Calibration and Reliability in Groundwater Modelling (Proceedings of the ModelCARE 99 Conference held at Zürich, Switzerland, September 1999). IAHS Publ. no. 265, 2000.

Automatic model calibration using a universal approximator

PAUL TORFS & GEORGE BIER
Subdepartment of Water Resources, Wageningen University, Nieuwe kanaal 11, 6709 PA Wageningen, The Netherlands
e-mail: paul.torfs@users.whh.wau.nl

Abstract To solve the inverse problem of groundwater flow models the fields of hydraulic properties (e.g. transmissivities) have to be parameterized. The number of parameters should be kept relatively low. Usually this is done with zonation techniques where each zone has constant hydraulic properties. In this contribution, a universal approximator, a neural network (NN) consisting of a relatively small number of neurons, represents the transmissivity field in a groundwater flow model. The objective of the calibration procedure is to minimize the squared error by adjusting the weights in the NN. Our approach does not require a predefined zonation or any other restricting measure in describing the transmissivity field. It will be illustrated with a hypothetical groundwater flow system similar to the Carrera and Neuman example.

INTRODUCTION

To solve the inverse problem of groundwater flow models, the fields of hydraulic properties (e.g. transmissivity or hydraulic conductivity) of the model need to be parameterized. However, the number of parameters should be kept relatively low. This problem is usually solved by zonation techniques where in each zone the hydraulic properties are described by a few parameters only.

In this article, the new technique of neural networks is introduced as an alternative to this kind of parameterization. It will be shown to work in the case of a classical example of inverse modelling.

NEURAL NETWORKS

Neural networks (NN) is a relatively new modelling technique that has found application in many branches. They are used for many types of black box modelling (for an example with time series see Masters, 1995a), and pattern recognition (Bishop, 1998).

Although there are many flavours of NN (Bishop, 1998), we will consider only one specific type: the feed forward NN. Figure 1 illustrates the working of a single neuron.

First, there are *inputs*. In both examples, there are two inputs named x and y, varying between -1.0 and $+1.0$. In general, any number of inputs of any kind is allowed. The first computational step consists of calculating a weighted sum of the inputs. For the top case, the weights are -10.4 and $+11.4$, and the weighted sum of the inputs is then $-10.4x + 11.4y$. This weighted sum is then transformed by a *nonlinear* function, often called an activation function. A typical activation function is tanh. The result (z) of this is called the *output* of the neuron. In this example:

$$z = \tanh(-10.4x + 11.4y)$$

The right top part of Fig. 1 shows the surface generated by this formula. This is clearly a nonlinear function. Moreover, because the weights were rather high, more of the "+1" and "–1" values of the activation function were selected, resulting in a surface that approximates a step function.

In the bottom part of the figure, the weights chosen are small, resulting in an almost linear surface. This shows that the weights control the functioning of a neuron, making linear as well as nonlinear behaviour possible.

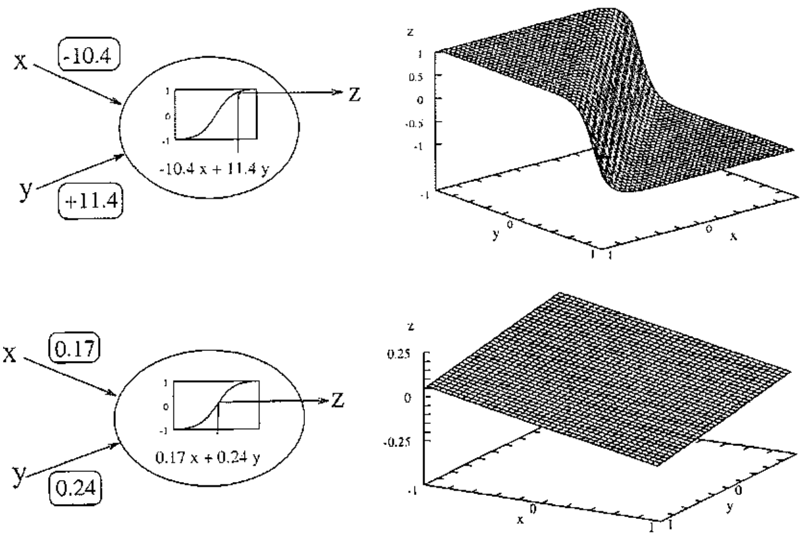

Fig. 1 Single neuron example.

