Increase Neural Network learning capability by adding hidden layers: an experimental investigation

Pieraut Francis, Caporossi Gilles, Bengio Yoshua

***Abstract*— Ours investigations to understand the reasons why huge neural networks seems to not be able to take advantage of their capacity to get better learning result compare to smaller one bring use to look at optimisation problems related to back-propagation algorithm. One of the solutions we try to reduce those problems was to add hidden layer during the training process. Experimental results showed us that this approach allows a learning process speedup and most important, a better capacity exploitation compared to a static architecture.**

***Index Terms*— Neural Networks, Incremental network architecture, back-propagation optimisation problems.**

I. INTRODUCTION

BEyond a certain capacity, experimental results show us that even if you increase the number of free parameters of a neural network, you won't get much better result on training error. Theoretically, a much higher number of free parameters should allow you to represent much complex function and get better result on training dataset. As mention earlier, it is not what we see in practice so the problem is not that we have not enough capacity but we think that it is related to optimisation problems.

II. BACK-PROPAGATION PROBLEMS

S. Fahlaman and C. Lebiere have already investigate optimisation problems of back-propagation, the most common learning algorithm to train neural networks. They present three limitations of back-propagation: *the step size problem*, *the moving target problem*, *the attenuation and dillution of error signal* as it is propagates backward through the layers of the network. Their solution to reduce those problems was the Cascade-Correlation, a different architecture and learning algorithm. Instead of just adjusting the weights on a neural network of fixed architecture, Cascade-Correlation begin with a minimum network, then automatically trains and adds new hidden units one by one, creating a multi layer structure.

In our investigation, we identify three others limitations: *the opposite gradient problem*, *the non-existence of specialisation parameters mechanism* and the *symmetry problem*. For convenience reasons, we gather these six problems under the term of *back-propagation problems*.

III. ADDING HIDDEN LAYER

One of the solution to increase learning capability of neural network was to add a complete hidden layer during training process. This mechanism should reduce *back-propagation problems* and involve perhaps an increase of learning speed and learning capability.

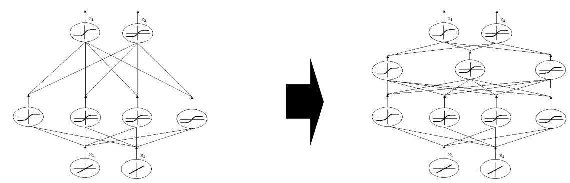

Fig. 1. Adding hidden layer

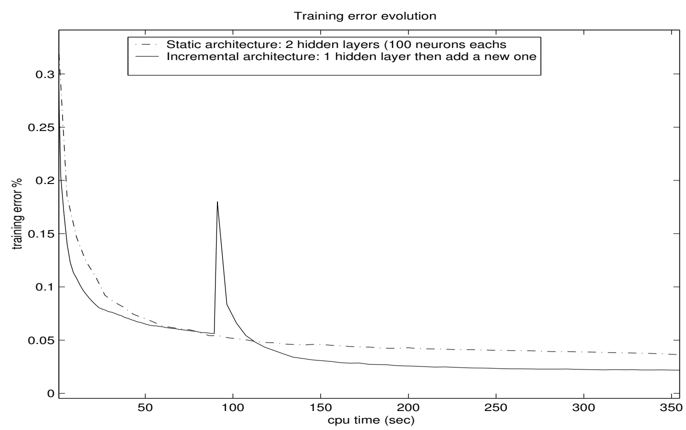

Fig. 2. Static architecture vs incremental architecture

IV. EXPERIMENTAL RESULTS

We compare a neural network with fixed architecture to incremental architecture by adding hidden layer on a letters recognition problems. At the end, the two networks have the same topology. First, we find the best learning rates for a neural network with two hidden layers to get the best training error value. Each layer got 100 hidden neurons. Then we train a one hidden neural network until error seem to stuck and we add a new hidden layer as it is draw on figure 1.

As you can see on figure 2, for the same topology at the end, adding hidden layer speedup the learning process but more important, it increase the learning capability.

V. CONCLUSION

Our experimental investigation show that adding hidden layer can speedup learning process and increase learning capability of neural network compare to a fixed architecture. We are convinced that optimisation problems related to back-propagation can explain this behavior. By starting with a smaller problem, we reduce back-propagation problems and over all, optimisation is easier.