XOR Has No Local Minima: A Case Study in Neural Network Error Surface Analysis

Leonard G. C. Hamey

Department of Computing

Macquarie University

NSW 2109 AUSTRALIA

Acknowledgements

The author would like to thank Dr. Michael Johnson for many helpful discussions in the preparation of this paper, and an anonymous referee for many helpful and detailed suggestions. This research was supported in part by Digital Equipment Corporation (Australia) Pty Ltd.

Address for reprint requests

Dr. Leonard G. C. Hamey

Department of Computing

Macquarie University

NSW 2109 AUSTRALIA

phone: +61–2–9850–9527

fax: +61–2–9850–9551

Running title

XOR Has No Local Minima

XOR Has No Local Minima: A Case Study in Neural Network Error Surface Analysis

Abstract

This paper presents a case study of the analysis of local minima in feedforward neural networks. Firstly, a new methodology for analysis is presented, based upon considering trajectories through weight space by which a training algorithm might escape a hypothesized local minimum. This analysis method is then applied to the well known XOR (exclusive-or) problem, which has previously been considered to exhibit local minima. The analysis proves the absence of local minima, eliciting significant aspects of the structure of the error surface. The present work is important for the study of the existence of local minima in feedforward neural networks, and also for the development of training algorithms which avoid or escape entrapment in local minima.

Keywords: Feedforward nets, Error surface, Local minimum, XOR, Exclusive-or

Submitted to Neural Networks, *Mathematical and Computational Analysis*

List of Mathematical Symbols

a Parameterisation for a trajectory or line.

d A discriminant term arising in equations 7 and 8, determining the escape trajectory for class (b) stationary points.

f The XOR network activation function for hidden nodes.

f_O The XOR network activation function for the output node.

g A scalar differentiable function; an error function E or L.

g' The image of g under T.

g_0 The value of g at a point $\mathbf{w}_0$ or the lower bound of g over $M_g(\mathbf{w}_0)$.

g_1 The minimum element of G^+.

g^* A value between $g(\mathbf{d})$ and g_1.

t^α A target value for the XOR network. $t^1 = t^4 = \delta$, $t^2 = t^3 = 1 - \delta$.

u_{ij} XOR network weight between hidden node j and input node i.

v_j XOR network weight between output node and hidden node j.

x_j^α The sum of weighted inputs to hidden node j when the XOR network is presented with pattern I^α.

y_j^α The output of hidden node j. $y_j^\alpha = f(x_j^\alpha)$.

$\overline{y_l}$ The mean output of hidden node l over all four training patterns.

z^α The sum of weighted inputs to the output node when the XOR network is presented with pattern I^α.

z_δ The value such that $f_O(z_\delta) = \delta$.

$A_{\mathbf{w}_1 \to \mathbf{w}_2}$ A trajectory; a continuous function $A_{\mathbf{w}_1 \to \mathbf{w}_2} : \mathbb{R} \to \mathcal{D}$ through $\mathbf{w}_1$ and $\mathbf{w}_2$.

A^* A trajectory constructed in lemma 2.

$A^{*\prime}$ A trajectory in $\mathcal{D}'$ resulting from the application of lemma 2.

B A second trajectory used in lemma 2.

C A trajectory contained in S^*.

E The mean squared error cost function.

G The set of values of minima of g along a trajectory B.

G^+ The subset of G which are greater than $g(\mathbf{d})$.

I^α An XOR network input pattern. $I^1 = (1,1)$, $I^2 = (1,0)$, $I^3 = (0,1)$, $I^4 = (0,0)$.

L The cross-entropy error cost function.

$M_g(\mathbf{w}_0)$ The minimum region of a point $\mathbf{w}_0$ for function g.

O^α The output value computed by the XOR network when presented with input pattern I^α. $O^\alpha = f_O(z^\alpha)$.

Q^+ The boundary of S^+ in S.

R^α Residual error of XOR network on pattern I^α. $R^\alpha = O^\alpha - t^\alpha$.

S A manifold that is a homeomorphic image of the disk, bounded by A and B.

S^+ The connected set of points $\mathbf{w}$ in S where $g(\mathbf{w}) > g^*$ and where $\mathbf{a}$ (the origin of the trajectory A) is in S^+.

S^* The level set of points $\mathbf{w}$ in S where $g(\mathbf{w}) = g^*$.

T A homeomorphic transformation $T : \mathcal{D} \to \mathcal{D}'$ between two domains.

$\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ Points in $\mathcal{D}$.

$\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{u}, \mathbf{v}, \mathbf{w}$ Points in $\mathcal{D}$.

$\mathbf{w}_0$ A point which is or represents a local minimum.

$\mathbf{w}_1, \mathbf{w}_2$ Points in $\mathcal{D}$.

$\mathbf{x}$ A point in $\mathbb{R}^n$.

$\mathcal{D}$ The domain of g.

$\mathcal{E}$ The extension by continuity of domain $\mathcal{D}$.

$\mathbb{R}$ The real numbers.

$\mathbb{R}^n$ n-dimensional Euclidean space.

α Indexes an XOR input pattern; $\alpha = 1, \ldots, 4$.

δ A value in the range $[0, 1]$ used to define bi-valued network targets.

θ_j XOR network bias of hidden node j.

ϕ Bias of the XOR network output node.

ψ A network weight or parameter.

Φ The Gaussian function $\Phi(\mathbf{x}) = e^{-|\mathbf{x}|^2}$.

1 Introduction

The existence of local minima in the error surfaces of feedforward neural networks is an important issue, since learning techniques such as back propagation (Rumelhart, Hinton, & Williams, 1986) may become trapped in local minima. The existence of local minima depends upon both the task being learnt and the network being trained.

An important area of current research is to characterize the conditions under which local minima may or may not occur in the error surfaces of feedforward neural networks. The present paper contributes to this study in three ways. Firstly, the discussion of the definition of local minimum clarifies the distinction between local minimum and plateaus, leading to a more useful definition of *regional local minimum*. Secondly, an analysis methodology is developed for regional local minima based upon consideration of trajectories through weight space by which a training method might escape hypothesized local minima. Thirdly, our analysis of the XOR (exclusive-or) error surface provides a valuable example of the absence of local minima in feedforward neural networks. Together, these contributions extend the understanding of the conditions under which local minima are absent in feedforward neural networks.

A second important area of current research is to develop training algorithms which avoid or escape entrapment in local minima of the error surface. Although such a study is beyond the scope of this paper, the present work contributes to the understanding of entrapment and the structure of neural network error surfaces. Such understanding is valuable in the development of new training algorithms.

Early experiments with feedforward neural networks suggested that local minima do not commonly occur. Rumelhart et al. (1986) "found local minima to be very rare" (p. 352). More recent work demonstrates the existence of local minima in specific situations (Brady, Raghavan, & Slawny, 1989; Sontag & Sussmann, 1989) and even methodology by which many local minima may be constructed for single-node networks (Auer, Herbster, & Warmuth, 1996). In response to these results, several authors (Baldi & Hornik, 1989; Gori & Tesi, 1992; Poston, Lee, Choie, & Kwon, 1991; Sontag & Sussmann, 1991; Yu & Chen, 1995) have proved the absence of local minima under various constraints. However, the gap between these theories is very large, and the possibility of local minima is simply not known for many practical learning tasks. The absence of local minima in the XOR problem, proved

below, provides a valuable result for the refinement of theories of the occurrence of local minima.

The current paper is concerned primarily with the finite weight behaviour of feedforward neural network learning. Although our analysis involves consideration of infinite weights, the results that are obtained relate to finite weight configurations because typical learning algorithms do not allow the weights to become truly infinite. Despite this observation, a finite weight configuration may be entrapped in a local minimum basin which extends to infinity. Our analysis considers this possibility and shows how the consideration of paths through weight space involving infinite weights leads to results for the finite weight case. It is beyond the scope of this paper to consider in detail training algorithms in which infinite weights are effectively realised.

The XOR problem was chosen for the present work for several reasons. Firstly, it is one of the simplest problems that is not covered by prior work on the existence of local minima. It has two hidden nodes—too few to be proven to have no local minima by existing theories (Poston et al., 1991; Yu & Chen, 1995). On the other hand, it is not covered by existing results demonstrating the existence of local minima. The XOR problem is one of the simplest problems which is not linearly separable. It is, however, sufficiently complex for back propagation training of the XOR task to become trapped without achieving a global optimum (Rumelhart et al., 1986).

Secondly, there is a significant body of existing analytical work on the XOR problem. Studies by Blum (1989) and Lisboa and Perantonis (1991) have concluded that the XOR problem exhibits local minima, a view that is widely accepted in the literature (Cetin, Burdick, & Barhen, 1993; Dayhoff, 1990; Gori & Tesi, 1992; Kramer & Sangiovanni-Vincentelli, 1989; Orchard & Phillips, 1991, pp. 82–84; Wasserman, 1989, p. 91). The present paper shows Blum's analysis to be in error, reopening the question of the existence of local minima in the XOR problem.

Thirdly, the analysis of the stationary points performed by Lisboa and Perantonis (1991) provides the ideal basis for the application of our new analysis method. Since our method is a technique for analysing hypothesized local minima, it is applicable in situations where suspected local minima have been identified by other means. The analysis of Lisboa and Perantonis provides a complete identification of the stationary points of the XOR error surface. Our method is then applied to determine that none of these stationary points is a local minimum.

Finally, the widespread use of the XOR problem as an example and a benchmark problem justifies further study despite its apparent simplicity. In particular, Rumelhart et al. (1986) suggest that "many other difficult problems involve an XOR as a subproblem" (p. 330). For these reasons, the XOR problem stands as a significant test case for theories concerning the existence of local minima.

