Classification of all Stationary Points on a Neural Network Error Surface

I.G. Sprinkhuizen-Kuyper
and
E.J.W. Boers
Department of Computer Science
Leiden University
email: {kuyper,boers}@wi.leidenuniv.nl

Abstract

The artificial neural network with one hidden unit and the input nodes connected to the output node is considered. It is proven that the error surface of this network for the patterns of the XOR problem has minimum values with zero error and that all other stationary points of the error surface are saddle points. Also, the volume of the regions in weight space with saddle points is zero, hence training this network, using e.g. backpropagation with momentum, on the four patterns of the XOR problem, the correct solution with error zero will be reached in the limit with probability one.

1 Introduction

A central theme in neural network research is to find the right network (architecture and learning algorithm) for a problem. Some learning algorithms also influence the architecture (pruning and construction, see e.g. [5, 7]). In our research [1, 2, 3] 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 problem. If the network is able to learn a solution, how fast, with what probability and how accurate will that solution 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 in the training set?

In order to learn more about these aspects we considered some simple networks for boolean functions. This paper is concerned with the simplest network that can represent the XOR function: one hidden unit and connections from the input units to the output unit (see figure 2). As training algorithm we take some gradient-based algorithm, 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: $\Delta w_{ij}(t) = -\alpha\, \partial E/\partial w_{ij} + \beta\, \Delta w_{ij}(t-1)$, with learning parameter α and momentum parameter β. This has the effect that the weights are updated such that a point on the error surface is reached with a smaller error value. Distinction can be made 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, while during on-line learning the weights are corrected after each pattern, with respect to the error for the pattern just seen by the network.

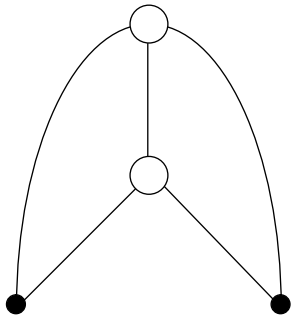

Figure 1. The simplest XOR network

1.1 Representation

First we looked at the representational power of the simplest XOR network. It is well known that this network with a threshold transfer function can represent the XOR function and that such a network with a sigmoid transfer function can approximate a solution of the XOR function. In this paper we will show that such a network with a sigmoid transfer function can represent the XOR function exactly if TRUE ~ 0.9 and FALSE ~ 0.1 (the values 0.9 and 0.1 are used, but all values 1-δ and δ, for some small positive number δ, can also be used). This result is not trivial, since for a one-layer network[1] for the AND function, it is possible to find an approximate representation, but it is not possible to solve the AND function exactly.

1.2 Learning

The next step is: what about learning? 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, and so find a reasonable solution. However if the

1. We do not count the input as a layer of the network.

error surface has so-called local minima, then the learning algorithm can wind up in such a local minimum and reach a suboptimal solution. From experiments by Rumelhart et al. [9] 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. The problem whether an error surface for a certain network that has to solve a certain problem, has local minima or not (and if they exist, how to avoid them) is investigated by many researchers [e.g. 6, 7, 8, 9]. Most researchers did numerical experiments, which gave a strong intuitive feeling of the existence of local minima, but not a real proof. Lisboa and Perantonis [8] give a local minimum, for example, for the XOR network with two hidden units and without connections from the inputs to the output, with the weights from the hidden units to the output unit equal to zero, while by similar techniques as used by Sprinkhuizen-Kuyper and Boers [12] it can be shown that such a point is a saddle point and *not* a local minimum. In contrast to Lisboa and Perantonis, who suggest that the simplest XOR network has local minima, this paper will analytically *prove* that the error surface of the simplest XOR network has *no* local minima.

