as local as desired, assuming back-propagation, while simultaneously approximating any continuous function over a compact set to any degree desired. Therefore, this result provides a theoretical framework for developing networks that are both *universal approximators* and *universal localizers.*

The paper is organized as follows. In Section 2 we formally define interference and localization. In Section 3 we evaluate the localization properties of radial basis functions (RBF) and multi-layer perceptron (MLP) networks with sigmoidal activation functions. We then present a localization theorem that shows that with a sufficient number of carefully chosen weights, an MLP exists that is as local as desired. This result is then extended in Section 4 to show that, in addition to the universal-localization property, such an MLP retains its universal-approximation ability. Some concluding remarks are presented in Section 5.

2 Definition of Interference and Localization

Consider a network whose input/output map is described by $y = f(\mathbf{x}, \mathbf{w})$, where $y \in \mathbb{R}$ is the output of the network, $\mathbf{x} \in \mathcal{X}$ is the input, $\mathbf{w} \in \mathcal{W}$ is the weight vector, $f : \mathcal{X} \times \mathcal{W} \rightarrow \mathbb{R}$ is a smooth map describing the network topology, $\mathcal{X} \subset \mathbb{R}^n$ is the input domain, and $\mathcal{W} \subset \mathbb{R}^m$ is the weight domain. During supervised learning, the objective is to adjust $\mathbf{w}$ such that the network approximates a desired function $y^* = f^*(\mathbf{x})$. We assume the learning algorithm has the form: $\Delta\mathbf{w} = \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e)$, where $\Delta\mathbf{w}$ is the weight change in discrete time, $\alpha > 0$ is the learning rate constant, $\mathbf{H}$ is the direction for weight change, and $e = y - y^*$ is the approximation error.

Typically, the localization properties of a network are described by the sensitivity of the network output with respect to weight perturbations. In general, however, this sensitivity function is insufficient for characterizing interference and localization because it ignores the learning algorithm, which is responsible for the weight perturbations. To incorporate the learning algorithm into a measure of

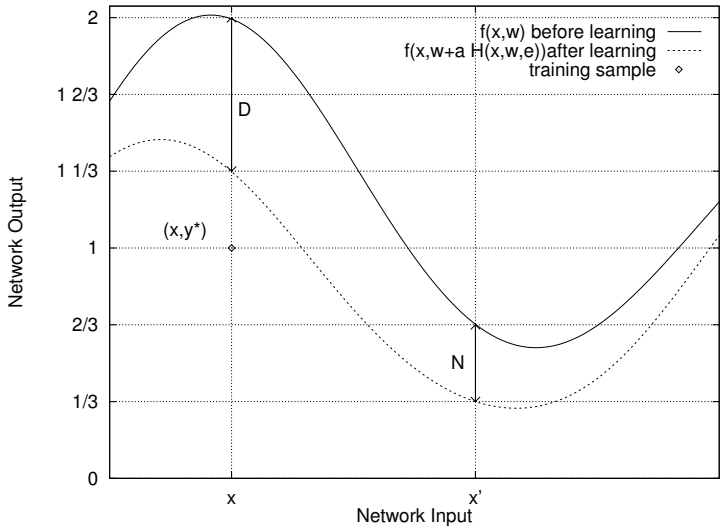

Figure 1: Illustrative example of how learning at x may cause interference at x'.

interference, consider what happens during one weight update. Given a training input/output sample $(\mathbf{x}, y^*)$, the current weight $\mathbf{w}$ is updated to a new weight $\mathbf{w} + \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e)$. At the point $\mathbf{x}$, where the network is trained, the network map changes from $f(\mathbf{x}, \mathbf{w})$ to $f(\mathbf{x}, \mathbf{w} + \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e))$. During this weight update the network I/O map is also affected at other points such as $\mathbf{x}' \neq \mathbf{x}$. We first define a measure for interference between points which is later used to derive a measure for localization of the overall network. To formulate a measure of interference at $\mathbf{x}'$ due to learning at $\mathbf{x}$, consider the ratio ρ of the change in the I/O map at $\mathbf{x}'$ divided by the change in the I/O map at $\mathbf{x}$ due to learning at $\mathbf{x}$; that is,

$$\rho \stackrel{\triangle}{=} \frac{f(\mathbf{x}', \mathbf{w}) - f(\mathbf{x}', \mathbf{w} + \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e))}{f(\mathbf{x}, \mathbf{w}) - f(\mathbf{x}, \mathbf{w} + \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e))}. \tag{1}$$

To illustrate the derivation of a measure of interference, consider Figure 1 which graphically depicts a typical network's output before and after a weight update. The ratio in (1) is constructed by dividing the change in the output at $\mathbf{x}'$, which in this example is $N = \frac{1}{3}$, by the change in output at $\mathbf{x}$, which is $D = \frac{2}{3}$. The ratio $\rho = N/D = 1/2$ is a scalar quantity that represents how much learning at $\mathbf{x}$ interferes with what is known at $\mathbf{x}'$.

In general, ρ is not a useful measure of interference because (i) ρ depends on an arbitrary learning rate α, (ii) ρ depends on an arbitrary desired training sample $(\mathbf{x}, y^*)$ via e, and (iii) ρ is undefined when the denominator is zero. To redress the first two deficiencies, we take the limit of ρ as α approaches zero

and set e to one. This choice of e does not affect $\lim_{\alpha \to 0} \rho$ for algorithms such as gradient descent on the standard quadratic cost function, $J = \frac{1}{2}e^2$, (back-propagation) where the resulting weight change is $\Delta \mathbf{w} = -\alpha e \nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$. In this case, the particular choice of e is irrelevant because it can be subsumed into α, which approaches zero. Finally, if the $\lim_{\alpha \to 0} \rho|_{e=1}$ does not exist, we define interference to be zero because (the attempt at) learning at $\mathbf{x}$ does not affect the output at $\mathbf{x}'$. Based on the above modifications, a general definition of interference is given.

Definition 1 *Let f represent a network I/O map with weight vector $\mathbf{w}$ which is updated according to a generic learning algorithm $\Delta \mathbf{w} = \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, e)$. Then the interference (with unit error) at $\mathbf{x}'$ due to learning at $\mathbf{x}$ is denoted by $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}, \mathbf{x}')$ and is defined as*

$$\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}, \mathbf{x}') \triangleq \begin{cases} \lim_{\alpha \to 0} \frac{f(\mathbf{x}', \mathbf{w}) - f(\mathbf{x}', \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))}{f(\mathbf{x}, \mathbf{w}) - f(\mathbf{x}, \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))} & \textit{if limit exists} \\ 0 & \textit{otherwise} \end{cases} \tag{2}$$

Interference, according to this definition, is a ratio of the change in the (network) output at $\mathbf{x}'$ divided by the change in the output at $\mathbf{x}$ due to learning at $\mathbf{x}$.

Therefore, this definition provides a measure of the degree to which training at an input point $\mathbf{x}$ influences the input/output function of the network at other points $\mathbf{x}'$. In general, the interference measure $\mathcal{I}$ can take any real value. In the case that $\mathcal{I}$ is negative, it represents opposite signs for the change in the output at $\mathbf{x}$ and $\mathbf{x}'$. It is worth noting that, in general, $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}, \mathbf{x}') \neq \mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}', \mathbf{x})$ indicating a non-symmetric behavior between learning at $\mathbf{x}$ and learning at $\mathbf{x}'$. As expected, in the special case that $\mathbf{x} = \mathbf{x}'$ the interference $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}, \mathbf{x}') = 1$.