The following sections of this paper develop our new method of analysing local minima and apply it to the XOR problem with two hidden nodes. Section 2 presents the notation used throughout the paper. Section 3 discusses the analyses of the XOR problem performed by Blum (1989) and Lisboa and Perantonis (1991), rebutting the conclusion of the former work that the XOR task exhibits a manifold of local minima. The discussion raises the question: *How should a local minimum be defined?* What is meant by "local minimum" is often not explicitly stated. In section 4 we consider several precise definitions of local minima. Choosing the most intuitively sensible of these, we show in section 5 that the XOR task has no local minima. The proofs, however, involve the treatment of infinite weights in a manner that requires formalization. Interested readers will find the required mathematical foundations in section 6. Others may wish to proceed directly to section 7 where the wider implications of the present result are discussed.

2 Notation

In this paper, the error surface of the XOR network with two hidden nodes (figure 1) is analysed. Our work builds upon the complete catalogue of the stationary points of the XOR task presented by Lisboa and Perantonis (1991), and for convenience, their notation is used.

<<< Insert figure 1 approximately here >>>

The network under consideration has two hidden nodes as shown in figure 1. The weight ϕ is the bias of the output node, v_j is the connecting weight between the output node and hidden node j, θ_j is the bias for hidden node j and u_{ij} is the weight connecting input node i to hidden node j. The four input patterns $I^1 = (1,1)$, $I^2 = (1,0)$, $I^3 = (0,1)$ and $I^4 = (0,0)$ have target values of $t^1 = \delta$, $t^2 = 1-\delta$, $t^3 = 1-\delta$ and $t^4 = \delta$ respectively. The network output is $O^\alpha = f_O(z^\alpha)$ where $z^\alpha = \phi + \sum_j v_j y_j^\alpha$.

The hidden node outputs are $y_j^\alpha = f(x_j^\alpha)$ where $x_j^\alpha = \theta_j + \sum_i u_{ij} I_i^\alpha$ and the activation function is the sigmoid $f(x) = 1/(1+e^{-x})$.

Our results have been derived for a network with an output node which employs the sigmoidal activation function $f_O(z) = 1/(1+e^{-z})$, although they readily generalize to common variants of this activation function such as tanh when the network target values are correspondingly transformed. The results are applicable to both the usual sum of squares cost function

$$E = \frac{1}{2} \sum_\alpha (O^\alpha - t^\alpha)^2 \tag{1}$$

and the "cross-entropy" (Hinton, 1989, p. 207) used by Lisboa and Perantonis (1991)

$$L = -\sum_\alpha \ln \left[(O^\alpha)^{t^\alpha} (1 - O^\alpha)^{1-t^\alpha} \right] \tag{2}$$

when the activation function for the output node is sigmoidal.

Our results also generalize to the case of a linear output node with the activation function $f_O(z) = z + \frac{1}{2}$ when the cost function E is used, since this case is analytically equivalent to the use of a sigmoidal output node with the cost function L.

It should be noted that the approach in section 5 does not rely on the specific form of either the output node activation function f_O or the error cost function. It will suffice for our purposes that f_O is monotonic increasing, invertible and satisfies $f_O(-z) = 1 - f_O(z)$. Similarly, it is sufficient if the error function is the sum of terms that are monotonic functions of $(O^\alpha - t^\alpha)^2$ for each individual α, since our proof involves trajectories in which at most one output value O^α is allowed to vary at a time. The breadth of our analysis is limited by that of Lisboa and Perantonis (1991), since we require a complete classification of the stationary points of the XOR network as a starting point.

3 Previous Work

Blum (1989) proves the existence of neural network solutions to the XOR task and other two-variable functions. They also claim the existence of a manifold of local minima for the XOR task. Their manifold

is a line which, in our context, is defined by $v_1 = v_2 = -\phi, u_{ij} = \theta_j = 0$. They attempt to prove, in their theorem 2, that for all points on this line, the derivative of the error surface along lines radiating from each point is strictly non-negative throughout a small neighborhood of the point. Unfortunately, there is an error in their proof. They employ a Taylor approximation to the derivative and discard the terms of second degree and above. However, when the first-order term is zero (i.e. $\Delta z = 0$ in their notation) then the higher-order terms dominate and there exist radiating lines along which the derivative of the error is negative. In particular, lines parameterized by a in the following equations

$$\begin{aligned} v_1' &= v_2' = -\phi \\ \phi' &= (1 - a/2)\phi \\ \theta_1' &= \theta_2' = -a \\ u_{21}' &= u_{12}' = -a \\ u_{11}' &= u_{22}' = a \end{aligned}$$

have a negative derivative for suitable, sufficiently small t, and lead directly by gradient descent to solution of the XOR task.

Lisboa and Perantonis (1991) present a complete analysis of the stationary points of the XOR task. They also claim the existence of local minima based upon an examination of the second derivatives of the error, but do not provide details of this aspect of their work. They prove, however, that some stationary points occur as asymptotes, leading us to reconsider the definition of local minima.

The stationary points of the XOR task are characterized by equations (3)–(6) of Lisboa and Perantonis (1991) as follows, where $R^\alpha = O^\alpha - t^\alpha$.

$$\partial L/\partial \phi = 0 \quad \Rightarrow \quad \sum_\alpha R^\alpha = 0 \tag{3}$$

$$\partial L/\partial v_j = 0 \quad \Rightarrow \quad \sum_\alpha R^\alpha y_j^\alpha = 0 \tag{4}$$

$$\partial L/\partial \theta_j = 0 \quad \Rightarrow \quad v_j \sum_\alpha R^\alpha y_j^\alpha (1 - y_j^\alpha) = 0 \tag{5}$$

$$\partial L/\partial u_{ij} = 0 \quad \Rightarrow \quad v_j \sum_\alpha R^\alpha y_j^\alpha (1 - y_j^\alpha) I_i^\alpha = 0 \tag{6}$$

Based upon these equations, they obtain a complete characterization of the stationary points of the XOR task. They classify the stationary points on the basis of the network outputs, yielding four equivalence classes. Class (a) is the situation where all the network outputs are incorrect. In that case, it is required that $O^1 = O^2 = O^3 = O^4 = \frac{1}{2}$. Blum's line is a subset of this class, which consists of a number of manifolds intersecting at the origin of weight space. Class (a) occurs for finite weight values.

Class (b) is the situation where two of the network outputs are correct. It is characterized byA $O^1 = O^2 = \frac{1}{2}, O^3 = 1 - \delta, O^4 = \delta$ and similar situations with $O^1 \leftrightarrow O^4$ and $O^2 \leftrightarrow O^3$. Classes (c) and (d) occur when exactly one of the network outputs is correct. Class (c) is characA $O^1 = O^2 = O^3 = (2 - \delta)/3, O^4 = \delta$ and the corresponding case with O^1 and O^4 interchanged. Class (d) is characA $O^1 = O^3 = O^4 = (1 + \delta)/3, O^2 = 1 - \delta$ and the corresponding case with O^2 and O^3 interchanged.

An interesting aspect of classes (b)–(d) is that these stationary points only occur when one or more of the network weights are infinite. These stationary points have the property that for each hidden node j either $v_j = 0$ or y_j^α is equal to 0 or 1 wherever $O^\alpha \neq t^\alpha$. The latter case, which must apply to at least one hidden node, implies that some of the hidden node weights are infinite. In such a situation, the definition of local minimum becomes a significant issue. Since infinite weight points do not actually exist, it may not be meaningful to discuss derivatives of error at infinity, or to compare values over some neighborhood of an infinite point. Accordingly, an alternative definition of local minimum is required—a definition which captures the concept of asymptotic minima cleanly. Such a definition is presented in the next section.

4 Definition of Local Minimum

Let $g : \mathcal{D} \rightarrow \mathbb{R}$ be a scalar differentiable function on domain $\mathcal{D}$. The usual definition of a local minimum is that $\mathbf{w}_0$ is a local minimum or *relative minimum* if $g(\mathbf{w}_0) \leq g(\mathbf{w})$ for all $\mathbf{w}$ in a neighborhood of $\mathbf{w}_0$. If $g(\mathbf{w}_0) < g(\mathbf{w})$ for all $\mathbf{w} \neq \mathbf{w}_0$ in the neighborhood then $\mathbf{w}_0$ is said to be a *strict relative minimum* (Luenberger, 1984, p. 168). Unfortunately, these definitions are unsuitable for an analysis of the error surfaces of artificial neural networks, since such surfaces often exhibit asymptotes that approach a minimum error value. For example, the stationary point classes (b)–(d) of Lisboa and Perantonis (1991)

occur as asymptotes. In fact, for feedforward neural networks with sigmoidal or related activation functions for the hidden nodes, the error surface is asymptotically flat for fixed output weights as the hidden node weights approach infinity. It follows that asymptotic relative minima are pervasive and asymptotic strict relative minima do not exist. Thus, neither definition is useful as a basis for analysis. Clearly, an alternative definition of local minimum is needed. The alternative definition used herein is based upon the concept of a trajectory—a one-dimensional path through a high-dimensional space.

