The Shape of the Error Surfaces of some simple Neural Networks[1]

Ida G. Sprinkhuizen-Kuyper
Department of Computer Science
Leiden University
P.O. Box 9512
2300 RA Leiden
kuyper@wi.leidenuniv.nl

Egbert J.W. Boers
Department of Computer Science
Leiden University
P.O. Box 9512
2300 RA Leiden
boers@wi.leidenuniv.nl

Abstract

We investigated the error surfaces of two neural networks used for learning the XOR function with gradient descent methods. We found that the error surface of the simplest network with one hidden unit and the inputs connected to the output unit has no local minima. All stationary points of the error surface are saddle points. The error surface of the network with two hidden units without connections from the inputs to the output unit has no local minima for finite weights. For infinite weights from the inputs to the hidden units regions of local minima exist in the sense that all points in a neighbourhood of such a point result in error values greater than or equal to the error in the given point. Furthermore, it is shown that considering these local minima as a basin, the water will flow out of such a basin, since boundary points of these regions of local minima are saddle points. These results provide us with a better insight in problems encountered by different learning algorithms, when training on these particular networks and also give a feeling for problems that can be encountered in more complicated networks.

1 Introduction

A central theme in neural network research is to find the right network (architecture and learning algorithm) for a problem. In our research [BKH93a, BKH93b] we are trying to generate good architectures for neural networks using a genetic algorithm which works on strings containing coded production rules of a graph grammar (L-systems). These production rules result in an architecture and training of the architecture on a given problem results in a fitness for the given string, which is used by the genetic algorithm. In order to be able to decide objectively which architecture is better, a distinction is made between the following aspects:

- representation,
- learning and
- generalization.

The representation aspect considers whether a network is able to represent a solution of the problem. The learning aspect concerns the ability of a network to learn a solution of the

[1] In: J.C. Bioch and Y.-H. Tan (eds.); “NAIC’95: Proceedings of the Seventh Dutch Conference on Artificial Intelligence”, 275-284, Rotterdam, 1995. Also available as: Technical Report TR95-19, Computer Science Department, Leiden University.
URL: ftp://ftp.wi.leidenuniv.nl/pub/CS/TechnicalReports/1995/tr95-19.ps.gz

problem. If the network is able to learn a solution the question arises, how fast, with what probability and how accurate that solution will be learned. The last point is whether the network is able to generalize, i.e. does the network give reasonable output for patterns that were not part of the training set?

In order to learn more about these aspects we considered some simple networks for boolean functions. This paper is concerned with two networks that can represent the XOR function: the network with one hidden unit and connections from the inputs to the output unit (see figure 1, left), and the network with two hidden units without connections from the inputs to the output unit (see figure 1, right).

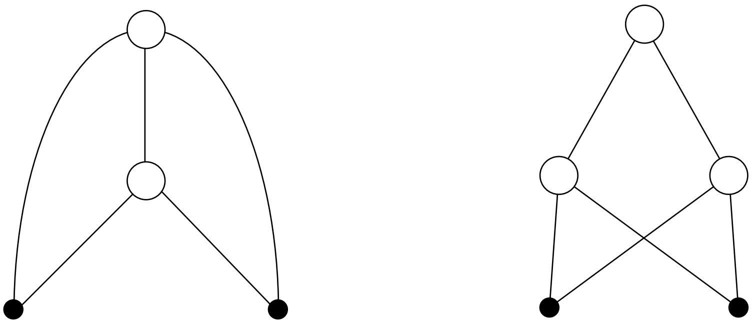

Figure 1. The XOR network with one hidden unit (left) and the XOR network with two hidden units (right).

