

## Unimodal Loading Problems

MONICA BIANCHINI<sup>1</sup>, STEFANO FANELLI<sup>2</sup>, MARCO GORI<sup>1</sup>, AND MARCO PROTASI<sup>2</sup>

<sup>1</sup>*Dipartimento di Sistemi e Informatica, Università degli Studi di Firenze*

*Via di Santa Marta, 3 - 50139 Firenze - Italy*

*Tel. +39 (55) 479.6265 - Fax +39 (55) 479.6363*

E-mail: {monica,marco}@mcculloch.ing.unifi.it

<sup>2</sup>*Dipartimento di Matematica, Università di Roma "Tor Vergata"*

*Via della Ricerca Scientifica - 00133 Roma - Italy*

*Tel. +39 (6) 7259.4681 - Fax +39 (6) 7259.4699*

E-mail: {fanelli,protasi}@mat.utorvm.it

This paper deals with optimal learning and provides a unified viewpoint of most significant results in the field. The focus is on the problem of local minima in the cost function that is likely to affect more or less any learning algorithm. We give some intriguing links between optimal learning and the computational complexity of loading problems. We exhibit a computational model such that the solution of all loading problems giving rise to unimodal error functions require the same time, thus suggesting that they belong to the same computational class.

**Keywords:** Backpropagation, computational complexity, optimal learning, premature saturation, spurious and structural local minima, terminal attractor.

### 1 Learning as optimisation

Supervised learning in multilayered networks (MLNs) can be accomplished thanks to Backpropagation (BP), which is used to minimise pattern misclassifications by means of gradient descent for a particular nonlinear least squares fitting problem. Unfortunately, BP is likely to be trapped in local minima and indeed many examples of local extremes have been reported in the literature.

The presence of local minima derives essentially from two different reasons. First, they may arise because of an *unsuitable* joint choice of the functions which defines the network dynamics and the error function. Second, local minima may be inherently related to the structure of the problem at hand. In [5], these two cases have been referred to as *spurious* and *structural* local minima, respectively. Problems of sub-optimal solutions may also arise when learning with *high* initial weights, as a sort of *premature neuron saturation* arises, which is strictly related to the neuron fan-in. An interesting way of facing this problem is to use the "*relative cross-entropy metric*" [10], for which the erroneous saturation of the output neurons does not lead to plateaux, but to very high values of the

cost. When using the cross-entropy metric, the repulsion from such configurations is much more effective, and underflow errors are likely to be avoided.

There have also been attempts to provide theoretical conditions aimed at guaranteeing local minima free error surfaces. So far, however, only some sufficient conditions have been identified that give rise to unimodal error surfaces. Examples are the case of *pyramidal networks* [8], commonly used in pattern recognition, *radial basis function networks* [2], and *non-linear autoassociators* [3]. The identification of similar conditions ensures global optimisation just by using simple gradient descent.

Instead of looking for local algorithms like gradient descent, techniques that guarantee global optimisation may be explored. Of course, one of the main problems to face is that most interesting tasks give rise to the optimisation of functions with even several thousand variables. This makes it very unlikely that most classic approaches [11] can be directly and successfully applied. Instead, the proposal of successful algorithms has to face effectively the *curse of dimensionality* typical of most interesting practical problems. Statistical training methods have been previously proposed in order to alleviate the local convergence problem. These methods introduce noise to connection weights during training, but suffer from extremely slow convergence due to their probabilistic nature.

Several numerical algorithms for global optimisation have also been presented, in which BP is revisited from the viewpoint of dynamical system theory. Barhen *et al.* [1] have proposed the *TRUST* algorithm (for *Terminal Repeller Unconstrained Subenergy Tunneling*) that formulates global optimisation as the solution of a system of deterministic differential equations, where  $E(W)$  is the function to be optimised, while the connection weights are the states of the system. The dynamics used is achieved upon application of the gradient descent to a modified cost which transforms each encountered local minimum into a maximum, so that the gradient descent can escape from it to a lower valley. A related algorithm, called *Magic Hair-Brushing*, has been proposed in [6]. The system dynamics is now modified through a deformation of the gradient field for eliminating the local minima, while preserving the global structure of the function. All these algorithms exhibit a good performance in many practical cases but, unfortunately, their optimal convergence is not formally guaranteed, unless starting from a “good” initial point.