Definitions. A *trajectory* A from $\mathbf{w}_1$ to $\mathbf{w}_2$ is a continuous function $A : \mathbb{R} \rightarrow \mathcal{D}$ such that $A(a_1) = \mathbf{w}_1$ and $A(a_2) = \mathbf{w}_2$ where $a_2 > a_1$. We will write $A_{\mathbf{w}_1 \rightarrow \mathbf{w}_2}$ to denote a trajectory from $\mathbf{w}_1$ to $\mathbf{w}_2$. Throughout this paper, all trajectories are required to be piece-wise differentiable. A *non-ascending* trajectory $A_{\mathbf{w}_1 \rightarrow \mathbf{w}_2}$ has the additional property that the function g is non-increasing along the trajectory; i.e. $\partial g(A(a))/\partial a \leq 0$ for $a_1 \leq a \leq a_2$ wherever $g \circ A$ is differentiable (where $\circ$ denotes function composition).

Our alternative definition of local minimum is in terms of regions of the domain that are known to contain one or more minima. The minimum region $M_g(\mathbf{w}_0)$ is defined by the function g and the point $\mathbf{w}_0$ and consists of the (possibly open) connected set of points in the domain $\mathcal{D}$ which are reachable by a non-ascending trajectory from $\mathbf{w}_0$. Under this definition, two local minimum regions $M_g(\mathbf{u})$ and $M_g(\mathbf{v})$ are distinct, and so contain distinct local minima, if they do not intersect; i.e. if $M_g(\mathbf{u}) \cap M_g(\mathbf{v}) = \emptyset$. Applying this definition to the Gaussian function $\Phi(x) = \exp\{-|x|^2\}$ on the domain $\mathbb{R}^1$, two distinct local minima exist in the local minimum regions $M_\Phi(-1)$ and $M_\Phi(1)$, which are distinct. However, the Gaussian function on the domain $\mathbb{R}^n, n > 1$ has only one local minimum, since the minimum region $M_\Phi(\mathbf{x})$ includes all points on or outside the sphere of radius $|\mathbf{x}|$ centred at the origin.

Definition. The *minimum region* $M_g(\mathbf{w}_0)$ is the (possibly open) connected set of points $\mathbf{w}$ for which there exists a non-ascending trajectory $A_{\mathbf{w}_0 \rightarrow \mathbf{w}}$.

Remark. Under the above definition, a point $\mathbf{w}_0$ is said to *represent* a local minimum with value g_0 if and only if $g(\mathbf{w})$ is bounded below by g_0 for all $\mathbf{w} \in M_g(\mathbf{w}_0)$. Further, a local minimum or *regional minimum* with value g_0 is considered to exist if and only if there exists a point $\mathbf{w}_0$ which represents that

minimum.

<<< Insert figure 2 approximately here. >>>

Remark. We have already remarked that the definition of regional minimum is suitable for considering asymptotic minima whereas the definitions of both relative minimum and strict relative minimum are not suitable. However, in the case where finite minima points exist, the three definitions also differ. Figure 2 illustrates these differences. The points at (a) satisfy the requirements to be relative minima but are not minima under the other definitions since the surface is locally a *plateau* which may be escaped. The points at (b) are relative minima and also part of a regional minimum, but they are not strict relative minima since the surface is locally a plateau. The point at (c) is a minimum under all three definitions. It is intuitively appealing to adopt a definition of local minimum which accepts the points at (b) and (c) as local minima, but has the power to identify the fact that points at (a) can be escaped by following a path which is non-ascending in the error function. It is especially useful in considering the local minima in feedforward neural networks where, as discussed in section 7, local plateaus may be escaped by simple algorithms that are incapable of escaping true local minima. It is clear that the set of strict relative minima are a subset of the finite regional minima which are themselves a subset of the relative minima.

Remark. We note that, for functions defined on Euclidean space which are bounded below and everywhere differentiable, regional minima only exist with values corresponding to stationary points of the function. It may readily be seen that asymptotic regional minima correspond to stationary points if the function is bounded below, since the function must approach a limit value as the infinite boundary of the domain is approached by a non-ascending trajectory. Finite regional minima are also seen to correspond to stationary points since a finite regional minimum has a finite point with the minimum value. That point is a relative local minimum and hence a stationary point since the function is everywhere differentiable. This result is relevant to the analysis of feedforward neural networks since the error function is normally defined on a Euclidean weight space, and is bounded below and everywhere differentiable.

5 The XOR Error Surface

Our analysis of the error surface of the XOR network with two hidden nodes proceeds by considering each of the classes (a)–(d) of stationary points presented by Lisboa and Perantonis (1991) and demonstrating non-ascending trajectories that achieve an error less than that of the claimed local minimum. This demonstration proves that none of the stationary points is a regional local minimum. Since Lisboa and Perantonis's analysis completely characterizes the stationary points of the error surface, and since the error surface is differentiable everywhere, it follows that the error surface has no regional minima.

In analysing classes (b)–(d) of Lisboa and Perantonis (1991), the trajectories involve infinite weights. Section 6 provides the theoretical foundation and the limitations for such reasoning. In the following, class (b) is considered last because of its relatively greater complexity.

(a) $O^1 = O^2 = O^3 = O^4 = \frac{1}{2}$.

In this class of stationary points, the output of the network is unaffected by the input. This means that for each hidden node, one of the following applies.

(i) $v_j = 0$. The connecting weight between the hidden node and the output node is zero so that the hidden node's activation is ignored. In this situation, it is clearly possible to continuously vary the hidden node's weights (u_{ij} and θ_j) to any desired values. Once suitable weights have been obtained for both hidden nodes, then v_j can be varied to solve the XOR task without at any point increasing the error.

(ii) $y_j^1 = y_j^2 = y_j^3 = y_j^4$. In this case, the trajectory described by $v_j' = (1-a)v_j$ and $\phi' = \phi + av_j y_j^1$ (where a varies from 0 to 1) can be used to continuously vary v_j to zero without modifying the network output or the error. Case (i) then applies.

(iii) $v_j \neq 0$ and y_j^α not all equal. Since $z^\alpha = \phi + v_j y_j^\alpha + v_k y_k^\alpha = 0$ for all α then the outputs of the other hidden node k must be linearly related to the outputs of hidden node j; i.e. $y_k^\alpha = by_j^\alpha + c$ and $v_k = -v_j/b$ for some b and c. In this case, v_j and v_k can both be continuously varied to 0 while ϕ is simultaneously varied using the trajectory described by $v_j' = (1-a)v_j$ and $v_k' = (1-a)v_k$ and $\phi' = \phi + a\left(v_j\overline{y_j} + v_k\overline{y_k}\right)$ where $\overline{y_l} = \sum_{\alpha=1}^4 y_l^\alpha/4$ and a varies from 0 to 1.

Case (i) then applies.

In the above cases (i)–(iii), then, there is a trajectory which leads to the situation $v_1 = v_2 = 0$ without increasing the error, and from there to a global minimum. Hence, the case $O^1 = O^2 = O^3 = O^4 = \frac{1}{2}$ is not a regional local minimum.

(c) $O^1 = O^2 = O^3 = (2 - \delta)/3, O^4 = \delta$ and the corresponding case with O^1 and O^4 interchanged.

In this class of stationary points, it can be shown from equations 5 and 6 that, for each hidden node j, either $v_j = 0$ or $y_j^1 = y_j^2 = y_j^3 = 0$ or 1. Two cases therefore arise.

(i) $v_j = 0$ and $v_k \neq 0$. Here, hidden node k distinguishes input 4 from the other input patterns (i.e. $y_k^1 = y_k^2 = y_k^3 = 0$ or 1 and $y_k^4 \neq y_k^1$). Hidden node j is ignored, so its weights can be continuously modified to any desired values. In particular, it can be made to distinguish input 1 from the other input patterns (e.g. $y_j^1 > 0$ and $y_j^2 = y_j^3 = y_j^4 = 0$). The weights of the output node can then be continuously modified to solve the XOR task without at any point increasing the error.

(ii) $v_1 \neq 0$ and $v_2 \neq 0$. In this case, it can be shown that $y_1^1 = y_1^2 = y_1^3 = 0$ or 1 and $y_2^1 = y_2^2 = y_2^3 = 0$ or 1. It follows that the output of hidden node 2 is linearly dependent upon the output of hidden node 1. Therefore, the weights of the output node can be continuously modified to achieve $v_1 = 0$ without changing the network output. Case (i) then applies.

In both cases (i) and (ii), then, there is a trajectory which leads to a solution of the XOR task without at any point increasing the error. Hence, the case $O^1 = O^2 = O^3 = (2 - \delta)/3, O^4 = \delta$ is not a regional local minimum.

(d) $O^1 = O^3 = O^4 = (1 + \delta)/3, O^2 = 1 - \delta$ and the corresponding case with O^2 and O^3 interchanged.

Similar reasoning to case (c) applies here.

(b) $O^1 = O^2 = \frac{1}{2}, O^3 = 1 - \delta, O^4 = \delta$ and similar cases with $O^1 \leftrightarrow O^4$ and $O^2 \leftrightarrow O^3$.

In this class of stationary points, it is shown by Lisboa and Perantonis (1991) that $v_j = 0$ or $y_j^1 = y_j^2 = 0$ or 1 for all hidden nodes j. In fact, it can easily be shown that v_1 and v_2 are both

non-zero. The following discussion considers only the case $y_j^1 = y_j^2 = 0$ for all j. Situations where $y_j^1 = y_j^2 = 1$ can be treated in the same manner by first applying a simple transformation of the weight space[1] yielding $y_j^{1\prime} = y_j^{2\prime} = 0$. The following discussion also assumes that $0 < y_j^3 < 1$ and $0 < y_j^4 < 1$, implying that $u_{1j} = -\infty$ while u_{2j} and θ_j are finite. The results are readily extended by continuity to the cases where u_{2j} and/or θ_j are infinite.