As training algorithm we consider gradient-based algorithms, e.g. backpropagation with momentum. The error depends on the training pattern(s) and the weights. With a fixed training set the error is a function of the weights: the *error surface*. In the backpropagation algorithm the error in the output is reduced by changing the weight vector in the direction opposite to the gradient of the error with respect to the weights. So each weight w_{ij} is updated according to the following formula: $\Delta w_{ij}(t) = -\alpha\, \partial E/\partial w_{ij} + \beta\, \Delta w_{ij}(t-1)$, with learning parameter α and momentum parameter β. The effect is that the weights are updated such that a point on the error surface is reached with a smaller error value. There is a distinction between batch learning and on-line learning. During batch learning the weights are updated after the whole training set is seen and the errors of the individual patterns are summed to the total error. During on-line learning the weights are corrected after each pattern, with respect to the error for the pattern just seen by the network.

In this paper we will concern ourselves only with the error surfaces of the networks in figure 1 for the XOR problem and the consequences for the learning aspect. When we assume that some kind of gradient-based learning algorithm is used, then the shape of the error surface is very important. The ideal error surface has one minimum value (ideally zero) corresponding to an acceptable solution and in each other point a nonzero gradient. With such an error surface each gradient-based learning algorithm will approximate the minimum in this way finding a reasonable solution. However if the error surface contains so-called local minima, the learning algorithm can wind up in such a local minimum and reach a suboptimal solution. From experiments by Rumelhart et al. [RuM86] it seems that the simplest XOR network does not have such local minima in contrast to the XOR network with two hidden units and without connections from the inputs to the output. Many researchers [e.g. GST93, LiP91, RuM86] investigated the question whether or not an error surface for a certain network, which has to solve a certain problem, has local minima, and, if it has, how they should avoid them. Most researchers did numerical experiments, which provided them with a strong intuitive feeling of the existence of local minima, but not with a real proof. Lisboa and Perantonis [LiP91] give a local minimum, for example, for the

XOR network with two hidden units, with the weights from the hidden units to the output unit equal to zero, while it is proved in [SpB95] that such a point is a saddle point and *not* a local minimum.

The global minimum, with zero error, is not a strict minimum for both networks, since a higher dimensional region in the weight space exists with zero error. All points in a neighbourhood of each point of this region have error values which are *not less* than the error in that point. In a *strict* minimum, however, it is necessary that all points in a neighbourhood give error values *larger* than the error value in that point. Saddle points are stationary points where for each neighbourhood both points with larger error values and with smaller error values can be found. We proved that the global minimum contains the only points with a gradient equal to zero for the error of all patterns individually. We call such a point a *stable* stationary point. The other stationary points have a zero gradient for the error of a fixed training set of patterns, but not for the error of the patterns individually, so on-line learning can probably escape from these points.

For the network with one hidden unit we proved that all stationary points with error unequal to zero are saddle points. The complete proof is given in [SpB94a].

For the network with two hidden units we proved that all stationary points with finite weights and error unequal to zero are saddle points. For some of the weights from the inputs to the outputs equal to plus or minus infinity we found regions of local minima. These local minima are again not strict, so each neighbourhood of such a point contains points with the same error value. The dimension of these regions with local minima is greater than one and less than the dimension of the weight space. The “area” of the regions is infinite, while their “volume” in weight space is zero. Furthermore these regions of local minima all have boundary points which are saddle points, so it is possible to escape from such a local minimum through a boundary point of the surrounding region of local minima.

In these simple networks we observed two kinds of difficult points for a gradient based learning algorithm:

- saddle points, which sometimes need information about third or even fourth order partial derivatives to find the direction of decreasing error,
- regions of local minima with some infinite weights, which have boundary points that are saddle points.

These observations can be used to explain why experiments with a higher numerical precision less often get stuck into “local minima”, since the higher the numerical precision, the greater the change to escape from a difficult saddle point.

Also the result that on-line learning with a reasonably large learning parameter leads best to avoiding such minima [GST93], can be explained, since movement in the neighbourhood of a real local minimum can lead to reaching a boundary point and finding the way down.