## 2 The class of unimodal loading problems

Most experiments with multilayer perceptrons and BP are performed in a sort of *magic* atmosphere where data are properly supplied to the network which begins learning without knowing whether or not the experiment will be successful either in terms of optimal convergence and generalisation. A *trial and error* scheme is usually employed, aimed at adjusting the architecture in subsequent experiments so as to meet the desired requirements. To some extent, this way of performing experiments is inherently plagued by the suspect that the used numerical optimisation algorithm might fail. Moreover, though optimal learning may be attained with networks having a growing number of hidden neurons [14], the generalisation to new examples is not guaranteed. The intuitive feeling that, in order to obtain a good convergence behaviour, generalisation must be sacrificed, may be effectively formalised in a sort of “*uncertainty principle of learning*” in which the variable representing optimal convergence and generalisation are like conjugate variable in Quantum Mechanics [7]. These potential sources of failure of learning algorithms give

rise to a sort of *suspiciousness* that turns out to be the unpleasant companion of every experiment. This seems to be interwound with the ambitious task of learning too general functions.

Let us focus on the complexity issues related to the loading of the weights independently of the consequent generalisation to new examples. This makes sense once a consistent formulation of the learning problem in terms of both the chosen examples and the neural architecture was provided. We address the problem of establishing the computational requirements of special cases in which the loading of the weights can be expressed in terms of optimisation of unimodal error functions.

### 2.1 Canonical form of gradient descent learning

Let us consider the following learning equation:

$$\frac{dW}{dt} = -\gamma \nabla_W E = f(t, W), \quad (1)$$

where  $E(W)$  is the cost function and  $W \in \mathcal{R}^m$  is the weight vector. Let us choose  $\gamma \doteq \frac{\Psi(E)}{\|\nabla_W E\|^2}$ , being  $\Psi$  a non-negative continuous function of  $E$ . Based on this choice of the learning rate, the dynamics of the error function becomes

$$\frac{dE}{dt} = (\nabla_W E)^T \frac{dW}{dt} = (\nabla_W E)^T \left( -\frac{\Psi(E)}{\|\nabla_W E\|^2} \nabla_W E \right) = -\Psi(E), \quad (2)$$

which makes the cost function continuously decreasing to zero. Those configurations for which  $\nabla_W E = 0$  are singular points that attract the learning trajectory [4].

Special cases of this reduction to a canonical structure, where the learning is forced by function  $\Psi$  and is independent of the problem at hand, have been explored in the literature. White [13] has suggested to introduce a varying learning rate so that the error dynamics evolves following the equation  $\frac{dE}{dt} = -\Psi(E) \doteq -aE$ ,  $a > 0$ , whose solution is a decaying exponential such that reaching the  $E = 0$  *attractor* will theoretically take infinite time. In practice, this may not necessarily be a problem, as the attractor may be approached sufficiently close in a reasonable amount of time, even if, for ill-conditioned systems, it can still be prohibitive to reach a satisfactory solution. Unfortunately, feedforward neural networks do often result in dynamical systems that are ill-conditioned or mathematically *stiff* and thus the convergence is generally very slow.

In [12] the canonical reduction of equation (2) is based on choosing  $\Psi(E) \doteq E^k$ ,  $0 < k < 1$ , which leads to an error dynamics based on the differential equation  $\frac{dE}{dt} = -E^k$ , having a singularity at  $E = 0$  violating the Lipschitz condition. If  $E_0 \geq 0$  is the initial value of the cost, then the closed form solution is  $E(t) = (E_0^{1-k} - (1-k)t)^{\frac{1}{1-k}}$ ,  $t \leq t_e$ , where  $t_e = \frac{E_0^{1-k}}{1-k}$  (Fig. 1a). In the finite time  $t_e$  the transient beginning from  $E(0) = E_0$  reaches the equilibrium point  $E = 0$ , which is a “*terminal attractor*.”