The global minimum, with zero error, is not a strict minimum, since a 3-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. There exist more stationary points (i.e. points where the gradient of the error is zero), but we were able to prove that all these points are saddle points. Saddle points are stationary points where for each neighbourhood both points with larger error values and with smaller error values can be found. Also 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 saddle 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 standard XOR network with two hidden units, we already proved that it has zero as stable global minimum, and that other minima can not be stable. Results that on-line learning with a reasonably large learning parameter leads best to avoiding such minima [6], can be explained from this fact.

1.3 Generalization

The third point is the ability to generalize. The work of Denker, Schwartz, Solla et al. [4, 10, 11] suggests to investigate the a priori probability that a network represents a certain function when the weights are chosen randomly. We did some computations for the XOR networks with threshold units and numerically determined the a priori probability of the network to represent an approximation of the XOR function, relatively to the probability of representing one of the other boolean functions of two inputs. Our results tell that this probability is very small ($\approx$0.005 for the simplest network, and $\approx$0.0013 for the network with

two hidden units). Thus, if less than four patterns of the XOR problem are used to train the network, almost always one of the other boolean functions corresponding to that pattern will be learned, and not the XOR function. However, this also suggests that more regular functions like AND and OR are preferred, if possible, above the XOR function. In a forthcoming paper we will publish our results of several measurements of a priori probabilities for several functions.

The remainder of the paper consists of the following sections: In section 2 the XOR problem and the network that is used to implement it are given. In section 3 the proof is sketched that a 3-dimensional region of the weight space exist with zero error. In section 4 it is shown that all finite points with nonzero error are unstable, i.e. the gradient of the error with respect to one single pattern is unequal to zero, and that local minima can occur only for finite values of the weights. Section 5 exists of the proof that all points with nonzero error and zero gradient (averaged for a training set) are saddle points. Finally section 6 contains the conclusions. This paper contains only a rough sketch of most proofs, elsewhere we give the complete proofs [12].

2 The XOR problem and the simplest network solving it

The network in figure 2 with one hidden unit H is studied. This network con-

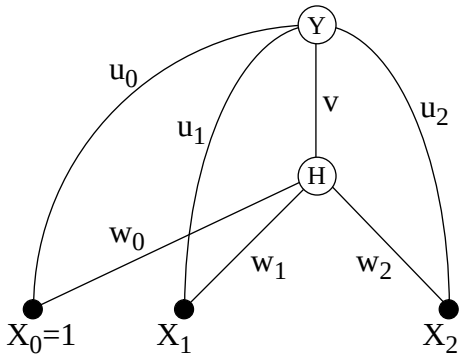

Figure 2. The simplest XOR network

sists of one threshold unit X_0, with constant value 1, two inputs X_1 and X_2, one hidden unit H and the output unit Y. There are seven weights which are labelled u_0, u_1, u_2, w_0, w_1, w_2 and v (see figure 2). If each unit uses a sigmoid transfer function f—the used transfer function is $f(x) = 1/(1+e^{-x})$—then the output of this network is, as function of the inputs X_1 and X_2:

$$y(X_1, X_2) = f(u_0 + u_1X_1 + u_2X_2 + vf(w_0 + w_1X_1 + w_2X_2)) \tag{2.1}$$

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:

$$E = \frac{1}{2}a_{00}\,(y\,(0,0) - 0.1)^{\,2} + \frac{1}{2}a_{01}\,(y\,(0,1) - 0.9)^{\,2} + \frac{1}{2}a_{10}\,(y\,(1,0) - 0.9)^{\,2} + \frac{1}{2}a_{11}\,(y\,(1,1) - 0.1)^{\,2} \tag{2.2}$$

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

Table 1: Patterns for the XOR problem

Pattern	X_1	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 The minimum $E = 0$ can occur

It can be shown that a 3-dimensional region in the 7-dimensional weight space exists for which the error is exactly zero. The error E consists of four quadratic terms, so $E = 0$ holds only if all terms are zero. The four equations for the weights thus obtained are considered. From these equations four linear equations for the three weights u_0, u_1 and u_2 in terms of the other weights are found. It is shown that for almost all values of the three weights w_0, w_1 and w_2 it is possible to find a value of v such that the equations for u_0, u_1 and u_2 have a (unique) solution. This results in a 3-dimensional region depending on w_0, w_1 and w_2. We will 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 will be called *stable minima*.
- Minima that depend on the given training set. For batch learning this is a minimum, but during on-line learning the weights will continue to change in a neighbourhood of such a minimum, since it is not a minimum for all patterns separately. These minima will be called *unstable minima*.