The rest of the paper consists of the following sections: Section 2 contains a description of the XOR problem and the networks that are used to implement it. Section 3 contains a sketch of the proof that the error surface of the network with one hidden unit has no local minima. Section 4 exists of a sketch of the results for the network with two hidden units. Especially some of the regions with local minima are given, while the other regions with local minima can be derived from these regions by using transformations of the weights. Finally, section 5 contains the conclusions. This paper contains only a rough sketch of the proofs, elsewhere we give the complete proofs [SpB94a, SpB95].

2 The XOR problem and the two networks solving it

We studied the networks in figure 2: one with one hidden unit (left) and one with two hidden units (right). These networks consist of one threshold unit X_0, with constant value 1, two inputs X_1 and X_2, one or two hidden units and the output unit Y. The network with one hidden unit has seven weights which are labelled u_0, u_1, u_2, w_0, w_1, w_2 and v (see figure 2, left), while the network with two hidden units has nine weights labelled u, w_{01}, w_{02}, w_{11}, w_{12}, w_{21}, w_{22}, v_1 and v_2 (see figure 2, right). If each unit uses a sigmoid transfer function f—the used transfer function is $f(x) = 1/(1 + \mathrm{e}^{-x})$—then the output of the left network is, as function of the inputs X_1 and X_2:

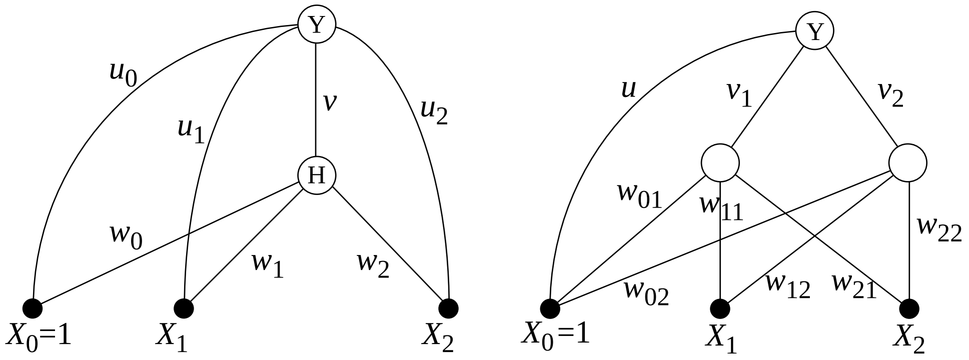

Figure 2. The XOR network with one hidden unit (left) and the XOR network with two hidden units (right). Here all weights including the threshold are shown.

$$y(X_1, X_2) = f(u_0 + u_1 X_1 + u_2 X_2 + v f(w_0 + w_1 X_1 + w_2 X_2)) \tag{2.1}$$

while the output of the right network is equal to:

$$y(X_1, X_2) = f(u_0 + v_1 f(w_{01} + w_{11} X_1 + w_{21} X_2) + v_2 f(w_{02} + w_{12} X_1 + w_{22} X_2)) \tag{2.2}$$

Table 1 shows the patterns for the XOR problem which have to be learned. The error E of the network when training a training set containing a_{ij} times the pattern P_{ij}, $a_{ij} > 0$, $i, j \in \{0,1\}$ is:

$$\begin{aligned} E = {} & \frac{1}{2} a_{00} (y(0,0) - 0.1)^2 + \frac{1}{2} a_{01} (y(0,1) - 0.9)^2 + \\ & \frac{1}{2} a_{10} (y(1,0) - 0.9)^2 + \frac{1}{2} a_{11} (y(1,1) - 0.1)^2 \end{aligned} \tag{2.3}$$

with $y(X_1, X_2)$ given in equation (2.1) and equation (2.2), respectively.