In the following derivation, the activation function of the output node f_O is required to be monotonic increasing and invertible and to satisfy $f_O(-z) = 1 - f_O(z)$. These properties hold for the sigmoidal activation function and also for a linear activation function $f_O(z) = z + \frac{1}{2}$. We assume, without loss of generality, that $\delta < \frac{1}{2}$.

Let $z_\delta = f_O^{-1}(\delta)$. Note that since $O^1 = \frac{1}{2}$ and $y_1^1 = y_2^1 = 0$ then $\phi = 0$. Since $O^3 = 1 - \delta$ and $O^4 = \delta$, it follows from the above that $v_1 y_1^3 + v_2 y_2^3 = -z_\delta$ and $v_1 y_1^4 + v_2 y_2^4 = z_\delta$. These equations lead to the following explicit formulations for the weights v_1 and v_2 in terms of the hidden node outputs and z_δ.

$$v_1 = \frac{z_\delta (y_2^3 + y_2^4)}{y_2^3 y_1^4 - y_1^3 y_2^4} \tag{7}$$

$$v_2 = \frac{-z_\delta (y_1^3 + y_1^4)}{y_2^3 y_1^4 - y_1^3 y_2^4} \tag{8}$$

Since $\delta < \frac{1}{2}$ it follows that $z_\delta < 0$ and thus that $v_1 < 0$ if and only if the discriminant $d = y_2^3 y_1^4 - y_1^3 y_2^4$ is strictly positive. (The situation $d = 0$ cannot arise since there is then no solution for the weights v_1 and v_2.) We consider only the case $d > 0$. The case $d < 0$ is similar, except that the roles of the two hidden nodes are exchanged.

In the case $d > 0$, note that reducing y_2^4 while maintaining y_2^3 constant increases d. In particular, y_2^4 can be reduced to zero. A trajectory that achieves this change involves reducing θ_2 and increasing u_{22}. Such a trajectory cannot be described in the weight space parameterization without

[1] When $y_j^1 = y_j^2 = 1$ then let $\theta_j' = -\theta_j$ and $u_{ij}' = -u_{ij}$ so that $y_j^{\alpha\prime} = 1 - y_j^\alpha$ and let $v_j' = -v_j$ and $\phi' = \phi + v_j$. This is a homeomorphism as required by the transformation lemma of section 6.

introducing conflicting infinite values. Adopting the reparameterization of example 1 in section 6.1 allows us to describe the trajectory without conflicting infinite values. To reduce y_2^4 to zero, x_2^4 is reduced from its original value to $-\infty$, while x_2^3 and x_2^1 are held constant. The relation $x_2^2 = x_2^1 + x_2^4 - x_2^3$ does not modify x_2^2 since $x_2^2 = -\infty$ already. Thus, y_2^4 can be continuously reduced to zero while simultaneously continuously adjusting v_1 and v_2 to maintain the network outputs O^1, O^2, O^3 and O^4 unchanged.

When $y_2^4 = 0$ then y_1^3 and y_1^1 can be increased without reducing d. Again, the reparameterization of example 1 in section 6.1 is adopted. The trajectory first proceeds by increasing x_1^3 to ∞ while maintaining x_1^4 and x_1^1 constant. The relation $x_1^2 = x_1^1 + x_1^4 - x_1^3$ does not modify x_1^2 since $x_1^2 = -\infty$ already. When $x_1^3 = \infty$ then x_1^1 is modified from its initial value of $-\infty$ to a finite value so that $y_1^1 > 0$. Both x_1^3 and x_1^4 are maintained constant and, as before, x_1^2 is unchanged. When $y_2^4 = 0$ and $y_1^1 > 0$, equation 7 simplifies to $v_1 = z_\delta / y_1^4$ and it follows that $z^1 = z_\delta y_1^1 / y_1^4 < 0$ which means that $O^1 < \frac{1}{2}$ and the network error has been reduced below that of the stationary point.

Thus, there is a trajectory which leads to reduced error without at any point increasing the error and the case $O^1 = O^2 = \frac{1}{2}, O^3 = 1 - \delta, O^4 = \delta$ is not a local minimum.

The cases (a)–(d) above are a complete catalogue of suboptimal stationary points of the XOR task with two hidden nodes. Each case has been proved to not represent a local minimum, leading to the conclusion that the XOR task has no local minima. The next section develops this result in a more rigorous manner.

It should be noted that the above reasoning is not directly applicable in the case when $\delta = 0$ since, in that case, the network can never achieve correct output values with finite weights. This is not a serious difficulty. Choose $\delta' = \inf_\alpha O^\alpha(1 - O^\alpha)$ which must be finite for all finite weight configurations. The escape from the stationary point is then achieved by following the trajectories described above, with δ' substituted for δ throughout. Observe that in this situation, all O^α are restricted to the range $[\delta', 1 - \delta']$ and that throughout the trajectories, each O^α only ever moves toward its respective target. Because of these conditions, the trajectories above, which are non-ascending with respect to target values of δ' and $1 - \delta'$, will also be non-ascending with respect to target values of δ and $1 - \delta$. Thus, there are no local

minima for the XOR task with $\delta = 0$.

6 Reasoning with Infinite Weights

In section 5, we demonstrated non-ascending trajectories from the stationary points of the XOR task to points in weight space with reduced error. However, classes (b)–(d) of stationary points only occur as asymptotes with at least one of the network weights approaching an infinite value. The escape trajectories for these cases therefore implicitly involve reasoning with infinite weights, requiring the rigorous theoretical foundations that will now be developed. We will show that the demonstration of escape trajectories involving infinite weights proves the existence of escape trajectories for all finite weight points. These escape trajectories contain only finite weights. Since, in practice, infinite weights are not achieved during network training, it is the finite weight behaviour that is of primary interest.

The necessary foundations are provided by the transformation lemma, extension by continuity and the approximation lemma. The transformation lemma provides a basis for transforming the weight space into an alternative parameterization which allows the desired trajectory to be demonstrated clearly, as in the analysis of case (b) for the XOR problem in the previous section. The weight space, possibly transformed, is extended by continuity to include infinite values for the purpose of trajectory analysis. Application of the approximation lemma to trajectories obtained in the extended domain can be used to prove the existence of suitable trajectories that avoid the boundary of the transformed domain and correspond, by the transformation lemma, to trajectories in the original weight space. Thus, the previous section has shown that finite escape trajectories exist for all finite weight points, leading to the conclusion that there are no points which represent local minima in the XOR task and hence that local minima do not exist.

6.1 Transformations of Weight Space

The transformation lemma allows one to reason about minima of a function g defined on domain $\mathcal{D}$ by considering the minima of another function g' defined on $\mathcal{D}'$. It requires a homeomorphic transformation T between the domains, and that $g = g' \circ T$.

Lemma 1 *(Transformation Lemma).* Let $T : \mathcal{D} \rightarrow \mathcal{D}'$ be a homeomorphism between two domains. Then $g : \mathcal{D} \rightarrow \mathbb{R}$ has a local minimum represented by $\mathbf{w}_0$ if and only if $g' : \mathcal{D}' \rightarrow \mathbb{R}$ has a local minimum represented by $\mathbf{w}_0{}'$ where $g'(T(\mathbf{w})) = g(\mathbf{w})$ and $\mathbf{w}_0{}' = T(\mathbf{w}_0)$.

Proof. The proof derives from the fact that the transformation T is topology preserving. It follows that there is a one-to-one correspondence between trajectories $A : \mathbb{R} \rightarrow \mathcal{D}$ and their images $A' : \mathbb{R} \rightarrow \mathcal{D}'$ and also a one-to-one correspondence between minimum regions $M_g(\mathbf{w})$ and their images $M_{g'}(\mathbf{w}')$. □

Example 1 *Reparameterization.* Consider hidden unit j with weights u_{1j}, u_{2j} and θ_j. It is useful to consider an alternative parameterization such as $\psi_{ij} = x_j^i$ for $i = 1, 3, 4$. Under this parameterization, $x_j^2 = \psi_{1j} + \psi_{4j} - \psi_{3j}$.

The transformation that achieves this parameterization is defined by

$$\begin{aligned} \psi_{1j} &= u_{1j} + u_{2j} + \theta_j \\ \psi_{3j} &= u_{2j} + \theta_j \\ \psi_{4j} &= \theta_j \end{aligned}$$

This is a homeomorphism, so the conditions of the transformation lemma are satisfied and the local minima obtained by reasoning in the transformed space are equivalent to the local minima of the original weight space.

Example 2 *Transforming infinite weights to finite points.* Consider the transformation $T(\psi_l) = 1/(1 + \exp\{-\psi_l\})$, applied to all the weights of the network. This transformation is a homeomorphism from $\mathbb{R}^n$ to the open unit hypercube $0 < T(\psi_l) < 1$. Thus, T provides a new domain in which infinite weight points become finite. The open unit hypercube may then be extended by continuity to consider infinite weights in the original weight space, without actually considering infinite values.

6.2 Extension by Continuity

For a function defined on an open domain, extension by continuity allows the domain to be partially closed. The value of the extended function at a boundary point is defined as the limit of the original

function, wherever that limit exists.

Example 3 *Extension of transformed weight space.* Consider a hidden unit j. It is desired to consider weight configurations in which the hidden unit output y_j^α is exactly 0 or exactly 1 for all of two or three α. (Stationary points of class (b) have two y_j^α equal to 0 or 1, and stationary points of classes (c) and (d) have three y_j^α equal to 0 or 1.) Suppose that y_j^1, y_j^2 and y_j^4 may be extreme. Then, applying the reparameterization of example 1, $y_j^1 = y_j^2 = 1$ if $\psi_{1j} = \infty$, while $y_j^2 = y_j^4 = 1$ if $\psi_{4j} = \infty$. Applying the non-linear transformation of example 2 remaps the infinite parameters to finite values so that $y_j^2 = y_j^4 = 1$ when $\psi'_{2j} = 1$.