If E is equal to zero for all patterns that are in the training set, given a certain set of weights, a stable minimum is found. E can become equal to zero if and only if values of the weights u_0, u_1, u_2, w_0, w_1, w_2 and v exist such that the following four equations hold:

$$\begin{aligned} f(u_0 + vf(w_0)) &= 0.1 \\ f(u_0 + u_2 + vf(w_0 + w_2)) &= 0.9 \\ f(u_0 + u_1 + vf(w_0 + w_1)) &= 0.9 \\ f(u_0 + u_1 + u_2 + vf(w_0 + w_1 + w_2)) &= 0.1 \end{aligned} \tag{3.1}$$

Application of the inverse function f^{-1} on both sides of these equations leads to:

$$\begin{aligned} u_0 + vf(w_0) &= f^{-1}(0.1) \approx -2.197 \\ u_0 + u_2 + vf(w_0 + w_2) &= f^{-1}(0.9) \approx 2.197 \\ u_0 + u_1 + vf(w_0 + w_1) &= f^{-1}(0.9) \approx 2.197 \\ u_0 + u_1 + u_2 + vf(w_0 + w_1 + w_2) &= f^{-1}(0.1) \approx -2.197 \end{aligned} \tag{3.2}$$

It can be shown that for each value of the weights w_0, w_1 and w_2 where

$$f(w_0) - f(w_0 + w_1) - f(w_0 + w_2) + f(w_0 + w_1 + w_2) \neq 0 \tag{3.3}$$

unique values of the other weights u_0, u_1, u_2 and v can be found such that all equations of (3.2) hold. The equation

$$f(w_0) - f(w_0 + w_1) - f(w_0 + w_2) + f(w_0 + w_1 + w_2) = 0 \tag{3.4}$$

has the solutions

$$w_1 = 0 \quad \text{or} \quad w_2 = 0 \quad \text{or} \quad w_1 + w_2 + 2w_0 = 0\,. \tag{3.5}$$

4 The minimum $E = 0$ is the unique stable minimum

In order to obtain a stable minimum, it is necessary that the gradient of the error for each pattern is zero. Let us consider the partial derivative of E with respect to u_0. Writing R_{ij} for the terms depending on pattern P_{ij} we obtain:

$$\frac{\partial E}{\partial u_0} = R_{00} + R_{01} + R_{10} + R_{11} \tag{4.1}$$

with

$$\begin{aligned} R_{00} &= a_{00}\,(f(u_0 + vf(w_0)) - 0.1)\,f'(u_0 + vf(w_0)) \\ R_{01} &= a_{01}\,(f(u_0 + u_2 + vf(w_0 + w_2)) - 0.9)\,f'(u_0 + u_2 + vf(w_0 + w_2)) \\ R_{10} &= a_{10}\,(f(u_0 + u_1 + vf(w_0 + w_1)) - 0.9)\,f'(u_0 + u_1 + vf(w_0 + w_1)) \\ R_{11} &= a_{11}\,(f(u_0 + u_1 + u_2 + vf(w_0 + w_1 + w_2)) - 0.1)\cdot \\ &\quad f'(u_0 + u_1 + u_2 + vf(w_0 + w_1 + w_2)) \end{aligned}$$