Table 1: Patterns for the XOR problem

Pattern	$\mathbf{X_1}$	$\mathbf{X_2}$	desired output
P_{00}	0	0	0.1
P_{01}	0	1	0.9
P_{10}	1	0	0.9
P_{11}	1	1	0.1

3 Sketch of the proof of the results for the simplest XOR network

The complete proof of the results for the network with one hidden unit and connections between the inputs and the output unit can be found in [SpB94a]. In the proof we distinguish two kinds of minima for the error E:

- Minima that remain stable during on-line learning independent of the chosen training sequence; these minima have the property that no pattern will lead to an error that can be decreased by a local chance of the weights. These minima are called *stable minima*.
- Minima that depend on the given training set. For batch learning these are minima, but during on-line learning the weights will continue to change in their neighbourhood, since they are not minima for all patterns separately. These minima are called *unstable minima*.

A similar distinction is made between stable and unstable stationary points.

The proof exists of the following steps:

- It is proved that the minimum with error zero can occur for finite values of the weights.
- It is proved that points with error zero are the unique stable minima.
- All unstable stationary points are investigated and are proved to be saddle points.

A typical saddle point is given in figure 3. This figure shows that indeed the error surface behaves as a saddle point in a neighbourhood of the point with all weights zero.

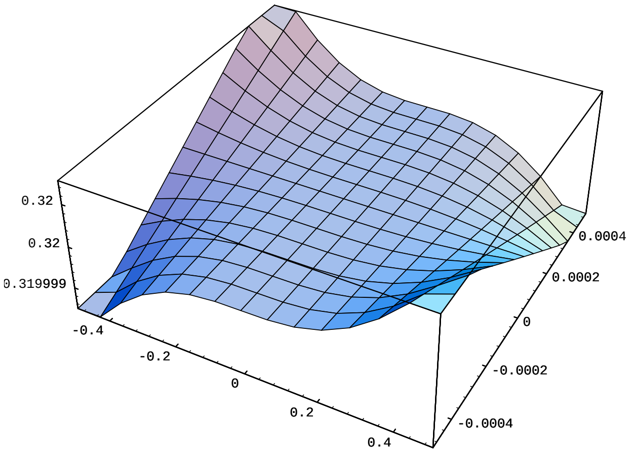

Figure 3. The error surface of the network with one hidden unit in the neighbourhood of $u_0 = u_1 = u_2 = w_0 = w_1 = w_2 = v = 0$. This picture is obtained by varying w_0, w_1 and w_2 equally from –0.5 to 0.5 and v from –0.0005 to 0.0005.

4 Sketch of the proof of the results for the XOR network with two hidden units

The complete proof of the results for the network with two hidden units can be found in [SpB95]. The proof exists of the following steps:

- It is proved that the minimum with error zero can occur for finite values of the weights.
- It is proved that points with error zero are the unique stable minima. This proof is straightforward for finite weights and more complicated if some of the weights to the

output unit are infinite such that the output is saturated to 0 or 1 for one or more of the patterns.

- All unstable stationary points with finite weights are investigated and are proved to be saddle points. So especially the point with all weights equal to zero is proved to be a saddle point in contrast to a paper of Blum [Blu89] who “proved” that this point was a local minimum. In [SpB94b] it is shown where Blum made the wrong conclusion. Figure 4 shows that this point is a saddle point indeed.

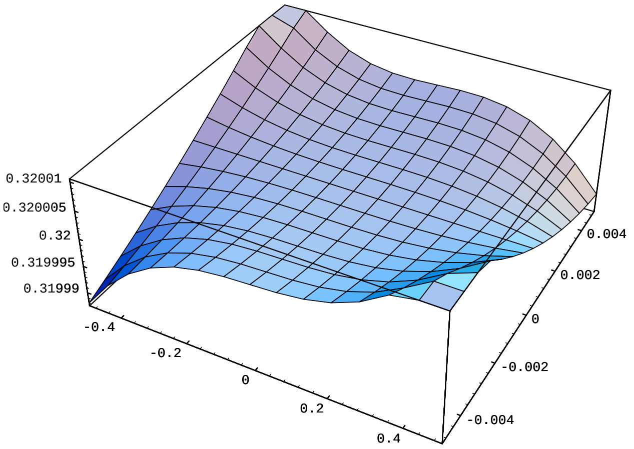

Figure 4. The error surface of the network with two hidden units around the point with all weights equal to zero. The values of the weights are varied such that $\Delta w_{11} = \Delta w_{12} = \Delta w_{21} = \Delta w_{22}$ varies from -0.5 to 0.5, while $\Delta v_1 = \Delta v_2$ varies from -0.005 to 0.005.