For any network function approximator, f, that has a well defined gradient (with respect to $\mathbf{w}$) everywhere in $\mathcal{X}$, applying L'Hospital's rule to (2) gives

$$\begin{aligned} \lim_{\alpha \to 0} \frac{f(\mathbf{x}', \mathbf{w}) - f(\mathbf{x}', \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))}{f(\mathbf{x}, \mathbf{w}) - f(\mathbf{x}, \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))} &= \lim_{\alpha \to 0} \frac{\frac{\partial}{\partial \alpha}[f(\mathbf{x}', \mathbf{w}) - f(\mathbf{x}', \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))]}{\frac{\partial}{\partial \alpha}[f(\mathbf{x}, \mathbf{w}) - f(\mathbf{x}, \mathbf{w} + \alpha \mathbf{H}(\mathbf{x}, \mathbf{w}, 1))]} \\ &= \left(\frac{\nabla_{\mathbf{w}} f(\mathbf{x}', \mathbf{w}) \cdot \mathbf{H}(\mathbf{x}, \mathbf{w}, 1)}{\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w}) \cdot \mathbf{H}(\mathbf{x}, \mathbf{w}, 1)} \right) \end{aligned} \tag{3}$$

where $\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$ is the gradient vector of $f(\mathbf{x}, \mathbf{w})$ with respect to $\mathbf{w}$, leading to an equivalent yet

simpler form of the interference measure given by

$$\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x},\mathbf{x}') \stackrel{\triangle}{=} \begin{cases} \left(\frac{\nabla_{\mathbf{w}} f(\mathbf{x}',\mathbf{w})\cdot\mathbf{H}(\mathbf{x},\mathbf{w},1)}{\nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w})\cdot\mathbf{H}(\mathbf{x},\mathbf{w},1)}\right) & \text{if } \nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w})\cdot\mathbf{H}(\mathbf{x},\mathbf{w},1) \neq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

In the special case that the learning algorithm is gradient descent applied to the standard quadratic cost function (back-propagation), $\mathbf{H}(\mathbf{x},\mathbf{w},e) = -e\nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w})$. Hence, in this case (4) reduces to

$$\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x},\mathbf{x}') = \begin{cases} \frac{\nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w})\cdot\nabla_{\mathbf{w}} f(\mathbf{x}',\mathbf{w})}{||\nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w})||^2} & \text{if } \nabla_{\mathbf{w}} f(\mathbf{x},\mathbf{w}) \neq \mathbf{0} \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The above definition of interference measure, given equivalently by (2) and (4), provides the underlying framework for defining a measure of localization, which is done next. Specifically, interference is defined as a function of two points $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, while localization is defined over the entire input domain $\mathcal{X}$. A definition for network localization, given below, provides a measure of how immune a network is to interference.

Definition 2 *Let f represent a network I/O map with weight vector $\mathbf{w}$ which is updated according to the learning algorithm $\Delta\mathbf{w} = \alpha\mathbf{H}(\mathbf{x},\mathbf{w},e)$. Then the localization of the network over an input domain $\mathcal{X}$ is denoted by $L_{f,\mathbf{w},\mathbf{H},\mathcal{X}}$ and is defined as*

$$L_{f,\mathbf{w},\mathbf{H},\mathcal{X}} \stackrel{\triangle}{=} 1/\bar{\mathcal{I}}_{f,\mathbf{w},\mathbf{H},\mathcal{X}} \tag{6}$$

where $\bar{\mathcal{I}}_{f,\mathbf{w},\mathbf{H},\mathcal{X}} \stackrel{\triangle}{=} E[\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x},\mathbf{x}')^2]$ and $E[\cdot]$ is the expected value over all $\mathbf{x}$ and $\mathbf{x}'$ chosen from some probability density function (pdf) over the input domain $\mathcal{X}$.

If the pdf of both $\mathbf{x}$ and $\mathbf{x}'$ is uniformly distributed over $\mathcal{X}$, (6) becomes

$$L_{f,\mathbf{w},\mathbf{H},\mathcal{X}} = \left[\int_{\mathcal{X}}\int_{\mathcal{X}} \mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x},\mathbf{x}')^2 d\mathbf{x} d\mathbf{x}'\right]^{-1}. \tag{7}$$

In general, the localization measure $L_{f,\mathbf{w},\mathbf{H},\mathcal{X}}$ can take any positive real value. Large values of $L_{f,\mathbf{w},\mathbf{H},\mathcal{X}}$ indicate the network is more local over the domain $\mathcal{X}$. This definition transforms a measure

of interference (between two points in the input domain) into a measure of localization (of a network over the entire input domain). The interference and localization measures of different networks are further illustrated below by specific examples. To simplify the analysis, in the rest of the paper we assume that the algorithm $\mathbf{H}(\mathbf{x}, \mathbf{w}, e)$ is gradient descent applied to the quadratic cost function.

3 Application of the Theory of Localization

The previous section developed interference and localization measures based on the learning algorithm. But it should be emphasized that these measures are also a function of the network architecture and weights. We demonstrate this with the following simple example which confirms our intuition that decreasing the widths (dilation) of a Gaussian RBF increases its degree of localization. This example also provides a better understanding of the definition of interference given in (2).

Example 1: Consider a single-input, single-output RBF network with 8 nodes, each of which has an adjustable amplitude a_i, inverted width b_i, and center c_i. The learning algorithm is gradient descent on the quadratic error surface, and the network has the form

$$f(x, \mathbf{w}) = \sum_{i=1}^{8} a_i e^{-((x-c_i)b_i)^2} \tag{8}$$

where $\mathbf{w} = [a_1 \cdots a_8 \; b_1 \cdots b_8 \; c_1 \cdots c_8]^T$, and $a_i = 1$, $b_i = 1$, $c_i = i/7$, for $i \in \{0, \cdots 7\}$. The centers are equally spaced along the input domain $\mathcal{X} = [0, \; 1]$. The plot of $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(x, x')$ shown in Figure 2(a) illustrates the localization properties of this network, with $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(x, x') = 1$ when $x = x'$ and $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(x, x') < 1$ when $x \neq x'$. Now we consider a new weight vector

$$\bar{\mathbf{w}} = [a_1 \cdots a_8 \; 2b_1 \cdots 2b_8 \; c_1 \cdots c_8]^T \tag{9}$$

whose widths are one-half the size of those in $\mathbf{w}$. As can be seen from Figure 2(b), the network exhibits more local properties since the average squared $\mathcal{I}_{f,\bar{\mathbf{w}},\mathbf{H}}(x, x')$ is less over the domain $\mathcal{X}$. By computing the localization measure for each network of the form (8), with $\mathbf{w}$ and $\bar{\mathbf{w}}$ respectively, a comparison of $L_{f,\mathbf{w},\mathbf{H},\mathcal{X}} = 1.475$ and $L_{f,\bar{\mathbf{w}},\mathbf{H},\mathcal{X}} = 2.298$ reveals that $f(x, \bar{\mathbf{w}})$ is more local. $\diamond$