The derivative $\partial E / \partial u_0$ is only equal to zero for each training set if

$$R_{00} = R_{01} = R_{10} = R_{11} = 0\,. \tag{4.2}$$

So all stable stationary points satisfy (4.2). The condition (4.2) is not only a necessary condition for a stable stationary point, but it is also sufficient, since if it holds then the partial derivatives of E with respect to the other weights also will be zero. Clearly the points such that the equations of (3.1) hold and thus the points with $E = 0$ are stable stationary points. Other stable stationary points can be found when one or more of the arguments of the derivative of the transfer function approach $\pm\infty$. The corresponding outputs for those patterns go to zero or one. We have shown that if such a point is approached, it is always possible to leave the neighbourhood of such a point via a path with decreasing error.

5 All unstable stationary points with $E \neq 0$ are saddle points

It is proved that all unstable stationary points with $E \neq 0$ are saddle points. Examination of the equations for $\nabla E = 0$ leads to three equations which have to be satisfied by the considered points. The proof is splitted into the cases where the weight $v = 0$ and the cases where $v \neq 0$. In the cases where $v = 0$ all partial derivatives of the error with respect to w_0, w_1 and w_2 are zero. It is proved that the first partial derivative of the error of the form $\partial^{i+j+1}E/\partial {w_1}^i \partial {w_2}^j \partial v$ which is unequal to zero determines that these points are saddle points. The cases where $v \neq 0$ are solved by considering the behaviour of the error on some carefully selected curves. The complete derivation is given elsewhere [12]. Here, some pictures are given showing some of the saddle points, visualized with Mathematica.

Figure 3 shows that indeed the error surface behaves as a saddle point when in

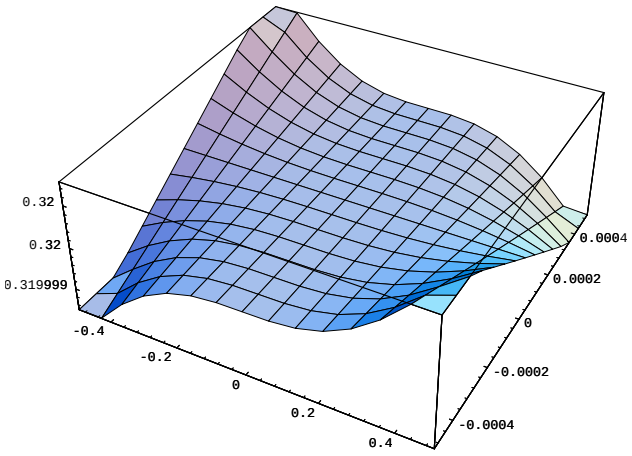

Figure 3. The error surface in the neighbourhood of $u_0 = u_1 = u_2 = w_0 = w_1 = w_2 = v = 0$. This picture is obtained by varying w_0, w_1 and w_2 equally from –0.5 to 0.5 and v from –0.0005 to 0.0005.

a neighbourhood of the point with all weights zero, the weights w_0, w_1, w_2 and v are varied such that $\Delta w_0 = \Delta w_1 = \Delta w_2$ and Δv is very small with respect to Δw_i. Figure 4, 5 and 5 concern two of the saddle points with $v = 0$.

6 Conclusion

The error surface of the network with one hidden unit for the XOR function has no local minima, only one global minimum with zero error. This minimum value is attained in a 3-dimensional region of the 7-dimensional weight space. Also a number of low dimensional regions exist where the error surface behaves as a saddle point (dimension 2 for the case $v = 0$, and dimension 1 for the other cases). The levels of the error surface in the saddle points are 0.32, 0.407392 and 0.403321, respectively, for a training set with exactly one example of each pattern (see [12]). When training is started with small weights, only