- The unstable stationary points, with some of the weights infinite, are investigated. Here we found the following local minima:

Local minima with two patterns learned

If the patterns P_{00} and P_{01} (see table 1) are learned exactly and the output is equal to 0.5 for the other two patterns, the following 4-dimensional regions of local minima exist.

Let us denote the following relations by the symbols Q_1, Q_2, S_1 and S_2, respectively:

Q_1: $v_1 \geq f^{-1}(0.9)$ and $w_{21} > 0$, Q_2: $v_1 \leq f^{-1}(0.1)$ and $w_{21} < 0$,

S_1: $v_2 \geq f^{-1}(0.9)$ and $w_{22} > 0$, S_2: $v_2 \leq f^{-1}(0.1)$ and $w_{22} < 0$.

The local minima occur for values of the weights which obey the following equalities and inequalities:

$$u + v_1 f(w_{01}) + v_2 f(w_{02}) = f^{-1}(0.1) \quad , \tag{4.1}$$

$$u + v_1 f(w_{01} + w_{21}) + v_2 f(w_{02} + w_{22}) = f^{-1}(0.9) \tag{4.2}$$

and either:

- $u+v_1+v_2 = 0$, $w_{01}+w_{11} = w_{01}+w_{11}+w_{21} = \infty$, $w_{02}+w_{12} = w_{02}+w_{12}+w_{22} = \infty$ and ((Q_1 and S_2) or (Q_2 and S_1)), or

- $u+v_1 = 0$, $w_{01}+w_{11} = w_{01}+w_{11}+w_{21} = \infty$, $w_{02}+w_{12} = w_{02}+w_{12}+w_{22} = -\infty$ and ((Q_1 and S_1) or (Q_2 and S_2)), or
- $u+v_2 = 0$, $w_{01}+w_{11} = w_{01}+w_{11}+w_{21} = -\infty$, $w_{02}+w_{12} = w_{02}+w_{12}+w_{22} = \infty$ and ((Q_1 and S_1) or (Q_2 and S_2)), or
- $u = 0$, $w_{01}+w_{11} = w_{01}+w_{11}+w_{21} = -\infty$, $w_{02}+w_{12} = w_{02}+w_{12}+w_{22} = -\infty$ and ((Q_1 and S_2) or (Q_2 and S_1)).

It is possible to escape from all these local minima through points with $w_{21} = 0$ or $w_{22} = 0$.

Similar local minima are found starting from the above mentioned, by applying transformations of the weights, resulting in local minima with two other patterns exactly learned.

Proof of the first local minimum

The first local minimum is found if $w_{01}+w_{11} = w_{01}+w_{11}+w_{21} = w_{02}+w_{12} = w_{02}+w_{12}+w_{22} = \infty$. Let A_{ij} be the input of the output unit corresponding to pattern P_{ij}. Then we have:

$$\begin{aligned} A_{00} &= u + v_1 f(w_{01}) + v_2 f(w_{02}) = f^{-1}(0.1) \\ A_{01} &= u + v_1 f(w_{01} + w_{21}) + v_2 f(w_{02} + w_{22}) = f^{-1}(0.9) \\ A_{10} &= A_{11} = u + v_1 + v_2 = 0 \end{aligned} \tag{4.3}$$

The corresponding error level is 0.16. Elimination of u results in:

$$\begin{aligned} A_{00} &= -v_1 f(-w_{01}) - v_2 f(-w_{02}) = f^{-1}(0.1) \approx -2.197 \\ A_{01} &= -v_1 f(-w_{01} - w_{21}) - v_2 f(-w_{02} - w_{22}) = f^{-1}(0.9) \approx 2.197 \end{aligned} \tag{4.4}$$