Although these results are not surprising, they do lead to an interesting question: Since the weight

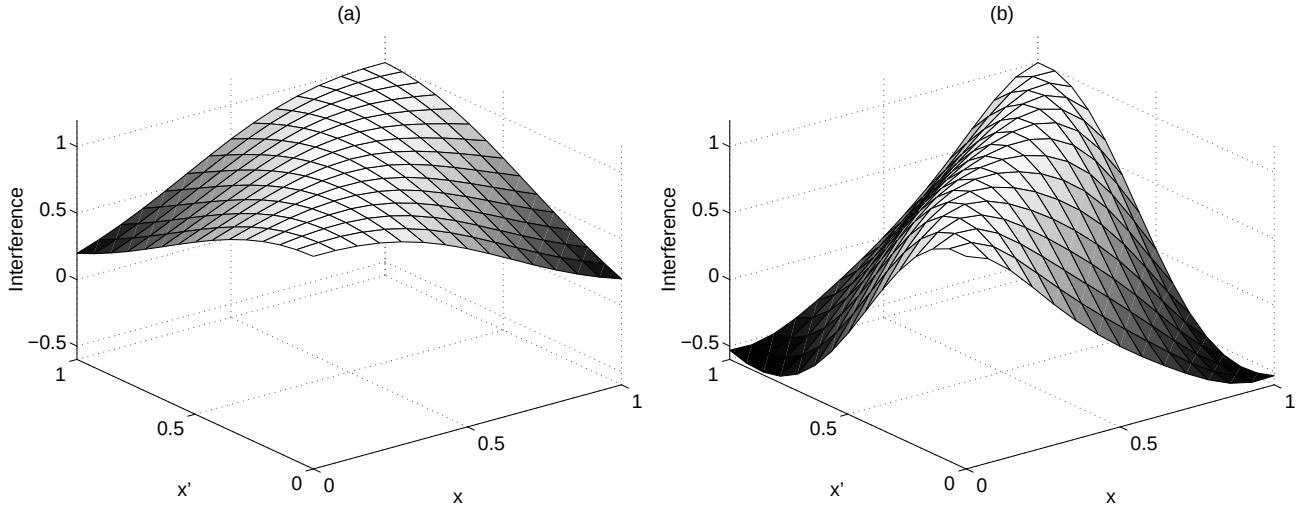

Figure 2: Interference for an eight-node RBF with (a) wide widths and (b) narrow widths.

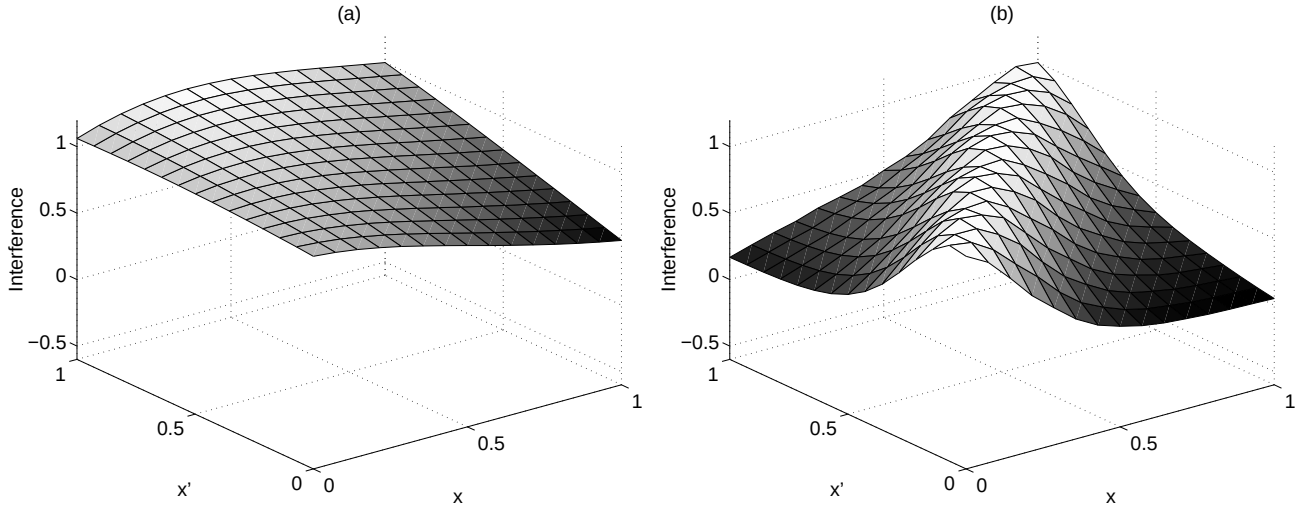

Figure 3: Interference for an eight-node, sigmoidal, single-hidden-layer MLP with (a) non-local and (b) local weight configurations.

configuration affects how local an RBF is, would the same be true of a single-hidden-layer MLP with a sigmoidal activation function? The following example shows that, indeed, different weight configurations produce different degrees of localization.

Example 2: Consider a single-input, single-output eight-node, single-hidden-layer MLP with sigmoidal activation functions whose network architecture is given as

$$f(x, \mathbf{w}) = \sum_{i=1}^{8} a_i (1 + e^{c_i - b_i x})^{-1} \tag{10}$$

where $\mathbf{w} = [a_1 \cdots a_8 \; b_1 \cdots b_8 \; c_1 \cdots c_8]^T$, and $a_i = 7$, $b_i = .7$, $c_i = i/7$, for $i \in \{0, \cdots 7\}$. Figure 3(a) plots $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(x, x')$ showing poor local properties as can be seen by noting that interference is greater than .4 over the entire domain. Using a new weight vector,

$$\bar{\mathbf{w}} = [a_1 \cdots a_8 \; 14b_1 \cdots 14b_8 \; 14c_1 \cdots 14c_8]^T \tag{11}$$

does indeed increase the network's localization, producing $L_{f,\mathbf{w},\mathbf{H},\mathcal{X}} = 1.059$ and $L_{f,\bar{\mathbf{w}},\mathbf{H},\mathcal{X}} = 2.362$. In fact, $f(x, \bar{\mathbf{w}})$ is more local than the corresponding local RBF network shown in Example 1. $\diamond$

Although the motivation provided for this example was to show that the localization properties of sigmoidal MLPs are dependent on the weights, the example raises a new question: How local can sigmoidal MLPs be? The following theorem shows that sigmoidal MLPs can be arbitrarily local by proving analytically that given a sufficiently large number of weights, a single-hidden-layer MLP exists that is arbitrarily local. The theorem assumes the back-propagation algorithm on an MLP given by

$$f(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N} a_i \left(1 + e^{c_i - \sum_{j=1}^{n} b_{ij} x_j}\right)^{-1} \tag{12}$$

whose weight vector is

$$\mathbf{w} = [\mathbf{a}^T \; \mathbf{b}^T \; \mathbf{c}^T]^T \tag{13}$$

where $\mathbf{a} = [a_1 \cdots a_N]^T$, $\mathbf{c} = [c_1 \cdots c_N]^T$, $\mathbf{b} = [\mathbf{b}_1 \cdots \mathbf{b}_N]^T$ and $\mathbf{b}_i = [b_{i1} \cdots b_{in}]$ for $i = 1, .., N$ and the input, $\mathbf{x} = [x_1 \cdots x_n]^T$ is in the domain $\mathcal{X}$.