Figure 2 shows typical NN with several connected neurons. In the top part, neuron 1 and neuron 2 are respectively the top and bottom of the examples of Fig. 1. The outputs of these two neurons are input to a third neuron. The right part shows the surface generated by this NN model. It is clearly a combination of the surfaces of Fig. 1.

The bottom part of Fig. 2 shows a more typical feed forward NN. It has two inputs and one output. Between the input and output layer are two layers of hidden neurons. Each connection in the figure has its own weight (not shown).

The bottom right part of Fig. 2 illustrates that a NN with sufficient complexity can produce surfaces of an arbitrary shape.

It has been shown that, after normalization of the z-values to the range [–1:1], *any* surface can be approximated by a NN with two hidden layers with sufficient neurons (Lipmann, 1987; Lapades & Faber, 1988), even error bounds can be derived (Barron, 1993). The user of a NN has to specify the complexity of the NN, expressed in the number of hidden layers (usually two), the number of neurons in each layer and the weights for each connection. These weights are the parameters of the NN. Usually they are determined by an optimization procedure. A set of weights is certainly not unique.

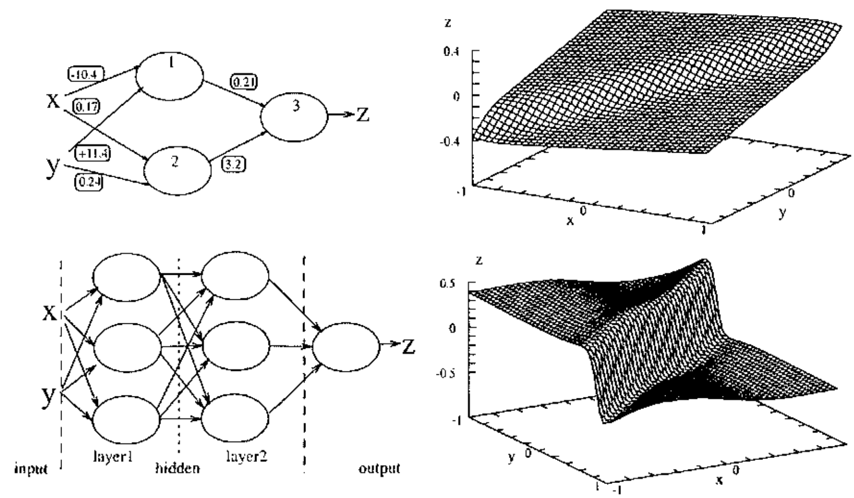

Fig. 2 Neural networks consisting of several neurons.

INVERSE MODELLING OF TRANSMISSIVITY FIELDS

In this contribution, we will only consider a rather simple inverse problem; a stationary one-layer groundwater flow model, without any prior knowledge. The only unknown is the transmissivity field T and the observations are all piezometric heads $hobs_1,\ldots,hobs_M$. The inverse problem can then be formulated as to adjust T such that:

$$\chi^2 = \sum_{i=1}^{M}(hobs_i - hmod_i(T))^2$$

is minimal. To solve this inverse problem, one has to use a finite parameterization of the transmissivity field T. A common way to do this is by zonation:

$$x, y \in D_i \rightarrow T(x,y) = T_i$$

Then, the optimal values for $T_1,\ldots,T_K$ can be found by an automatic calibration procedure. The choice of the number of zones and their shape is left to the modeller, introducing subjectivity into the whole procedure. Moreover, as shown in (Carrera & Neuman, 1986), this choice has a major influence on the final result.

Here we propose a more objective method, namely to model the transmissivity field using a NN. As shown above, NN are flexible enough to model any shape of transmissivity field. The modeller prescribes the number of hidden layers and neurons. An automatic calibration procedure is then used to determine the optimal values of the weights of the NN (Fig. 3).

THE OPTIMIZATION PROBLEM

Finding the optimum values of the weights of a NN is a well-known difficult problem, due to the fact that there are many local minima. Many different optimization

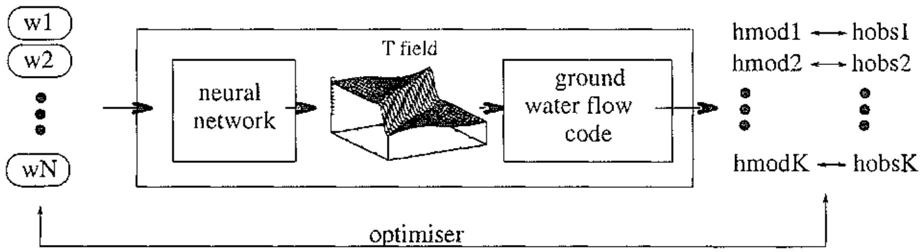

Fig. 3 The modelling and optimization procedure.

techniques have been proposed to solve this problem (Masters, 1995b). One classical iterative technique that uses the structure of a NN is *back-propagation* (errors are propagated from the final layer backwards to the first, adjusting the weights accordingly). In this approach, a random initial choice for the weights is made. The user stops the iterative optimization technique when a given criterion is met. Because of the subjective aspects of this procedure, more rigid procedures were used. Most other optimization techniques in the field of NN have two components: a local one to approximate a local minimum, and a global one to find another starting point for the local minimum search.