In this paper, we are interested in finding terminal attractors and, particularly, in minimising the time  $t_e$  required to approach the optimal solution. The choice  $\Psi(E) \doteq \eta$  fulfills our needs. Consequently  $t_e = E_0/\eta$  and, in particular, when selecting  $\eta = E_0/\sigma$ , the terminal attractor is approached for  $t_e = \sigma$  (Fig. 1b), independently of the problem at hand, while the corresponding weight updating equation becomes

$$\frac{dW}{dt} = -\frac{E_0}{\sigma} \frac{\nabla E}{\|\nabla E\|^2}. \quad (3)$$

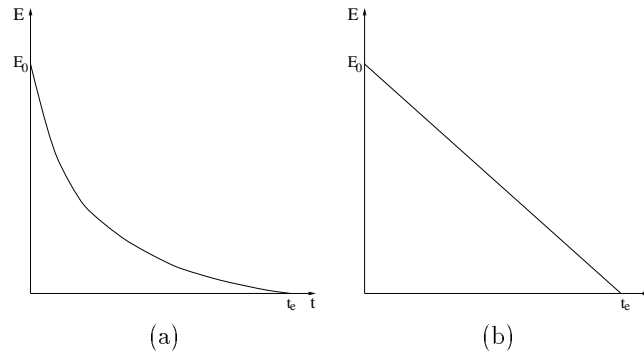


Figure 1: Terminal attraction using (a)  $\Psi(E) \doteq E^k$ ,  $0 < k < 1$ , and (b)  $\Psi(E) \doteq \eta$ .

This way of forcing the dynamics leads to establish intriguing links between the concept of unimodal problems and their computational complexity. In fact, learning attractors in finite time is not only useful from a numerical point of view but, in the light of the considerations on the canonical equations (2), is interesting for the relationships that can be established between different problems giving rise to local minima free cost functions.

## 2.2 Computational analyses

Let us introduce the following classes of loading problems [9].

### Definition 1

A loading problem  $P$  is unimodal,  $P \in \mathcal{UP}$ , provided that there exists an associated unimodal error function  $E(P, W)$  whose optimisation represents the solution of  $P$ .

Note that a given loading problem can be approached in the framework of optimisation using different error functions. For example, the loading of the weights in a multilayer perceptron using linearly-separable patterns may led to sub-optimal solution when choosing error functions where the targets are different from the asymptotical values of the squashing functions. Nevertheless, it is always possible to get rid of these spurious local minima and provide a formulation based on a local minima free error function.

In order to evaluate the computational cost for learning problems belonging to  $\mathcal{UP}$  it is convenient to refer to the parallel computational model offered by differential equation (1). We assume that there exists a continuous system implementing this differential equation and then consider the following computational class.

### Definition 2

Let us consider the class of loading problems  $P$  for which there exists a formulation according to the differential equation (1) such that  $\forall \tau > 0$  the loading of the weights ends in a finite time  $t_e : t_e \leq \tau$ . This class is referred to as the class of finite time loading problems and is denoted by  $\mathcal{FT}$ .

Because of the previous analysis on gradient descent the following result holds.

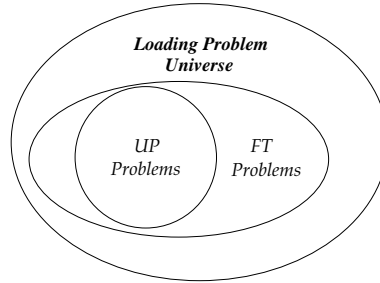


Figure 2: The class of unimodal learning problem can be learned in constant time.

### Theorem 3

$\mathcal{UP} \subset \mathcal{FT}$ .

### Proof

If  $P \in \mathcal{UP}$  then one can always formulate the loading according to differential equation (3) and the gradient descent is guaranteed not to get stuck in local minima. Because of equation (2) the learning process ends for  $t_e = \sigma$ . Hence,  $\forall \tau > 0$ , if we choose  $\sigma \leq \tau$  we conclude that  $P \in \mathcal{FT}$ .  $\square$

This theoretical result should not be overvalued since it is based on a computational model that does not care of problems due to limited energy. When choosing  $\tau$  arbitrarily small, the slope of the energy in Fig. 1b goes in fact to infinite.