Theorem 1 *(Universal Localization) Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^n$ and $\mathbf{H} = -e\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$. Then*

for arbitrary $M > 0$, there exist an integer N and real weights given by (13) as a_i, $c_i (i = 1, ..., N)$, and $b_{ij} (i = 1, ..., N)(j = 1, ..., n)$, such that (12) satisfies

$$L_{f,\mathbf{w},\mathbf{H},\mathcal{X}} > M. \tag{14}$$

The proof is given in the Appendix.

To illustrate how a network's localization measure is influenced, we compare interference in the RBF and MLP networks used in Examples 1 and 2, again assuming the back-propagation learning algorithm. Using the RBF network, we find the partial derivatives of the output with respect to a representative amplitude weight a_i, inverted width weight b_i, and center weight c_i as shown in Figure 4. According to equation (5) and by continuity of f, if $|x - x'|$ is small then $\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(x, x') \approx 1$. If $|x - x'|$ is large then it is not possible for corresponding elements of the two vectors $\nabla_{\mathbf{w}} f(x, \mathbf{w})$ and $\nabla_{\mathbf{w}} f(x', \mathbf{w})$ to both be large, which can be seen by noting that all three plots of Figure 4 go to zero as x and x' separate, therefore $\nabla_{\mathbf{w}} f(x, \mathbf{w}) \cdot \nabla_{\mathbf{w}} f(x', \mathbf{w})$ will be small. The magnitude of $\nabla_{\mathbf{w}} f(x, \mathbf{w})$ will not be near zero, however, if for every x there is a corresponding element in $\nabla_{\mathbf{w}} f(x, \mathbf{w})$ whose magnitude is large, a reasonable assumption for a useful local network[1]. Therefore, in this analysis of (5), the numerator will be small compared to the denominator when $|x - x'|$ is large, producing a local network. For basis functions whose widths are small we see according to (5) interference is small and that a small perturbation in a single weight affects only a small portion of the I/O map. As width weights are decreased one can see how an RBF network becomes more local as shown in Example 1.

Repeating this analysis for single-hidden-layer MLPs leads to Figure 5. Figure 5(a) shows non-local properties because it is possible for $\nabla_{\mathbf{w}} f(x, \mathbf{w}) \cdot \nabla_{\mathbf{w}} f(x', \mathbf{w})$ to be large even when x and x' are far from one another. The functions shown in Figure 5(b) and 5(c), however, vanish rapidly at positive and negative infinity and therefore exhibit local properties (see Sjoberg et al. [3]). The network architecture and weights help determine the interference that occurs and hence play an important role in determining network localization.

[1] Baker and Farrell [4] use the term "coverage" for this condition.

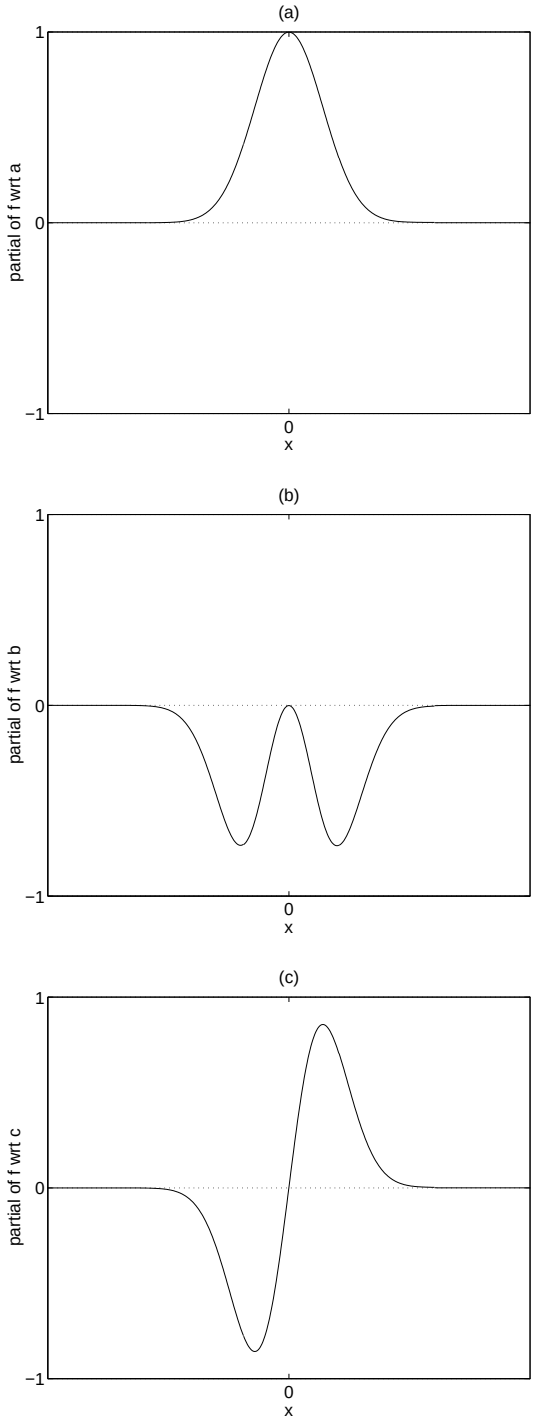

Figure 4: Radial Basis Function partial derivatives of the output $f = \sum_{i=1}^{N} a_i e^{-((x-c_i)b_i)^2}$ with respect to an amplitude a_i, inverted width b_i, and center c_i.

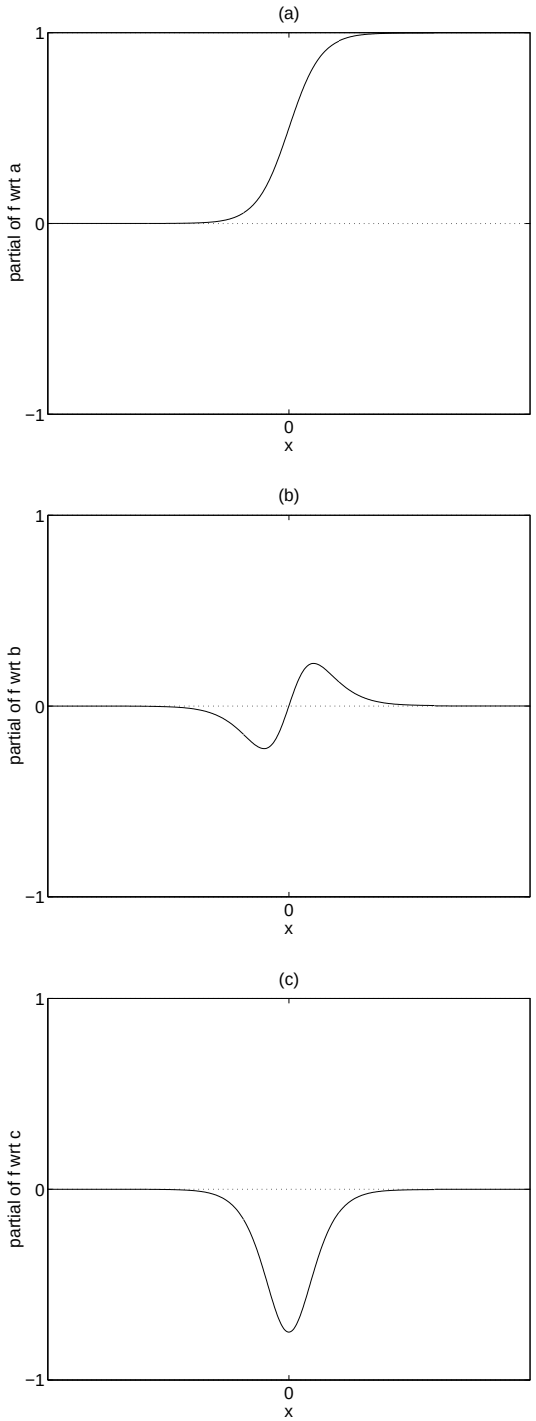

Figure 5: Sigmoidal MLP network partial derivatives of the output $f = \sum_{i=1}^{N} a_i(1 + e^{c_i - b_i x})^{-1}$ with respect to an amplitude a_i, steepness (at the center) b_i, and center c_i.