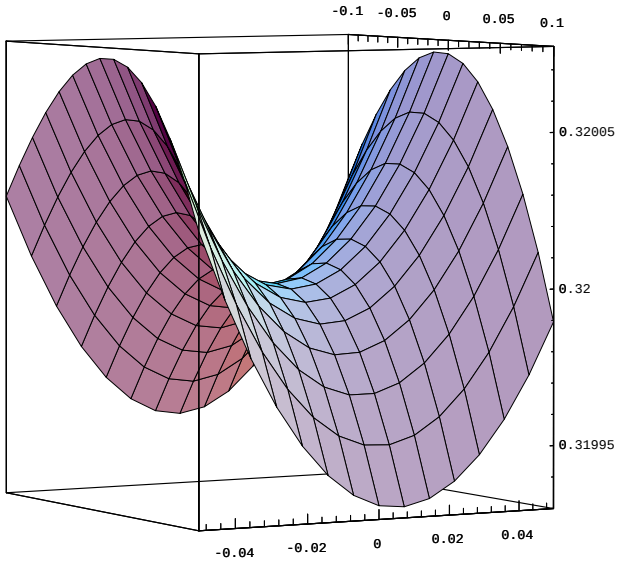

Figure 4. The error surface in the neighbourhood of the point $u_0 = -f(0.5)$, $u_1 = u_2 = 0$, $w_0 = 0.5$, $w_1 = w_2 = 0$, $v = 1$. The downward bow of the saddle is obtained by varying w_1, w_2, u_1 and u_2 such that $\Delta u_1 = \Delta u_2 = -f'(0)\Delta w_1 = -f'(0)\Delta w_2$. The other direction is given by varying w_0 and u_0 such that $\Delta u_0 = f'(0)\Delta w_0$.

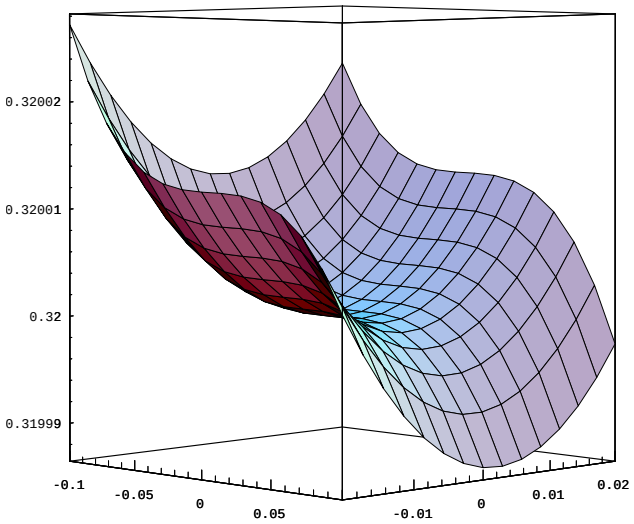

Figure 5. The saddle point in the neighbourhood of $u_0 = -f(0)$, $u_1 = u_2 = 0$, $w_0 = w_1 = w_2 = 0$ and $v = 1$. This picture is obtained by plotting the error against $u_1 = u_2 = -f'(0)w_1 = -f'(0)w_2$ and $u_0 = f'(0)w_0$. The weight w_1 runs from –0.1 to 0.1 and the weight w_0 runs from –0.02 to 0.02.