The extended function is defined on all boundary points for which the limit of g exists. In the present example, ψ'_{1j} and ψ'_{4j} may take any values in the square $[0,1] \times [0,1]$ except the points $(0,1)$ and $(1,0)$.[2] This allows two or three of y_j^1, y_j^2 and y_j^4 to be all exactly 0 or 1.

6.3 Approximation Lemma

The approximation lemma says that, if there exists a non-ascending trajectory $A_{\mathbf{a}\to\mathbf{b}}$ which passes through $\mathbf{d}$ then there exist other non-ascending trajectories from $\mathbf{a}$ to other points $\mathbf{q}$ where $g(\mathbf{q}) = g(\mathbf{d})$. The alternate trajectories do not include the point $\mathbf{d}$ or any points beyond it in the original trajectory. This important lemma allows us to consider trajectories on an extended domain $\mathcal{E}$ and infer the existence of suitable trajectories on the original domain $\mathcal{D}$. In particular, it allows us to consider trajectories involving infinite weights and to infer the existence of trajectories involving only finite weights.

Lemma 2 *(Approximation Lemma).* Let $A_{\mathbf{a}\to\mathbf{b}}$ be a non-ascending trajectory which passes through $\mathbf{d}$ (figure 3). Let $B_{\mathbf{a}\to\mathbf{b}}$ be an arbitrary smooth trajectory which does not have any points other than $\mathbf{a}$ and $\mathbf{b}$ in common with A. The approximation lemma guarantees the existence of both a point $\mathbf{q}$ where $g(\mathbf{q}) = g(\mathbf{d})$ and a non-ascending trajectory $A^*_{\mathbf{a}\to\mathbf{q}}$. The trajectory A^* has no points in common with the subtrajectory $A_{\mathbf{d}\to\mathbf{b}}$ except possibly the end point $\mathbf{b}$ (and then only if $g(\mathbf{b}) = g(\mathbf{d})$).

<<< Insert figure 3 approximately here. >>>

[2] If $\psi'_{1j} = 1$ and $\psi'_{4j} = 0$ then $x_j^1 = \infty$, $x_j^4 = -\infty$ and $x_j^2 = x_j^1 + x_j^4 - x_j^3$ has conflicting infinities.

Proof. The proof proceeds by constructing a level trajectory C from a point $\mathbf{c}$ on trajectory A to a point $\mathbf{p}$ on B. We choose $\mathbf{c}$ such that $g(\mathbf{c}) > g(\mathbf{d})$ but ensure that $g(\mathbf{c})$ is small enough that there exists a portion of the trajectory B which is non-ascending and leads from $\mathbf{p}$ to a point $\mathbf{q}$ where $g(\mathbf{q}) = g(\mathbf{d})$. This is possible provided that the set of values of local minima along B is finite.

Let S be a two-dimensional manifold bounded by the trajectories A and B that is a homeomorphic image of the disk. We require S to be piece-wise smooth. Let G be the set of values of regional minima of g along the trajectory B and let G^+ be the subset of G that are greater than $g(\mathbf{d})$. We require G^+ to be finite.[3] Let g_1 be the minimum value of the set G^+. Clearly, such a minimum exists and $g_1 > g(\mathbf{d})$.

Choose g^* such that $g(\mathbf{d}) < g^* < g_1$ and $g^* < g(\mathbf{a})$.[4] Define S^+ as the connected set of points $\mathbf{w} \in S$ where $g(\mathbf{w}) > g^*$ and where $\mathbf{a} \in S^+$. Since g is continuous over S, the boundary Q^+ of S^+ in S consists of portions of the trajectories A and B and portions of the level set $S^* = \{\mathbf{w} : \mathbf{w} \in S; g(\mathbf{w}) = g^*\}$. Since A is non-ascending, the portion of A included in Q^+ is only the subtrajectory $A_{\mathbf{a}\to\mathbf{c}}$ for some $\mathbf{c}$ where $g(\mathbf{c}) = g^*$. Note that $g(\mathbf{w}) \geq g^*$ for all $\mathbf{w} \in A_{\mathbf{a}\to\mathbf{c}}$ and $g(\mathbf{w}) \leq g^*$ for all $\mathbf{w} \in A_{\mathbf{c}\to\mathbf{b}}$.

Now, Q^+ is a closed boundary, so from $\mathbf{c}$ there exists a portion $Q^+(\mathbf{c}, \mathbf{p})$ of the boundary which connects from A at $\mathbf{c}$ to B at some point $\mathbf{p}$.[5] It follows that there exists a level trajectory $C \subseteq S^*$ which connects $\mathbf{c}$ to $\mathbf{p}$. In particular, we may take $C \subseteq Q^+(\mathbf{c}, \mathbf{p})$, eliminating any loops or spikes in $Q^+(\mathbf{c}, \mathbf{p})$ such as might arise from saddle points or valleys in g over S. From $\mathbf{p}$ it is possible to follow a non-ascending portion of B to a nearby minimum $\mathbf{r}$. Note that the value of that minimum $g(\mathbf{r})$ is included in the set G. Since $g(\mathbf{p}) = g^* < g_1$, and since g_1 is the minimum of the set G^+ it follows that $g(\mathbf{r}) \notin G^+$ and hence that $g(\mathbf{r}) \leq g(\mathbf{d})$. It follows that there exists a point $\mathbf{q}$ on the non-ascending sub-trajectory B between $\mathbf{p}$ and $\mathbf{r}$ where $g(\mathbf{q}) = g(\mathbf{d})$.

The non-ascending trajectory $A^*_{\mathbf{a}\to\mathbf{q}}$ consists of three sub-trajectories: follow A from $\mathbf{a}$ to $\mathbf{c}$, then follow C to $\mathbf{p}$ and finally follow B to $\mathbf{q}$. □

[3] In the context of feedforward neural network training, B can be taken as linear. A finite number of hidden units and a finite number of training patterns is then sufficient to ensure that G^+ is finite. This holds because the error surface for a single pattern has a finite number of local minima along any straight line, and the overall error surface is the sum of a finite number of such surfaces.

[4] Such g^* always exists except in the degenerate case where $g(\mathbf{a}) = g(\mathbf{d})$; i.e. $A_{\mathbf{a}\to\mathbf{d}}$ is flat. In that case the trivial trajectory $A^*_{\mathbf{a}\to\mathbf{a}}$ satisfies the lemma. In application to the study of the XOR network, this degenerate case does not arise since the point $\mathbf{a}$ is a finite approximation to an infinite stationary point and $\mathbf{a}$ itself is not a stationary point.

[5] From $\mathbf{p}$, the boundary Q^+ consists of portions of B and possibly S^* returning to $\mathbf{a}$.

6.4 Application

In order to characterise the regional minima of the XOR error surface, it is only necessary to consider the stationary points as noted in section 4.

In considering the stationary points of the XOR task in section 5, we developed non-ascending trajectories from the stationary points to points with reduced error. For stationary points of class (a), which are finite, the trajectories involve only finite weight configurations, proving that this class of stationary points do not represent regional local minima.

For stationary points of classes (b)–(d), however, both the stationary points and the trajectories involve infinite weights. As suggested previously, infinite weight points are not achieved in practice and the consideration of infinities requires care. What is actually required is a proof that, for any finite point hypothesized to represent an asymptotic minimum, there exists a non-ascending trajectory of finite points leading to a point with error less than that of the claimed asymptotic minimum. Then, under the definition of regional local minimum, there is no finite point representing the claimed asymptotic minimum and so there is no minimum. The necessary proof is incorporated into the following theorem.

Theorem 1. The XOR task with two hidden nodes has no regional local minima.

Proof. As noted earlier, it is only necessary to consider the stationary points of the XOR task which have been characterized by Lisboa and Perantonis (1991) since there are no other potential regional minima.

For class (a) stationary points, which are finite, we have already shown the existence of non-ascending trajectories to points of reduced error which involve only finite weight configurations. This is sufficient to prove that the class (a) stationary points do not correspond to regional local minima.

Classes (b)–(d) of stationary points involve infinite weights. However, we have already shown in examples 1, 2 and 3, how the transformation lemma and extension by continuity can be applied to obtain finite weight points representing these stationary points in a transformed space; i.e. the trajectories presented in section 5 correspond to finite trajectories in the extended transformed domain $\mathcal{D}'$. For classes (c)–(d), the original weight space is transformed in a manner such as that given in example 2 to a finite space which is then extended by continuity as in example 3. For class (b), the transformation of

example 1 is first used to eliminate potential conflicts of infinite weights, and then the transformation of example 2 and extension by continuity lead to a finite transformed domain.

Now, as noted above, neither the stationary points nor the trajectories exist in the original (finite) weight space $\mathcal{D}$. Consider, therefore, a finite point $\mathbf{w}_1 \in \mathcal{D}$ which is hypothesized to represent an asymptotic local minimum of error with value g_0. If $\mathbf{w}_1$ does, in fact, represent an asymptotic local minimum then, by the definition of regional minimum, there exists a non-ascending trajectory from $\mathbf{w}_1$ in $\mathcal{D}$ which asymptotically approaches the hypothesized local minimum. Then there exists a non-ascending trajectory[6] in $\mathcal{D}'$ from $\mathbf{w}'_1$ to the finite representation of the claimed local minimum $\mathbf{w}_0$. From that point, we have already demonstrated in section 5 the existence of a trajectory in $\mathcal{D}'$ to a point with error less than g_0. Thus, from $\mathbf{w}'_1$ there exists a non-ascending trajectory, which we will call A, to a point $\mathbf{w}'_2$ where $g(\mathbf{w}'_2) < g_0$. The trajectory A first touches the boundary of $\mathcal{D}'$ at $\mathbf{w}_0$.