A special case occurs for *linearly parameterized* networks, that is, networks whose input/output characteristics are expressed as

$$f(\mathbf{x}, \mathbf{w}) = \mathbf{w} \cdot \xi(\mathbf{x}), \tag{15}$$

where $\xi : \mathcal{X} \rightarrow \mathbb{R}^m$ is the *basis vector*. Examples of linearly parameterized approximation structures include polynomials and RBFs with fixed centers and widths. For linearly parameterized networks, $\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w}) = \xi(\mathbf{x})$ which is independent of the weight vector $\mathbf{w}$. Therefore, the measure of interference given by (5) becomes

$$\mathcal{I}_{f,\mathbf{w},\mathbf{H}}(\mathbf{x}, \mathbf{x}') = \begin{cases} \frac{\xi(\mathbf{x}) \cdot \xi(\mathbf{x}')}{\|\xi(\mathbf{x})\|^2} & \text{if } \xi(\mathbf{x}) \neq \mathbf{0} \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

which is also independent of the weights. Clearly, the localization measure will also be independent of $\mathbf{w}$, implying that for linearly parameterized networks modifying its weights will not affect its localization properties. For example, an RBF network with fixed centers and widths cannot become more or less local by changing the amplitude weights.

4 Localization and Approximation

In the preceding section only the local properties of a network were discussed; theorem 1 only addresses the degree of localization found in the network. Because these networks are operating as function approximators it is imperative that the issue of approximation be addressed in conjunction with localization. This leads us to a new question, "Does there exist a single-hidden-layer MLP network that is arbitrarily local *and* approximates any smooth function arbitrarily closely in a compact set?" The following theorem combines the universal-localization theorem of Section 3 and well-known, universal-approximation results of single-hidden-layer sigmoidal MLPs. (See, for example, [10, 11, 12]).

Theorem 2 *Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^n$, $\mathbf{H} = -e\nabla_{\mathbf{w}} h(\mathbf{x}, \mathbf{w})$, and $g^*(\mathbf{x})$ be a real valued continuous function on $\mathcal{X}$. Then for arbitrary $\epsilon > 0, M > 0$, there exist an integer N and real constants given by (13) as a_i, $c_i (i = 1, ..., N)$, and $b_{ij} (i = 1, ..., N)(j = 1, ..., n)$, such that $h(\mathbf{x}, \mathbf{w}) =$*

$\sum_{i=1}^{N} a_i \left(1 + e^{c_i - \sum_{j=1}^{n} b_{ij} x_j}\right)^{-1}$ *satisfies*

$$\max_{\mathbf{x} \in \mathcal{X}} |h(\mathbf{x}, \mathbf{w}) - g^*(\mathbf{x})| < \epsilon \tag{17}$$

and

$$L_{h,\mathbf{w},\mathbf{H},\mathcal{X}} > M. \tag{18}$$

Proof. This proof is a combination of the proof of Theorem 1 and a universal-approximation theorem in [10], which says that given a real-valued continuous function $g^*(\mathbf{x})$ on $\mathcal{X}$, then for arbitrary $\epsilon > 0$ there exist a number of nodes N, and weights $\mathbf{w}$, such that $g(\mathbf{x}, \mathbf{w}) = \sum_{i=1}^{N} a_i (1 + e^{c_i - \sum_{j=1}^{n} b_{ij} x_j})^{-1}$ satisfies $\max_{\mathbf{x} \in \mathcal{X}} |g(\mathbf{x}, \mathbf{w}) - g^*(\mathbf{x}))| < \epsilon$. Using this weight vector $\mathbf{w}$ and function $g(\mathbf{x}, \mathbf{w})$, we construct a sequence (indexed by k) of functions

$$h_k(\mathbf{x}, \bar{\mathbf{w}}_k) \stackrel{\triangle}{=} f_k(\mathbf{x}, \mathbf{w}_k) - f_k(\mathbf{x}, \mathbf{w'}_k) + g(\mathbf{x}, \mathbf{w}), \tag{19}$$

where $\bar{\mathbf{w}}_k \stackrel{\triangle}{=} [\mathbf{w}_k{}^T \mathbf{w'}_k{}^T \mathbf{w}^T]^T$, f_k, and $\mathbf{w}_k$ are defined in (26) and (34). Then interference, $\mathcal{I}_{h_k, \bar{\mathbf{w}}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x'})$, is given by

$$\left(\frac{\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x'}, \mathbf{w}_k) + \nabla_{\mathbf{w'}_k} f_k(\mathbf{x}, \mathbf{w'}_k) \cdot \nabla_{\mathbf{w'}_k} f_k(\mathbf{x'}, \mathbf{w'}_k) + \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w}) \cdot \nabla_{\mathbf{w}} g(\mathbf{x'}, \mathbf{w})}{\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) + \nabla_{\mathbf{w'}_k} f_k(\mathbf{x}, \mathbf{w'}_k) \cdot \nabla_{\mathbf{w'}_k} f_k(\mathbf{x}, \mathbf{w'}_k) + \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w}) \cdot \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w})} \right) \tag{20}$$

If we then set $\mathbf{w'}_k = \mathbf{w}_k$ we see the function $h_k(\mathbf{x}, \bar{\mathbf{w}}_k)$ has the same I/O map as $g(\mathbf{x}, \mathbf{w})$ for all k which proves the universal-approximation condition (17) and because $\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) = \nabla_{\mathbf{w'}_k} f_k(\mathbf{x}, \mathbf{w'}_k)$, (20) reduces to

$$\mathcal{I}_{h_k, \bar{\mathbf{w}}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x'}) = \left(\frac{2 \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x'}, \mathbf{w}_k) + \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w}) \cdot \nabla_{\mathbf{w}} g(\mathbf{x'}, \mathbf{w})}{2 \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) + \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w}) \cdot \nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w})} \right) \tag{21}$$

Because (21) is similar to (27) we borrow from the proof of Theorem 1. The denominator of (27) equals the denominator of (35), which approaches infinity as k approaches infinity (as can be seen by noting

that $\hat{x}\|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\|^2$ is bounded below by B_1 as shown using (40) and the definition of $\hat{x}$). Therefore

$$\lim_{k\to\infty} 2\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) = \infty \tag{22}$$

and because $\nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w})$ and $\nabla_{\mathbf{w}} g(\mathbf{x}', \mathbf{w})$ are positive and independent of k, we see that

$$\lim_{k\to\infty} \frac{\nabla_{\mathbf{w}} g(\mathbf{x}, \mathbf{w}) \cdot \nabla_{\mathbf{w}} g(\mathbf{x}', \mathbf{w})}{2\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k)} \;=\; 0. \tag{23}$$