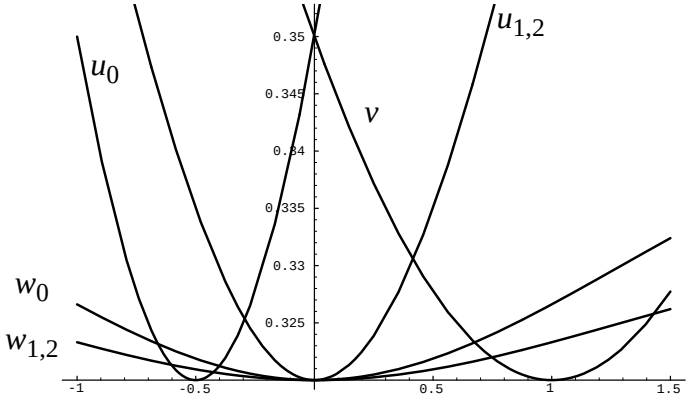

Figure 6. The error as function of each of the weights in the neighbourhood of $u_0 = –0.5$, $u_1 = u_2 = 0$, $w_0 = w_1 = w_2 = 0$ and $v = 1$. The curves for w_1 and w_2 and those for u_1 and u_2 are identical. This picture gives the (false) impression that the error has a local minimum if $u_0 = –0.5$, $u_1 = u_2 = 0$, $w_0 = w_1 = w_2 = 0$ and $v = 1$. Figure 5 showed already that this point is a saddle point.

a saddle point with error level 0.32 is possibly reached. The probability that the learning process will start in a saddle point or will end up in a saddle point is zero since the dimension of the region consisting of saddle points is at most 2, so its volume as part of the 7-dimensional weight space is zero.

When a saddle point is encountered, a batch learning process with zero momentum term can wind up in such a saddle point, but an on-line learning process can probably escape from such a saddle point, since the error surface is not horizontal for each individual pattern, only the average error surface for all patterns is horizontal. So a small change of the weights in the right direction will decrease the error, moving away from the saddle point. We did some experiments starting on-line learning exactly in the saddle point with all weights equal to zero and found that even with a small value of the learning parameter (0.01) and no momentum term the learning algorithm escaped from the saddle point and reached a solution with (almost) zero error. Using batch learning no progress was made to escape from the saddle point.

In this paper distinction is made between stable minima (minima for each pattern) and unstable minima (minima for a training set of patterns, but not for each pattern separately). This distinction is relevant, since if an exact solution can be represented by the network, then only the absolute minima with $E = 0$ are stable minima and all other (local) minima are unstable.

The fact that all local minima are unstable can be exploited by the learning algorithm to escape from these minima. Also the shape of the error surface at a minimum (narrow or wide) might determine how easy it is to escape from this minimum. Further research is necessary to examine this.

Another possible use of looking at the shape of the error surface when an exact representation of a problem is not possible (e.g. when noise is present), is finding an estimator for the generalization of a given weight configuration. We expect that very narrow minima will show a worse generalization than very wide minima with the same residual error on the training patterns. Further research is necessary to find a good measure for the shape of a minimum (especially considering the fact that several scaling factors are present) and to obtain experimental and theoretical results in this direction.

References

[1] E.J.W. Boers and H. Kuiper; *Biological Metaphors and the Design of Modular Artificial Neural Networks*, Master's Thesis, Leiden University, 1992.

[2] 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.

[3] 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.

[4] J. Denker, D. Schwartz, B. Wittner, S. Solla, R. Howard, L. Jackel and J. Hopfield; "Large Automatic Learning, Rule Extraction, and Generalization". *Complex Systems 1*, pp. 877–922, 1987.

[5] S.E. Fahlman and C. Lebiere; "The Cascade-Correlation Learning Architecture". In: D.S. Touretzky (ed.); *Advances in Neural Information Processing Systems II*, Morgan Kaufmann, San Mateo, pp. 542–532, 1989.

[6] 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)

[7] J. Herz, A. Krogh and R.G. Palmer; *Introduction to the Theory of Neural Computation*. Addison-Wesley, Redwood, CA, 1991.

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

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

[10] D.B. Schwartz, V.K. Samalam, S.A. Solla and J.S. Denker: "Exhaustive Learning". *Neural Computation 2*, pp. 374–385, 1990.

[11] S. A. Solla; "Supervised Learning: A Theoretical Framework". In: M. Casdagli, S. Eubank (eds.), *Nonlinear Modeling and Forecasting, SFI Studies in the Science of Complexity, Proc. Vol. XII*, Addison-Wesley, 1992.

[12] I.G. Sprinkhuizen-Kuyper and E.J.W. Boers; "*The Error Surface of the simplest XOR Network has no local Minima*", to appear.