Since $f(x)$ is positive and A_{00} and A_{01} have opposite sign, v_1 and v_2 will have opposite sign and will not be equal to zero in this case. Since $f(x) \in [0,1]$ it follows that either $v_1 \geq f^{-1}(0.9)$ and $v_2 \leq f^{-1}(0.1)$ or $v_1 \leq f^{-1}(0.1)$ and $v_2 \geq f^{-1}(0.9)$. In order that the equations (4.4) have a solution for v_1 and v_2 the following condition must be fulfilled:

$$f(-w_{01}) f(-w_{02} - w_{22}) \neq f(-w_{02}) f(-w_{01} - w_{21}) \tag{4.5}$$

Since $w_{01}+w_{11} = w_{02}+w_{12} = +\infty$, we make the substitution:

$$p_1 = e^{-w_{01} - w_{11}} \text{ and } p_2 = e^{-w_{02} - w_{12}}$$

If $w_{01}+w_{11} \to \infty$ then $p_1 \downarrow 0$ and if $w_{02}+w_{12} \to \infty$ then $p_2 \downarrow 0$. Computation of the partial derivatives of the error E with respect to p_1 and p_2 for p_1 and p_2 equal to zero, choosing w_{01}, w_{21}, w_{02} and w_{22} independent of p_1 and p_2, results in:

$$\begin{aligned} \left.\frac{\partial E}{\partial p_1}\right|_{p_1 = 0} &= 0.4 f'(0)\, v_1 (1 - e^{-w_{21}}) \\ \left.\frac{\partial E}{\partial p_2}\right|_{p_2 = 0} &= 0.4 f'(0)\, v_2 (1 - e^{-w_{22}}) \end{aligned} \tag{4.6}$$

Since both p_1 and p_2 are greater then or equal to zero it is clear that if one of the derivatives in equation (4.6) is negative, then the error will decrease if p_1 or p_2 moves away from zero (and $w_{01}+w_{11}$ or $w_{02}+w_{12}$ moves away from infinity, correspondingly). Thus then the stationary point is not a local minimum.

If both derivatives in equation (4.6) are positive, increasing p_1 and/or p_2 will lead to an increase of the error, but when p_1 and p_2 are equal to zero, it is clear from equations (4.3) and (4.4) that decreasing the error can be done only by altering $u+v_1+v_2$, such that the error corresponding to A_{10} and A_{11} decreases, and altering the other weights in order to keep the error corresponding to A_{00} and A_{01} equal to zero. But the error corresponding to A_{10} and A_{11} as a function of $x = u+v_1+v_2$ is equal to:

$$E = \frac{1}{2}(f(x) - 0.9)^2 + \frac{1}{2}(f(x) - 0.1)^2 \tag{4.7}$$

with derivatives for $x = 0$:

$$\left.\frac{\partial E}{\partial x}\right|_{x=0} = (f(x) - 0.9) f'(x) + (f(x) - 0.1) f'(x) \big|_{x=0} = 0 \tag{4.8}$$

and

$$\left.\frac{\partial^2 E}{\partial x^2}\right|_{x=0} = 2\{f'(x)\}^2 - f''(x) \big|_{x=0} = \frac{1}{8} > 0 \tag{4.9}$$

So each variation of $u+v_1+v_2$ will increase the error with respect to A_{10} and A_{11}. So in this case a local minimum is found! The sign of the derivatives in (4.6) is determined by the signs of v_1, v_2, w_{21} and w_{22}. Both derivatives are positive if (Q_1 and S_2) or (Q_2 and S_1).

The dimension of the region in which this minimum value is attained follows from (4.3), (4.4) and (4.5): if w_{01}, w_{02}, w_{21} and w_{22} are chosen such that the inequality (4.5) holds, then u, v_1 and v_2 are determined by (4.3) and (4.4). So the dimension of this region of local minima is 4.