To apply the approximation lemma, choose B a smooth trajectory from $\mathbf{w}'_1$ to $\mathbf{w}'_2$ and choose a suitable manifold[7] S in $\mathcal{D}'$. It is required that B does not include any boundary points of $\mathcal{D}'$ except possibly $\mathbf{w}'_2$. With the transformations we are using, $\mathcal{D}'$ is convex and a linear trajectory satisfies for B. The approximation lemma then implies the existence of another trajectory $A^{*\prime}$ from $\mathbf{w}'_1$ to $\mathbf{w}'_0$ where $\mathbf{w}'_0$ is not on the boundary of $\mathcal{D}'$. Further, since the trajectory A only touches the boundary at points where $g(\mathbf{w}) \leq g_0$, and since none of the points in S are in the boundary of $\mathcal{D}'$ except for points in A, the trajectory $A^{*\prime}$ consists purely of non-boundary points of $\mathcal{D}'$.

By the transformation lemma, $A^{*\prime}$ is the image of a trajectory A^* in $\mathcal{D}$, and A^* consists only of finite points. A^* is thus a non-ascending trajectory from $\mathbf{w}_1$ to $\mathbf{w}_0$ where $g(\mathbf{w}_0) = g_0$. However, since g_0 is the value of a class (b)–(d) stationary point, it is less than the error of a class (a) stationary point. Since there are no finite stationary points with error less than that of the class (a) stationary points, $\mathbf{w}_0$ is not a stationary point. The derivative of the error surface at $\mathbf{w}_0$ is therefore non-zero and the error may be reduced by following the gradient, either to the global minimum or to a lower valued stationary point. Thus, there exists a trajectory of finite points from $\mathbf{w}_1$ to a point with error less than g_0 and

[6] We require that the transformed trajectory uniformly approach a limit point on the boundary of the transformed domain. Such a trajectory can be found because the number of hidden units and training patterns are both finite and thus the number of local minima along any straight line is finite. The asymptotic behaviour of the trajectory in $\mathcal{D}$ is therefore a straight line which maps to a single point on the boundary of $\mathcal{D}'$.

[7] The manifold S is required to not include any boundary points of $\mathcal{D}'$ except those boundary points contained in A.

hence $\mathbf{w}_1$ does not represent a local minimum with value g_0.

Since this reasoning applies for all $\mathbf{w}_1$ and for all classes (b)–(d) of stationary points, there are no points that represent local minima in stationary point classes (b)–(d) and hence, regional local minima of classes (b)–(d) do not exist. Since points of class (a) have already been shown to not be regional local minima, it follows that the XOR task with two hidden nodes has no regional local minima. □

A direct consequence of this theorem is that, for every finite weight configuration of the XOR task with two hidden nodes, there exists a non-ascending trajectory to a global minimum of error.

The above result for networks with two hidden nodes readily extends to the case of networks with short-cut (direct bottom-to-top) connections and one or more hidden nodes since, as shown by Lisboa and Perantonis (1991), networks with short-cut connections only have stationary points of class (a). The description of non-ascending solution trajectories from such stationary points is not difficult. The particular case of one hidden node with short-cut connections has recently been analysed showing that all stationary points are saddle points (Sprinkhuizen-Kuyper & Boers, 1996). The finite stationary points (class (a)) for the architecture of figure 1 have similarly been analysed (Hamey, 1996) and shown to be saddle points. In the case of networks without short-cut connections and with three or more hidden nodes, the results of Poston et al. (1991) and Yu and Chen (1995) apply (see also Hamey, 1994), proving that no local minima exist and that all stationary points are saddle points. These considerations lead to the result that the XOR task has no local minima.

7 Discussion

Having proved above that the XOR task has no local minima, we now consider the implications of this result in the wider area of neural network research.

Perhaps the most obvious implication of the present result is that local minima are not as common as once thought. The strong evidence of entrapment in learning the XOR task has previously been taken as evidence of a local minimum, but this judgement has now been shown to be incorrect. Hush, Horne, and Salas (1992) observe that "there are many cases where there is no minimum in a given direction, just a flat spot that extends to infinity" (p. 1157). In a tutorial presentation at Sydney University,

Australia in 1993, G. Hinton observed that in many situations where people believe that their networks are in local minima, they have actually encountered a plateau. The present result shows that the XOR problem with two hidden units, long thought to exhibit local minima, is an example of this phenomenon, the gradient descent algorithm being drawn towards an infinite plateau.

It has been observed (Baldi & Hornik, 1989; Poston et al., 1991; Rumelhart et al., 1986) that back propagation and related gradient descent learning algorithms can become trapped at points that are not local minima. Stationary points of any type, even saddle points, may entrap a gradient descent algorithm. It was to avoid such an entrapment point, where two or more hidden units are performing the same computation and can never learn to perform separate tasks, that Rumelhart et al. (1986) introduced random starting weights. When the entrapment points are not local minima, escaping and avoiding entrapment does not require temporarily increasing the network error and is therefore easier to achieve.

Alternative optimization techniques such as conjugate gradient methods (Johansson, Dowla, & Goodman, 1992; Kinsella, 1992; Kramer & Sangiovanni-Vincentelli, 1989; Møller, 1993) may, depending upon the line search technique used, be capable of traversing entrapment regions such as plateaus and saddles (but see the warning concerning line searches in Hush et al., 1992). Simulated annealing (Kirkpatrick, Gelatt, & Vecchi, 1983) is designed to escape local minima, but escaping entrapment that is not due to a local minimum is simpler since there is no need for any particular annealing schedule—even a very low temperature will eventually escape. Similarly, other learning algorithms such as stochastic gradient descent (Darken & Moody, 1991, 1992) and tunneling (Cetin, Barhen, & Burdick, 1993; Cetin, Burdick, & Barhen, 1993) can escape entrapment that is not caused by local minima without the burden of careful parameter selection. Even back propagation may successfully learn in the presence of some forms of entrapment since the finite steps involved can traverse them. Plateaus of large extent can be traversed slowly if the learning algorithm oscillates through the manifold of the plateau and if the error derivatives are not anti-parallel on opposite sides of the plateau manifold. In this case, a large value of momentum (Rumelhart et al., 1986) would be a disadvantage to learning because it would dampen the oscillations. An example of this oscillatory behaviour may be found in Hush et al. (1992), although in their example the oscillation leads the search away from the global minimum.

Learning of the XOR task has been observed to rarely become trapped in practice (Rumelhart et al., 1986) although the probability of entrapment varies depending upon the initial random weight range. The high rate of successful learning, particularly when the initial random weight range is suitably small (Kolen & Pollack, 1990), shows that back-propagation learning is capable of escaping or avoiding the finite (class (a)) stationary points in the error surface of the XOR task, which have been shown to be saddle points (Hamey, 1996; Lisboa & Perantonis, 1991; Sprinkhuizen-Kuyper & Boers, 1996). In simulation experiments, we have observed back-propagation learning appearing to become trapped in stationary points of classes (b)–(d), but escaping after sufficient epochs of training. In many cases, however, back-propagation fails to escape even after a million training epochs, probably indicating that the gradient descent algorithm is drawn toward the asymptotic stationary point rather than following an escape trajectory.

Other researchers (Cetin, Burdick, & Barhen, 1993; Lisboa & Perantonis, 1991) have found that the Hessian matrix is positive definite for some finite weight points approaching stationary points of classes (b)–(d). The present work shows that it is incorrect to interpret this result as proof that these stationary points are strict relative minima. We conjecture that these stationary points resemble a river valley descending to the coast. At any point in the valley, the valley rises around you and the floor of the valley curves up (the Hessian matrix is positive definite) yet the banks on either side also become lower as the river descends into the coastal plain. It is possible, then, to follow the contour of the land and escape the river valley without traveling up hill. Another possible explanation is that the positive definite Hessian matrix arises from numerical imprecision in the location of points in weight space; that the point where the matrix is computed is near a point where the Hessian is singular.

Our correction of the analysis by Blum (1989) reveals a problem with the visualization of the error surfaces of neural networks. Because the error surface exists in a high-dimensional space, linear slices (Gori & Tesi, 1992) and two-dimensional planar slices have been used for visualization (Brady et al., 1989; Hush et al., 1992). This technique does not always reveal the relevant features of the error surface. Our discussion of Blum's line of stationary points showed the existence of reduced error in the neighborhood of each point on the line. This behaviour can be visualized by plotting the error surface over a two-dimensional manifold of the weight space (Hamey, 1995). The required manifold

is non-planar, however, having the equation $\phi' = (1 - a/2)\phi$. The existence of reduced error near all points of Blum's line is not evident in *any* planar slice through the weight space. This example serves to illustrate that the practice of visualizing planar slices of the weight space may be misleading. A careful analysis of the full weight space is required to obtain an accurate understanding of its properties.