One may wonder whether problems can be found in  $\mathcal{FT}$  that are not in  $\mathcal{UP}$  (see Fig. 2). This does not seem easy to establish and is an open problem that we think deserves further attention.

## 3 Conclusions

The presence of local minima does not necessarily imply that a learning algorithm will fail to discover an optimal solution, but we can think of their presence as a boundary beyond which troubles for any learning technique are likely to begin.

In this paper we have proposed a brief review of results dealing with optimal learning, and we have discussed of problems of sub-optimal learning. Most importantly, when referring to a continuous computational model, we have shown that there are some intriguing links between computational complexity and the absence of local minima. Basically all loading problems that can be formulated as the optimisation of unimodal functions are proven to belong to a unique computational class. Note that this class is defined on the basis of computational requirements and, therefore, seems to be of interest independently of the neural network context in which it has been formulated.

We are confident that these theoretical results open the doors for more thoroughly analyses involving discrete computations, that could shed light on the computational complexity based on ordinary models of Computer Science.

### Acknowledgments

We thank P. Frasconi, M. Maggini, F. Scarselli, and F. Schoen for fruitful discussions and suggestions.

### References

- [1] J. Barhen, J. W. Burdick, and B. C. Cetin, Terminal Repeller Unconstrained Subenergy Tunneling (TRUST) for fast global optimization, *Journal of Optimization Theory and Applications*, 77, (1993), 97–126.
- [2] M. Bianchini, P. Frasconi, and M. Gori, Learning without local minima in radial basis function networks, *IEEE Transactions on Neural Networks*, 6, (1995), 749–756.
- [3] M. Bianchini, P. Frasconi, and M. Gori, Learning in multilayered networks used as autoassociators, *IEEE Transactions on Neural Networks*, 6, (1995), 512–515.
- [4] M. Bianchini, M. Gori, and M. Maggini, Does terminal attractor backpropagation guarantee global optimization?, in *International Conference on Artificial Neural Networks*, Springer-Verlag, 1994, pp. 377–380.
- [5] M. Bianchini and M. Gori, Optimal learning in artificial neural networks: a theoretical view. Accepted for publication on *Neurocomputing*.
- [6] J. Chao, W. Ratanasuwana, and S. Tsujii, How to find global minima in finite times of search for multilayer perceptrons training, in *International Joint Conference on Neural Networks*, IEEE Press, Singapore, 1991, pp. 1079–1083.
- [7] P. Frasconi and M. Gori, Multilayered networks and the C-G uncertainty principle, in *SPIE International Conference, Science of Artificial Neural Networks*, Orlando, Florida, 1993, pp. 396–401.
- [8] M. Gori and A. Tesi, On the problem of local minima in backpropagation, *Transactions on Pattern Analysis and Machine Intelligence*, 14, (1992), 76–86.
- [9] J. S. Judd, *Neural Network Design and the Complexity of Learning*. Cambridge, London: The MIT Press, 1990.
- [10] T. Samad, Backpropagation improvements based on heuristic arguments, in *International Joint Conference on Neural Networks*, IEEE Press, Washington DC, 1990, pp. 565–568.
- [11] Torn and Zilinkas, *Global Optimization, Lecture Notes in Computer Sciences*, 1987.
- [12] S. Wang and C. H. Hsu, Terminal attractor learning algorithms for backpropagation neural networks, in *International Joint Conference on Neural Networks*, IEEE Press, Singapore, 1991, pp. 183–189.
- [13] H. White, The learning rate in backpropagation systems: an application of Newton’s method, in *International Joint Conference on Neural Networks*, IEEE Press, Singapore, 1991, pp. 679–684.
- [14] X. Yu, Can backpropagation error surface not have local minima?, *IEEE Transactions on Neural Networks*, 3, (1992), 1019–1020.