Local minima with one pattern learned

Also local minima exist with one pattern learned exactly. The following conclusion gives one of the results if the pattern P_{00} is exactly learned. Results for the case with one of the other patterns exactly learned follow from the conclusions for P_{00} by transformations of the weights.

Conclusion *If P_{00} is learned exactly and the other patterns are not, then regions with local minima with error* 0.213333 *will be found if $w_{01}+w_{11} = w_{01}+w_{21} = w_{01}+w_{11}+w_{21} = w_{02}+w_{12} = w_{02}+w_{22} = w_{02}+w_{12}+w_{22} = \infty$ and if the following equations hold:*

- $u+v_1 f(w_{01})+v_2 f(w_{02}) = f^{-1}(0.1)$, *and* $u+v_1+v_2 = f^{-1}(1.9/3)$

  and if one of the following conditions is fulfilled:

- $w_{01} = w_{12} = w_{22} = \infty$, w_{11}, w_{21} *and* w_{02} *are finite and either*
  - $v_1 > 0$, $v_2 > 0$ *and* $e^{-w_{11}} + e^{-w_{21}} - 2e^{-w_{11} - w_{21}} > 0$ *or*
  - $v_1 < 0$, $v_2 > 0$ *and* $e^{-w_{11}} + e^{-w_{21}} - 2e^{-w_{11} - w_{21}} < 0$

  or

- $w_{11} = w_{21} = w_{02} = \infty$, w_{01}, w_{12} *and* w_{22} *are finite and either*
  - $v_1 > 0$, $v_2 > 0$ *and* $e^{-w_{12}} + e^{-w_{22}} - 2e^{-w_{12}-w_{22}} > 0$ *or*
  - $v_1 > 0$, $v_2 < 0$ *and* $e^{-w_{12}} + e^{-w_{22}} - 2e^{-w_{12}-w_{22}} < 0$

  or
- $w_{11} = w_{21} = w_{12} = w_{22} = \infty$, w_{01} *and* w_{02} *are finite and* $v_1 > 0$ *and* $v_2 > 0$.

Similar conclusions exist with respect to the existence of local minima for the cases

- $w_{01}+w_{11} = w_{01}+w_{21} = w_{01}+w_{11}+w_{21} = \pm\infty$ and $w_{02}+w_{12} = w_{02}+w_{22} = w_{02}+w_{12}+w_{22} = \mp\infty$ and
- $w_{01}+w_{11} = w_{01}+w_{21} = w_{01}+w_{11}+w_{21} = w_{02}+w_{12} = w_{02}+w_{22} = w_{02}+w_{12}+w_{22} = -\infty$.

The boundary points of all these regions with local minima are saddle points.

In [SpB95] it is shown that 4 of the 5 examples given by Lisboa and Perantonis [LiP91] correspond to one of the regions of local minima found in this research. The fifth example was not a correct example of a local minimum, since the given point is a saddle point. The first example given by Lisboa and Perantonis is:

> *The patterns* P_{00} *and* P_{10} *are learned exactly, the output of the other patterns is equal to* 0.5, *while the weights are equal to:* $u = 5.05670$, $v_1 = -2.78335$, $v_2 = -5.05670$, $w_{01} = 1.41913$, $w_{11} = -5.52058$, $w_{21} = -13.69016$, $w_{02} = 4.73579$, $w_{12} = -4.50867$ *and* $w_{22} = 12.27468$.

This point is a numerical approximation of a point with $w_{01}+w_{21} = w_{01}+w_{11}+w_{21} = -\infty$, $w_{02}+w_{22} = w_{02}+w_{12}+w_{22} = \infty$, $v_1 \le f^{-1}(0.1)$, $w_{11} < 0$, $v_2 \le f^{-1}(0.1)$, $w_{12} < 0$ and further restrictions on the weights such that the patterns indeed give the given output values. Also the local minimum given by Rumelhart and McClelland [RuM86] corresponds to a region of local minima found in [SpB95].