It is clear from the above discussion that the distinction between local minima and other types of stationary point is of great significance for learning algorithms for feedforward neural networks. Unfortunately, this distinction has at times become blurred, as in (Gori & Tesi, 1990, 1992; Wessels & Barnard, 1992) which are concerned with back-propagation becoming trapped during training (also an issue of great significance) rather than the issue of "local minimum" in a strict sense. Such confusion hides the potential to develop alternative algorithms which, though subject to entrapment in true local minima, are capable of avoiding or escaping other forms of entrapment. An accurate understanding of the entrapment of the XOR task will assist in the development of such algorithms, with benefits in a wide range of neural network applications.

The mathematical framework developed in this paper is general and may be applied to any feedforward neural network learning task. In particular, the transformation lemma formalizes the use of alternate representations for weight space which do not change the essential nature of the optimization task. Homeomorphic transformations of the input data, such as translation and scaling, are also covered by a simple extension of the transformation lemma. Such transformations may have practical benefits, however, as they modify the error gradient vector. Appropriate transformations may serve a function similar to regularizers (Chauvin, 1989, 1990; Hanson & Pratt, 1989; Weigend, Huberman, & Rumelhart, 1990), favoring the development of less extreme or more diverse network weights. However, unlike regularizers which add an extra term to the error surface and so may introduce local minima, homeomorphic transformations of weight space preserve the local and global minima of the learning task.

Finally, we note that during the revision of this paper, additional analytical work on the XOR problem has appeared. The error in the analysis of Blum (1989) has been independently identified (Sprinkhuizen-Kuyper & Boers, 1994). As mentioned previously, the XOR network with one hidden node has been analysed and found to exhibit only saddle points as stationary points (Sprinkhuizen-Kuyper & Boers, 1996) and a similar results has been obtained for the finite stationary points (class (a) above) of the XOR

network with two hidden nodes (Hamey, 1996). None of this work, however, addresses the significant issue of the appropriate definitions of local minimum for analysing the error surfaces of feedforward neural networks. The present work stands alone in suggesting a new definition of local minimum and an analysis method which distinguishes between extended flat regions of the error surface (plateaus) and local minima.

8 Conclusion

We have presented a new definition of local minimum called regional local minimum. We have shown that this definition is suitable for the analysis of the error surfaces of feedforward neural networks, especially the asymptotic stationary points where existing definitions fail to provide a useful basis for analysis. An additional benefit of this new definition is that it distinguishes escapable plateaus from local minima in the error surface, revealing the value of learning algorithms that are capable of traversing plateaus.

Consideration of trajectories through weight space led to an analysis method for hypothesized regional local minima based upon analysing trajectories through weight space. The analysis is facilitated by the transformation lemma which allows reparameterization of the neural network and by the approximation lemma which proves the existence of trajectories of finite weights given that suitable trajectories with infinite weights have been demonstrated.

Application of the new analysis technique to the XOR problem with two hidden nodes has shown the absence of regional local minima. In the process, we have corrected the analysis by Blum (1989) of the XOR problem and extended existing analysis to reveal more of the structure of the error surface of this problem. In view of the placement of the XOR problem as one of the simplest problems not covered by existing results on the existence of local minima, our result represents a significant data point for theorems concerning the existence of local minima in feedforward neural networks. In particular, our result shows that existing theories do not represent the lower bound on the non-existence of local minima. The observation by Rumelhart et al. (1986) that local minima are "very rare" (p. 352) still presents the tantalizing possibility that local minima may not exist in a large class of practical problems.

The present work has also contributed to the understanding of entrapment and the structure of neural network error surfaces, especially with regard to asymptotic stationary points. We have demonstrated that, in the case of the XOR problem with two hidden nodes, trajectories exist by which a learning algorithm may escape entrapment in the asymptotic stationary points without increasing the error temporarily. Further study is required to determine whether this result is applicable to a wider range of neural network problems and architectures. The development of learning algorithms which avoid entrapment in asymptotic stationary points is another opportunity for further research.

Our study has been restricted to the consideration of finite weight behaviour, as infinite weights are normally not possible in existing learning algorithms. The consideration of learning algorithms in which infinite weights are effectively realised, and the implications of the present work for such algorithms, is another interesting avenue for further research and an opportunity for practical application of the transformation lemma.

References

Auer, P., Herbster, M., & Warmuth, M. K. (1996). *Exponentially many local minima for single neurons* (NeuroCOLT Technical Report No. NC-TR-96-030). Egham, Surrey, England: Royal Holloway, University of London, Department of Computer Science.

Baldi, P., & Hornik, K. (1989). Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, *2*, 53–58.

Blum, E. K. (1989). Approximation of boolean functions by sigmoidal networks: Part I: XOR and other two-variable functions. *Neural Computation*, *1*, 532–540.

Brady, M. L., Raghavan, R., & Slawny, J. (1989). Back propagation fails to separate where perceptrons succeed. *IEEE Transactions on Circuits and Systems*, *36*, 665–674.

Cetin, B. C., Barhen, J., & Burdick, J. W. (1993). Terminal repeller unconstrained subenergy tunneling (TRUST) for fast global optimization. *Journal of Optimization Theory and Applications*, *77*, 97–126.

Cetin, B. C., Burdick, J. W., & Barhen, J. (1993). Global descent replaces gradient descent to avoid local minima problem in learning with artificial neural networks. In *Proceedings of the IEEE international conference on neural networks* (Vol. 2, pp. 836–842). Piscataway, NJ: IEEE.

Chauvin, Y. (1989). A back-propagation algorithm with optimal use of hidden units. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 1* (pp. 519–526). San Mateo, CA: Morgan Kaufmann.

Chauvin, Y. (1990). Dynamic behaviour of constrained back-propagation networks. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 2* (pp. 642–649). San Mateo, CA: Morgan Kaufmann.

Darken, C., & Moody, J. (1991). Note on learning rate schedules for stochastic optimization. In R. P. Lippmann, J. E. Moody, & D. S. Touretzky (Eds.), *Advances in neural information processing systems 3* (pp. 832–838). San Mateo, CA: Morgan Kaufmann.

Darken, C., & Moody, J. (1992). Towards faster stochastic gradient search. In J. E. Moody, S. J. Hanson, & R. P. Lippmann (Eds.), *Advances in neural information processing systems 4* (pp. 1009–1016). San Mateo, CA: Morgan Kaufmann.

Dayhoff, J. E. (1990). The exclusive-or: A classic problem. In *Neural network architectures: an introduction* (pp. 76–79). New York: Van Nostrand Reinhold.

Gori, M., & Tesi, A. (1990). Some examples of local minima during learning with back-propagation. In *Parallel architectures and neural networks: Third Italian workshop* (pp. 87–94). Singapore: World Scientific.

Gori, M., & Tesi, A. (1992). On the problem of local minima in backpropagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*, 76–85.

Hamey, L. G. C. (1994). Comments on "can backpropagation error surface not have local minima". *IEEE Transactions on Neural Networks*, *5*, 844.

Hamey, L. G. C. (1995). The structure of neural network error surfaces. In M. Charles & C. Latimer (Eds.), *Proceedings of the sixth Australian conference on neural networks* (pp. 197–200). Sydney, Australia: University of Sydney, Dept. of Electrical Engineering.

Hamey, L. G. C. (1996). Analysis of the error surface of the XOR network with two hidden nodes. In P. Bartlett, A. Burkitt, & R. C. Williamson (Eds.), *Proceedings of the seventh Australian conference on neural networks* (pp. 179–183). Canberra, Australia: The Australian National University.

Hanson, S. J., & Pratt, L. Y. (1989). Comparing biases for minimal network construction with back-propagation. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 1* (p. 177-185). San Mateo, CA: Morgan Kaufmann.

Hinton, G. E. (1989). Connectionist learning procedures. *Artificial Intelligence*, *40*, 185–234.

Hush, D. R., Horne, B., & Salas, J. M. (1992). Error surfaces for multilayer perceptrons. *IEEE Transactions on Systems, Man and Cybernetics*, *22*, 1152–1161.

Johansson, E. M., Dowla, F. U., & Goodman, D. M. (1992). Backpropagation learning for multilayer feed-forward neural networks using the conjugate gradient method. *International Journal of Neural Systems*, *2*, 291–301.

Kinsella, J. A. (1992). Comparison and evaluation of variants of the conjugate gradient method for efficient learning in feed-forward neural networks with backward error propagation. *Network: Computation in Neural Systems*, *3*, 27–35.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*, 671–680.

Kolen, J. F., & Pollack, J. B. (1990). Back propagation is sensitive to initial conditions. *Complex Systems*, *4*, 269–280.

Kramer, A. H., & Sangiovanni-Vincentelli, A. (1989). Efficient parallel learning algorithms for neural networks. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 1* (pp. 40–48). San Mateo, CA: Morgan Kaufmann.

Lisboa, P. J. G., & Perantonis, S. J. (1991). Complete solution of the local minima in the XOR problem. *Network: Computation in Neural Systems*, *2*, 119–124.

Luenberger, D. G. (1984). *Linear and nonlinear programming.* Reading, MA: Addison-Wesley.

Møller, M. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, *6*, 525–533.

Orchard, G. A., & Phillips, W. A. (1991). *Neural computation: A beginner's guide.* London: Lawrence Erlbaum Associates.

Poston, T., Lee, C.-N., Choie, Y., & Kwon, Y. (1991). Local minima and back propagation. In *International joint conference on neural networks* (Vol. 2, pp. 173–176). New York: IEEE.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representations by error propagation. In *Parallel distributed processing* (pp. 318–362). Cambridge, MA: MIT Press.

Sontag, E. D., & Sussmann, H. J. (1989). Backpropagation can give rise to spurious local minima even for networks without hidden layers. *Complex Systems*, *3*, 91–106.