Using (27) and (45) we see that

$$\lim_{k\to\infty} \mathcal{I}_{h_k, \bar{\mathbf{w}}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}') = 0 \qquad \text{almost everywhere.} \tag{24}$$

Because $\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')$ is bounded via (41) one can see that (21) is also bounded. Using the Lebesgue Dominated Convergence Theorem (as was done in the appendix) one can show

$$\lim_{k\to\infty} L_{h_k, \bar{\mathbf{w}}_k, \mathbf{H}, \mathcal{X}} = \infty. \tag{25}$$

$\diamond$

5 Conclusion

The measures of localization and interference proposed in this paper provide a framework of network localization based on the ability of the network to deter interference during learning. This framework allows us to see that since different learning algorithms give different degrees of interference in a network, then a measure of localization should incorporate the learning algorithm as well as the network architecture and weights. The proposed measure of localization can be applied to any continuous approximating structure of the form $f(\mathbf{x}, \mathbf{w})$ and any learning algorithm of the form $\Delta\mathbf{w} = \alpha\mathbf{H}(\mathbf{x}, \mathbf{w}, e)$, where $\mathbf{x}$ is the input vector, $\mathbf{w}$ represents the adjustable weights and $\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$ is defined everywhere on the input domain.

We show that this framework is consistent with other descriptions of local learning in the litera-

ture. When applied to radial basis function networks it delivers results consistent with the literature's heuristic understanding of local networks. We also apply these measures to sigmoidal networks and show, for back-propagation as the learning algorithm, that a single-hidden-layer sigmoidal network can approximate any desired continuous function and be arbitrarily local, provided an arbitrarily large number of weights are available. This result may give designers confidence to use MLPs for applications previously delegated to networks whose localization property is easier to visualize, such as RBFs. Applying this localization theory to other network-architecture/learning-algorithm combinations may uncover other useful localization properties.

Since interference is, in general, a function of the weight vector, it may be incorporated into a cost function, leading to new learning algorithms that, in addition to approximation, optimize the localization properties of a network. Although the number of weights and the degree of localization are not directly related, we can say that the number of weights is related to the potential for a network to be local. A network's extra degrees of freedom (in the form of extra weights) may be used to make a network more local, while simultaneously approximating a desired function. In certain applications where interference causes problems, this learning algorithm may decrease training times. Exploring such algorithms is an area for future research.

Appendix: Proof of Theorem 1

We prove Theorem 1 by construction. A sequence of sigmoidal networks based on (12)

$$f_k(\mathbf{x}, \mathbf{w}_k) = \sum_{i=1}^{N_k} a_{i,k} \left(1 + e^{c_{i,k} - \sum_{j=1}^{n} b_{ij,k} x_j}\right)^{-1} \tag{26}$$

is used to generate a sequence of interferences using (5)

$$\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}') = \left(\frac{\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}', \mathbf{w}_k)}{\nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k) \cdot \nabla_{\mathbf{w}_k} f_k(\mathbf{x}, \mathbf{w}_k)}\right) \tag{27}$$

where $\mathbf{H} = -e\nabla_{\mathbf{w}} f(\mathbf{x}, \mathbf{w})$, that is, the learning algorithm is back-propagation. We assume without loss of generality that the ratio on the right-hand-side of (27) exists. If the ratio does not exist,

then according to (5), $\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}')$ is zero and the arguments of this proof remain valid. With an appropriate choice of $\mathbf{w}_k$ and N_k, we show the following two statements:

1. $\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}')$ is bounded for all $k \in \mathbb{N},\ \mathbf{x},\mathbf{x}' \in [0,\ 1]^n$.
2. The limit of $\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}')$ as k approaches infinity is zero almost everywhere.

Expanding (27) allows us to write $\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}')$ as

$$\frac{\sum_{i=1}^{N_k}\left[\frac{\partial}{\partial a_{i,k}}f_k(\mathbf{x},\mathbf{w}_k)\ \frac{\partial}{\partial a_{i,k}}f_k(\mathbf{x}',\mathbf{w}_k)+\sum_{j=1}^{n}\frac{\partial}{\partial b_{ij,k}}f_k(\mathbf{x},\mathbf{w}_k)\ \frac{\partial}{\partial b_{ij,k}}f_k(\mathbf{x}',\mathbf{w}_k)+\frac{\partial}{\partial c_{i,k}}f_k(\mathbf{x},\mathbf{w}_k)\ \frac{\partial}{\partial c_{i,k}}f_k(\mathbf{x}',\mathbf{w}_k)\right]}{\sum_{i=1}^{N_k}\frac{\partial}{\partial a_{i,k}}f_k(\mathbf{x},\mathbf{w}_k)^2+\sum_{i=1}^{N_k}\sum_{j=1}^{n}\frac{\partial}{\partial b_{ij,k}}f_k(\mathbf{x},\mathbf{w}_k)^2+\sum_{i=1}^{N_k}\frac{\partial}{\partial c_{i,k}}f_k(\mathbf{x},\mathbf{w}_k)^2}$$

and by defining

$$y_{i,k}(\mathbf{x},\mathbf{w}_k) \stackrel{\triangle}{=} \left(1+e^{c_{i,k}-\sum_{j=1}^{n}b_{ij,k}x_j}\right)^{-1} \tag{28}$$

as the output of the ith node in the hidden layer, we see that we can write $\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}')$ as

$$\frac{\sum_{i=1}^{N_k}\left[y_{i,k}(\mathbf{x},\mathbf{w}_k)y_{i,k}(\mathbf{x}',\mathbf{w}_k)+a_{i,k}^2(1+\sum_{j=1}^{n}x_jx_j')y_{i,k}(\mathbf{x},\mathbf{w}_k)y_{i,k}(\mathbf{x}',\mathbf{w}_k)(1-y_{i,k}(\mathbf{x},\mathbf{w}_k))(1-y_{i,k}(\mathbf{x}',\mathbf{w}_k))\right]}{\sum_{i=1}^{N_k}\left[y_{i,k}(\mathbf{x},\mathbf{w}_k)^2+a_{i,k}^2(1+\sum_{j=1}^{n}x_j{}^2)y_{i,k}(\mathbf{x},\mathbf{w}_k)^2(1-y_{i,k}(\mathbf{x},\mathbf{w}_k))^2\right]}. \tag{29}$$