5 Conclusion

Our main conclusion is that the error surface of the network with one hidden unit for the XOR problem has no local minima. So for this network from each point in weight space a path with decreasing error exists leading to a point with error 0. Regions of saddle points exist where some algorithms can get stuck or be retarded.

The error surface of the network with two hidden units has no local minima for finite values of the weights. Regions of local minima exist for some of the weights from the inputs to the hidden units having infinite values, but these regions have boundary points which are saddle points. So for this network from each point in weight space a non-increasing path exist to a point with error 0, while from each point outside a local minimum—so especially from each point with finite weights—a strictly decreasing path exists to a point with error zero.

In this paper we gave the results of our investigation of the error surfaces of two simple neural networks. To investigate such error surfaces thoroughly is important in order be able to explain the behaviour of learning algorithms dealing with such surfaces. Many researchers who study learning algorithms start testing their algorithms on the XOR or n-bit parity problem (Prechelt [Pre94]). Not knowing why exactly some learning algorithms wind up in a point with positive error, while other algorithms do not or less often, they give vague explanations and, for example, say that a shallow local minimum is found which is abandoned by an algorithm with a higher momentum term. Since in this paper we showed the existence of regions of local minima with boundary points which are saddle points, it is interesting to investigate what kind of learning algorithms easily can escape from such local minima. A suggestion could be some learning algorithm that works like pouring water in a local minimum and sees what direction the water flows.

6 Acknowledgements

We would like to thank the referees for their valuable suggestions for improving this paper.

7 References

[Blu89] E.K. Blum; "Approximation of Boolean Functions by Sigmoidal Networks: Part I: XOR and other Two-variable functions". In: *Neural Computation* 1, 532-540, 1989.

[BKH93a] E.J.W. Boers, H. Kuiper, B.L.M. Happel and I.G. Sprinkhuizen-Kuyper; "Biological metaphors in designing modular artificial neural networks". In: S. Gielen and B. Kappen (eds.); *Proceedings of the International Conference on Artificial Neural Networks*, Springer-Verlag, Berlin, 1993.

[BKH93b] E.J.W. Boers, H. Kuiper, B.L.M. Happel and I.G. Sprinkhuizen-Kuyper; "Designing Modular Artificial Neural Networks". In: H.A. Wijshoff (ed.); *Proceedings of Computing Science in the Netherlands CSN'93*, 87–96, 1993.

[GST93] D. Gorse, A. Shepherd and J.G. Taylor; "Avoiding Local Minima by Progressive Range Expansion". In: S. Gielen and B. Kappen (eds.); *Proceedings of the International Conference on Artificial Neural Networks*, Springer-Verlag, Berlin, 1993. (added)

[LiP91] P.J.G. Lisboa and S.J. Perantonis; "Complete solution of the local minima in the XOR problem", *Network 2*, pp. 119–124, 1991.

[Pre94] L. Prechelt; *A study of Experimental Evaluations of Neural Network Learning Algorithms: Current Research Practice*, Technical Report 19/94, Fakultät für Informatik, Universität Karlsruhe, 1994.

[RuM86] D.E. Rumelhart, J.L. McClelland and the PDP Research Group; *Parallel Distributed Processing, Volume 1*. The MIT Press, Cambridge, Massachusetts, 1986.

[SpB94a] I.G. Sprinkhuizen-Kuyper and E.J.W. Boers; *The Error Surface of the simplest XOR Network has no local Minima*. Technical Report 94-21, Leiden University, Dept. of Computer Science, The Netherlands.

[SpB94b] I.G. Sprinkhuizen-Kuyper and E.J.W. Boers; *A Comment on a Paper of Blum: Blum's "local minima" are saddle points.* Technical Report 94-34, Leiden University, Dept. of Computer Science, The Netherlands.

[SpB95] I.G. Sprinkhuizen-Kuyper and E.J.W. Boers; *The Error Surface of the XOR network with two hidden Units*, to appear.