Sontag, E. D., & Sussmann, H. J. (1991). Back propagation separates where perceptrons do. *Neural Networks*, *4*, 243–249.

Sprinkhuizen-Kuyper, I. G., & Boers, E. J. W. (1994). *A comment on a paper of Blum: Blum's "local minima" are saddle points* (Tech. Rep. No. 94-34). Leiden, The Netherlands: Leiden University, Dept. of Computer Science.

Sprinkhuizen-Kuyper, I. G., & Boers, E. J. W. (1996). The error surface of the simplest XOR network has only global minima. *Neural Computation*, *8*, 1301–1320.

Wasserman, P. D. (1989). *Neural computing: Theory and practice.* New York: Van Nostrand Reinhold.

Weigend, A. S., Huberman, B. A., & Rumelhart, D. E. (1990). Predicting the future: A connectionist approach. *International Journal of Neural Systems*, *1*, 193–209.

Wessels, L. F. A., & Barnard, E. (1992). Avoiding false local minima by proper initialization of connections. *IEEE Transactions on Neural Networks*, *3*, 899–905.

Yu, X.-H., & Chen, G.-A. (1995). On the local minima free condition of backpropagation learning. *IEEE Transactions on Neural Networks*, *6*, 1300–1303.

Figure Captions

Figure 1: Feedforward network to solve the XOR task, showing notation for connecting and bias weights (Lisboa & Perantonis, 1991).

Figure 2: Points at (a) are relative minima only, points at (b) are relative minima and also part of a regional minimum while point (c) is a strict relative minimum, a regional minimum and a relative minimum.

Figure 3: Development of the approximation lemma (see text).

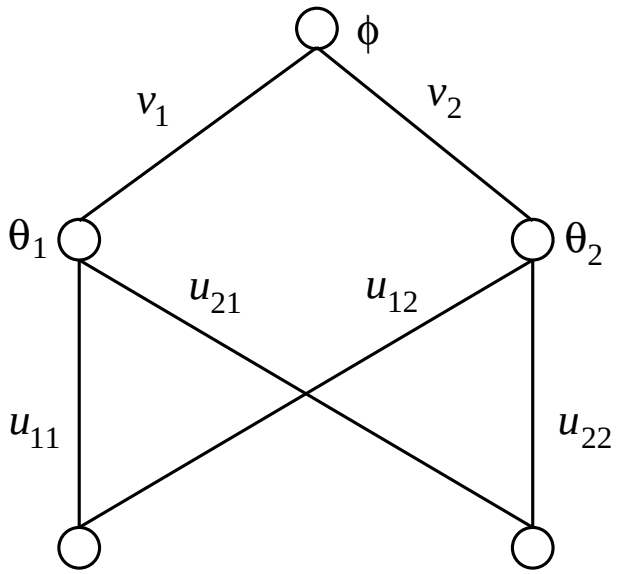

Figure 1: Feedforward network to solve the XOR task, showing notation for connecting and bias weights (Lisboa & Perantonis, 1991).

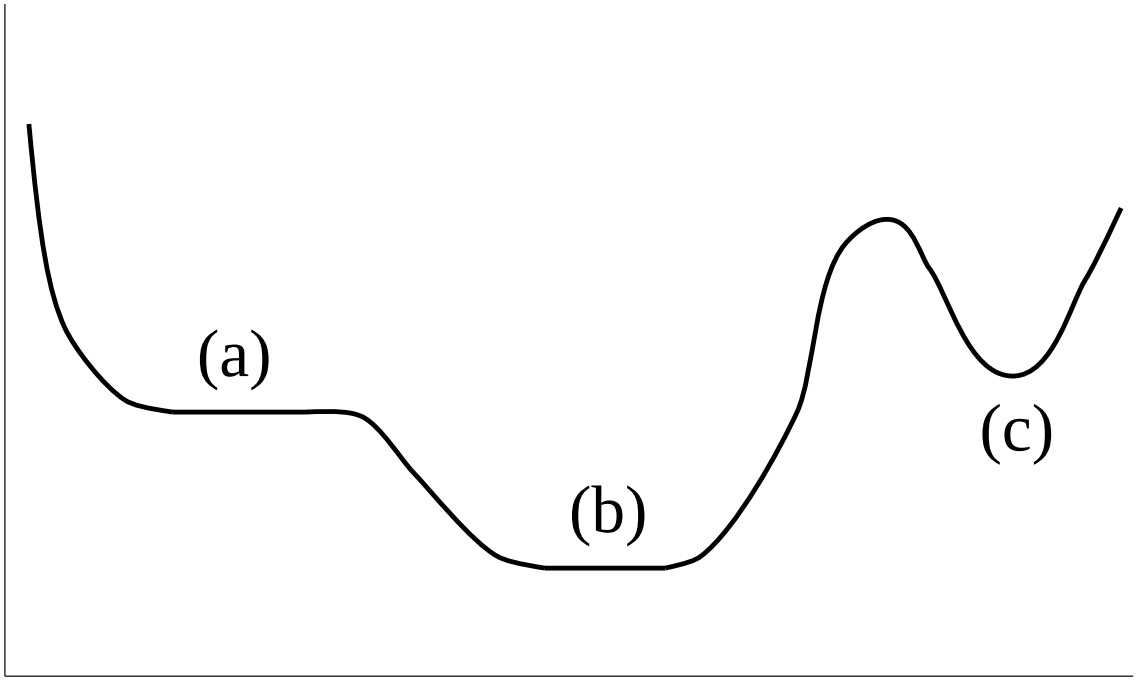

Figure 2: Points at (a) are relative minima only, points at (b) are relative minima and also part of a regional minimum while point (c) is a strict relative minimum, a regional minimum and a relative minimum.

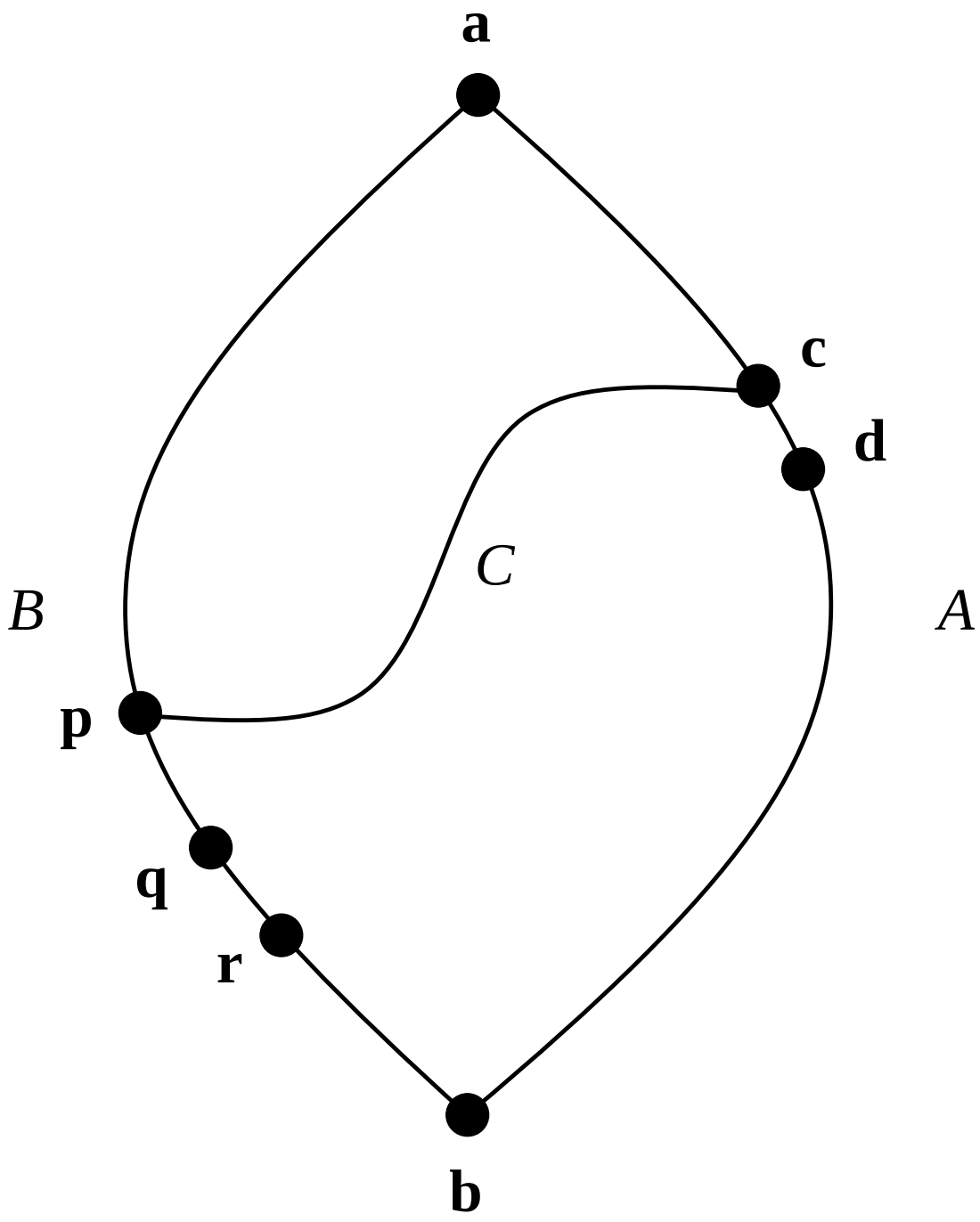

Figure 3: Development of the approximation lemma (see text).