Simplifying further by letting

$$z_{i,k}(\mathbf{x},\mathbf{w}_k) \stackrel{\triangle}{=} y_{i,k}(\mathbf{x},\mathbf{w}_k)(1-y_{i,k}(\mathbf{x},\mathbf{w}_k)), \tag{30}$$

$$\hat{x}' \stackrel{\triangle}{=} 1+\sum_{j=1}^{n}x_jx_j', \tag{31}$$

and

$$\hat{x} \stackrel{\triangle}{=} 1+\sum_{j=1}^{n}x_j{}^2, \tag{32}$$

we see

$$\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}') = \frac{\sum_{i=1}^{N_k}\left[y_{i,k}(\mathbf{x},\mathbf{w}_k)y_{i,k}(\mathbf{x}',\mathbf{w}_k)+a_{i,k}^2\hat{x}'z_{i,k}(\mathbf{x},\mathbf{w}_k)z_{i,k}(\mathbf{x}',\mathbf{w}_k)\right]}{\sum_{i=1}^{N_k}\left[y_{i,k}(\mathbf{x},\mathbf{w}_k)^2+a_{i,k}^2\hat{x}z_{i,k}(\mathbf{x},\mathbf{w}_k)^2\right]}. \tag{33}$$

Let the weights, $\mathbf{w}_k$, and number of nodes for our constructed sequence of sigmoidal networks be selected as

$$a_{i,k} = 2^k, \quad b_{ij,k} = 2^k, \quad c_{i,k} = i, \text{ and } \quad N_k = n2^k. \tag{34}$$

Letting $\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k) = [z_{1,k}(\mathbf{x}, \mathbf{w}_k), ..., z_{N_k,k}(\mathbf{x}, \mathbf{w}_k)]^T$ and $\mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) = [y_{1,k}(\mathbf{x}, \mathbf{w}_k), ..., y_{N_k,k}(\mathbf{x}, \mathbf{w}_k)]^T$, we recall that $\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k) \cdot \mathbf{z}_k(\mathbf{x}', \mathbf{w}_k) = \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\| \, \|\mathbf{z}_k(\mathbf{x}', \mathbf{w}_k)\| \, \cos\theta$, where θ is the angle between the vectors $\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)$ and $\mathbf{z}_k(\mathbf{x}', \mathbf{w}_k)$. Therefore we can write

$$\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) \cdot \mathbf{y}_k(\mathbf{x}', \mathbf{w}_k) + 2^{2k}\hat{x}' \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\| \|\mathbf{z}_k(\mathbf{x}', \mathbf{w}_k)\| \cos\theta}{\mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) \cdot \mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) + 2^{2k}\hat{x} \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\|^2} \tag{35}$$

$$= \frac{2^{-2k}\mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) \cdot \mathbf{y}_k(\mathbf{x}', \mathbf{w}_k) + \hat{x}' \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\| \|\mathbf{z}_k(\mathbf{x}', \mathbf{w}_k)\| \cos\theta}{2^{-2k}\mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) \cdot \mathbf{y}_k(\mathbf{x}, \mathbf{w}_k) + \hat{x} \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\|^2} \tag{36}$$

In order to show $\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')$ is bounded, we first find an upper and lower bound on $\|\mathbf{z}_k(\cdot)\|$. Using weight values given in (34) we see $y_{i,k}(\mathbf{x}, \mathbf{w}_k) = [1 + e^{(i - 2^k \bar{x})}]^{-1}$ where $\bar{x} \stackrel{\triangle}{=} \sum_{j=1}^{n} x_j$. Substituting $y_{i,k}(\mathbf{x}, \mathbf{w}_k)$ into (30) and letting $g(s) \stackrel{\triangle}{=} (1 + e^s)^{-2}[1 - (1 + e^s)^{-1}]^2 = e^{2s}(1 + e^s)^{-4}$ gives

$$\|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\|^2 = \sum_{i=1}^{n2^k} g(i - 2^k \bar{x}). \tag{37}$$

Let $i_k \stackrel{\triangle}{=} floor(2^k \bar{x})$, then $a \stackrel{\triangle}{=} 2^k \bar{x} - i_k$ has the property that $0 \leq a < 1$. Using the change of variables $\bar{i} = i - i_k$, to change the bounds on the summation in (37), and because $g(i)$ is non-negative, even, and monotonically decreasing as $|i|$ increases, we see

$$\begin{aligned} \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\|^2 &= \sum_{\bar{i}=1-i_k}^{0} g(\bar{i} - a) + \sum_{\bar{i}=1}^{n2^k - i_k} g(\bar{i} - a) \\ &< \sum_{\bar{i}=1-i_k}^{0} g(\bar{i}) + \sum_{\bar{i}=1}^{n2^k - i_k} g(\bar{i} - 1) \\ &< \sum_{\bar{i}=-\infty}^{0} g(\bar{i}) + \sum_{\bar{i}=1}^{\infty} g(\bar{i} - 1) \\ &= 2\sum_{\bar{i}=0}^{\infty} g(\bar{i}) \end{aligned} \tag{38}$$

where the right-hand side is no longer a function of $\mathbf{x}$ or k. To show $\|\mathbf{z}_k(\cdot)\|$ is bounded from above we use the ratio test (Bartle [13] p. 296) and see that

$$\lim_{i\to\infty}\frac{g(i+1)}{g(i)} = \lim_{i\to\infty} e^2\left(\frac{e^{-i}+1}{e^{-i}+e}\right)^4 = e^{-2} < 1 \tag{39}$$

therefore $\sum_{i=0}^{\infty} g(i)$ is convergent and less then some value B_2 and therefore $\|\mathbf{z}_k(\cdot)\|^2 < 2B_2$.

To establish a lower bound for $\|\mathbf{z}_k(\cdot)\|$ we see that because g is always positive and from (38) we see

$$\|\mathbf{z}_k(\mathbf{x},\mathbf{w}_k)\|^2 \geq g(-a) \geq g(-1) \stackrel{\triangle}{=} B_1 > 0. \tag{40}$$

Because $B_1 > 0$ and both B_1 and B_2 are independent of $\mathbf{x}$ and k we can write $0 < B_1 \leq \|\mathbf{z}_k(\cdot)\|^2 \leq 2B_2$. Using these results and noting that $0 \leq y_{i,k}(\mathbf{x},\mathbf{w}_k) \leq 1$ and hence $\mathbf{y}_k(\mathbf{x},\mathbf{w}_k)\cdot\mathbf{y}_k(\mathbf{x}',\mathbf{w}_k) \leq n2^k$ and because $\hat{x}$ and $\hat{x}'$ are bounded below by 1 and bounded above by $1+n$, we see interference has an upper bound that is not a function of $\mathbf{x}$ or k, that is,

$$\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}') \leq \frac{2^{-2k}n2^k + \hat{x}'\|\mathbf{z}_k(\mathbf{x},\mathbf{w}_k)\|\|\mathbf{z}_k(\mathbf{x}',\mathbf{w}_k)\|}{\hat{x}\|\mathbf{z}_k(\mathbf{x},\mathbf{w}_k)\|^2} \leq \frac{n+(1+n)2B_2}{B_1} \stackrel{\triangle}{=} B \tag{41}$$

for all $\mathbf{x},\mathbf{x}' \in [0,\ 1]^n$. In similar fashion one can show interference is bounded below by $-B$.

Now we show that $\lim_{k\to\infty}\mathcal{I}_{f_k,\mathbf{w}_k,\mathbf{H}}(\mathbf{x},\mathbf{x}') = 0$ almost everywhere. Let $\bar{x}' \stackrel{\triangle}{=} \sum_{j=1}^{n} x'_j$ and consider part of the numerator of (36) we see

$$\hat{x}'\|\mathbf{z}_k(\mathbf{x},\mathbf{w}_k)\|\|\mathbf{z}_k(\mathbf{x}',\mathbf{w}_k)\|\cos\theta \quad = \quad \hat{x}'\sum_{i=1}^{nv}\frac{e^{(i-v\bar{x})}}{(1+e^{(i-v\bar{x})})^2}\frac{e^{(i-v\bar{x}')}}{(1+e^{(i-v\bar{x}')})^2} \tag{42}$$

where $v \stackrel{\triangle}{=} 2^k$. We consider the case where $\bar{x} \neq \bar{x}'$ and, without loss of generality, we assume $\bar{x} < \bar{x}'$ and break the sum of (42) into three parts and use the closed form of the geometric series. Letting $M \stackrel{\triangle}{=} floor(v\bar{x})$ and $M' \stackrel{\triangle}{=} floor(v\bar{x}')$ gives

$$\begin{aligned}&\hat{x}'\|\mathbf{z}_k(\mathbf{x},\mathbf{w}_k)\|\|\mathbf{z}_k(\mathbf{x}',\mathbf{w}_k)\|\cos\theta \\ &\quad\leq \hat{x}'\left(\sum_{i=1}^{M} e^{(i-v\bar{x})}e^{(i-v\bar{x}')} + \sum_{i=M+1}^{M'}\frac{e^{(i-v\bar{x}')}}{e^{(i-v\bar{x})}} + \sum_{i=M'+1}^{nv}\frac{1}{e^{(i-v\bar{x})}e^{(i-v\bar{x}')}}\right)\end{aligned}$$

$$= \hat{x}' \left(e^{-v(\bar{x}+\bar{x}')} \frac{(e^2 - e^{2(M+1)})}{(1-e^2)} + e^{v(\bar{x}-\bar{x}')}(M' - (M+1) + 1) + e^{v(\bar{x}+\bar{x}')} \frac{(e^{-2(M'+1)} - e^{-2(nv+1)})}{(1-e^{-2})} \right)$$
$$= \hat{x}' \left(e^{-v(\bar{x}+\bar{x}')} e^2 \frac{(e^{2M} - 1)}{(e^2-1)} + e^{v(\bar{x}-\bar{x}')}(M' - (M+1) + 1) + e^{v(\bar{x}+\bar{x}')} \frac{(e^{-2(M'+1)} - e^{-2(nv+1)})}{(1-e^{-2})} \right)$$

and using $M \leq v\bar{x} \leq M+1$ and $M' \leq v\bar{x}' \leq M'+1$ leads to

$$\hat{x}' \|\mathbf{z}_k(\mathbf{x}, \mathbf{w}_k)\| \|\mathbf{z}_k(\mathbf{x}', \mathbf{w}_k)\| \cos\theta$$
$$\leq \hat{x}' \left(e^2 \frac{e^{-v(\bar{x}'-\bar{x})} - e^{-v(\bar{x}+\bar{x}')}}{(e^2-1)} + e^{-v(\bar{x}'-\bar{x})}(v(\bar{x}'-\bar{x}) + 1) + \frac{(e^{-v(\bar{x}'-\bar{x})} - e^{-2}e^{-v(2n-\bar{x}-\bar{x}')})}{(1-e^{-2})} \right) \quad (43)$$

and because $0 \leq \bar{x} < \bar{x}' \leq n$, one sees that the right hand side of (43) is a sum of elements of the form $c_1 v^{c_2} e^{-c_3 v}$ where c_1, c_2, c_3 are not functions of v and $c_3 > 0$. With these conditions we can show

$$\lim_{k\to\infty} c_1 v^{c_2} e^{-c_3 v} = 0 \quad (44)$$

which tells us that when $\bar{x} \neq \bar{x}'$, the numerator of (36) approaches zero as k approaches ∞ and because the denominator of (36) is bounded below by a positive constant, B_1, we see

$$\lim_{k\to\infty} \mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}') = 0 \qquad \text{almost everywhere.} \quad (45)$$

Equation (45) holds because the set $S = \{(\mathbf{x}, \mathbf{x}') \in \mathcal{X} \times \mathcal{X} : \bar{x} = \bar{x}'\}$ is of measure zero on $\mathcal{X} \times \mathcal{X}$ because $S \subset \mathbb{R}^{2n-1}$ defines a hyperplane of lower dimension within $\mathcal{X} \times \mathcal{X} \subset \mathbb{R}^{2n}$. At this point we have met the conditions of the Lebesgue Dominated Convergence Theorem : $\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')$ is a sequence of integrable functions on $[0,\ 1]^n \times [0,\ 1]^n$. Because there exists a bound $B > 0$ such that $|\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')| \leq B$ for all $k \in \mathbb{N},\ \mathbf{x}, \mathbf{x}' \in [0,\ 1]^n$ and (45) is an integrable function, we see

$$\lim_{k\to\infty} E[\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')^2] = E[\lim_{k\to\infty} \mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')^2] = 0 \quad (46)$$

for $\mathcal{X} = [0,\ 1]^n$. Equation (46) implies that there exists a k such that $E[\mathcal{I}_{f_k, \mathbf{w}_k, \mathbf{H}}(\mathbf{x}, \mathbf{x}')^2] < \epsilon$ for arbitrary $\epsilon > 0$ and hence $L_{f_k, \mathbf{w}_k, \mathbf{H}, \mathcal{X}}$ can be made arbitrarily large. A simple scaling and translation

will allow for arbitrary compact $\mathcal{X} \subset \mathbb{R}^n$. This completes the proof of Theorem 1.

References

[1] D. Sofge and D. White, "Applied learning: optimal control for manufacturing," in *Handbook of Intelligent Control Neural, Fuzzy, and Adaptive Approaches* (D. White and D. Sofge, eds.), (New York, NY), pp. 259–281, Van Nostrand Reinhold, 1992.

[2] J. H. Friedman, "Local learning based on recursive covering." submitted to *The Annals of Statistics*, Aug. 1996.

[3] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: A unified overview," *Automatica*, vol. 31, pp. 1691–1724, 1995.

[4] W. Baker and J. Farrell, "An introduction to connectionist learning control systems," in *Handbook of Intelligent Control Neural, Fuzzy, and Adaptive Approaches* (D. White and D. Sofge, eds.), (New York, NY), pp. 35–63, Van Nostrand Reinhold, 1992.

[5] R. French, "Dynamically constraining connectionist networks to produce distributed, orthogonal representations to reduce catastrophic interference," in *Proceedings of the 16th Annual Cognitive Science Society Conference*, vol. 5, pp. 207–220, 1994.

[6] A. G. Barto, "Connectionist learning for control," in *Neural Networks for Control* (I. W. Thomas Miller, R. S. Sutton, and P. J. Werbos, eds.), (Massachusetts Institute of Technology, MA), pp. 5–58, MIT Press, 1990.

[7] R. E. Bellman, *Adaptive control processes: A guided tour.* Princeton University, NJ: Princeton University Press, 1961.

[8] M. A. Cohen and S. Grossberg, "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 5, pp. 815–826, 1983.

[9] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Computer Vision, Graphics, and Image Processing*, vol. 37, pp. 54–115, 1987.

[10] K.-I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol. 2, pp. 183–192, 1989.

[11] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 303–314, 1989.

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

[13] R. Bartle, *The Elements of Real Analysis*. John Wiley and Sons, 1976.