For our first attempts we used PEST (Doherty, 1998), implementing a Levenberg-Marquardt algorithm as a local technique, and a pure random choice as a global one. The finite element groundwater flow program we used was MICROFEM (Hemker & Nijsten, 1997). We combined all components through a batch file. After that, we developed a program that combined a finite element flow code, a simulated annealing technique (for the global search) and a Levenberg-Marquardt algorithm (for the local search). Several mixtures of both techniques were tested.

THE CARRERA NEUMAN EXAMPLE

To illustrate this new approach in parameterizing hydraulic properties, a synthetic flow problem as in Carrera & Neuman (1986) will be used. Figure 4 summarizes the problem. In Carrera & Neuman (1986) zonation techniques were used to inverse model the transmissivity. Several different zonations were tested. The zonation, which resembles the true transmissivity zonation, performed best. The models tested with alternative zones negatively influenced the resulting estimation of the parameters.

In our approach, we choose to model the transmissivity by a 2–4–4–1 NN. An optimization algorithm determined the optimal values of the 37 weights. Several sets of "good choices" were found.

Figure 5 shows two typical results. The two pictures on the left form the first result. The top picture shows the initial value of the transmissivity field, and the bottom picture the final field, after optimization. The final misfit was $\chi^2 = 0.017$ m^2. This last picture should be compared to the right upper one of Fig. 4. Clearly, the NN was capable of approximating this difficult non-continuous field. The two pictures on the right form a second example. The starting field (top picture) was chosen differently

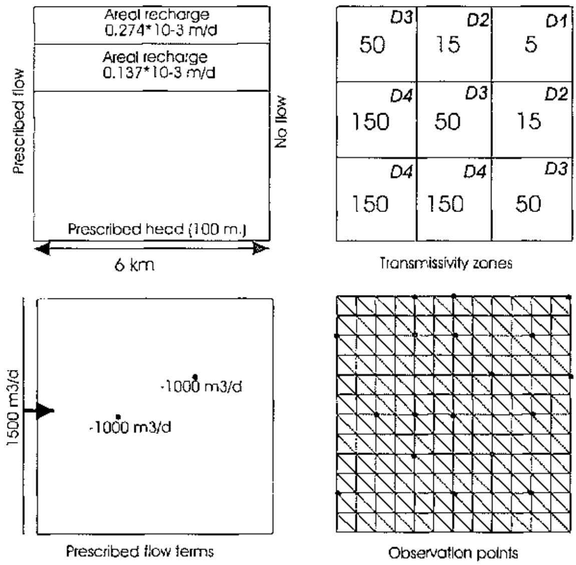

Fig. 4 The Carrera Neuman example.

from the first example. After optimization, this resulted in a different field (bottom picture) with a misfit of χ^2= 0.025 m^2. This clearly shows *equifinality* or non-uniqueness of the inverse modelling problem: many different fields exist with almost the same χ^2 value. A NN can help in investigating this, giving the modeller feedback on the uncertainty of the determined field.

Similar insight can of course also be obtained by using a classical approach by zonation techniques, as shown in Carrera & Neuman (1986). In that case however, the modeller has to create himself a bandwidth of variation by choosing different zonations. In a NN, randomly chosen starting weights generate different starting fields (Fig. 5 shows two of these). This is a less subjective procedure for generating this bandwidth.

CONCLUSIONS AND DISCUSSION

It seems that this new technique at combining a NN with a groundwater flow model to calibrate hydraulic properties looks promising.

- With relatively few parameters, one can describe the flow domain with a large amount of flexibility for gradually varying or abruptly changing hydraulic properties.
- The only *a priori* choice to be made is the number of hidden neurons. Zonation is not needed.
- The technique produced acceptable results in the case of the classical Carrera-Neuman example.
- It confronts the user in an objective way with the non-uniqueness of the inverse problem.

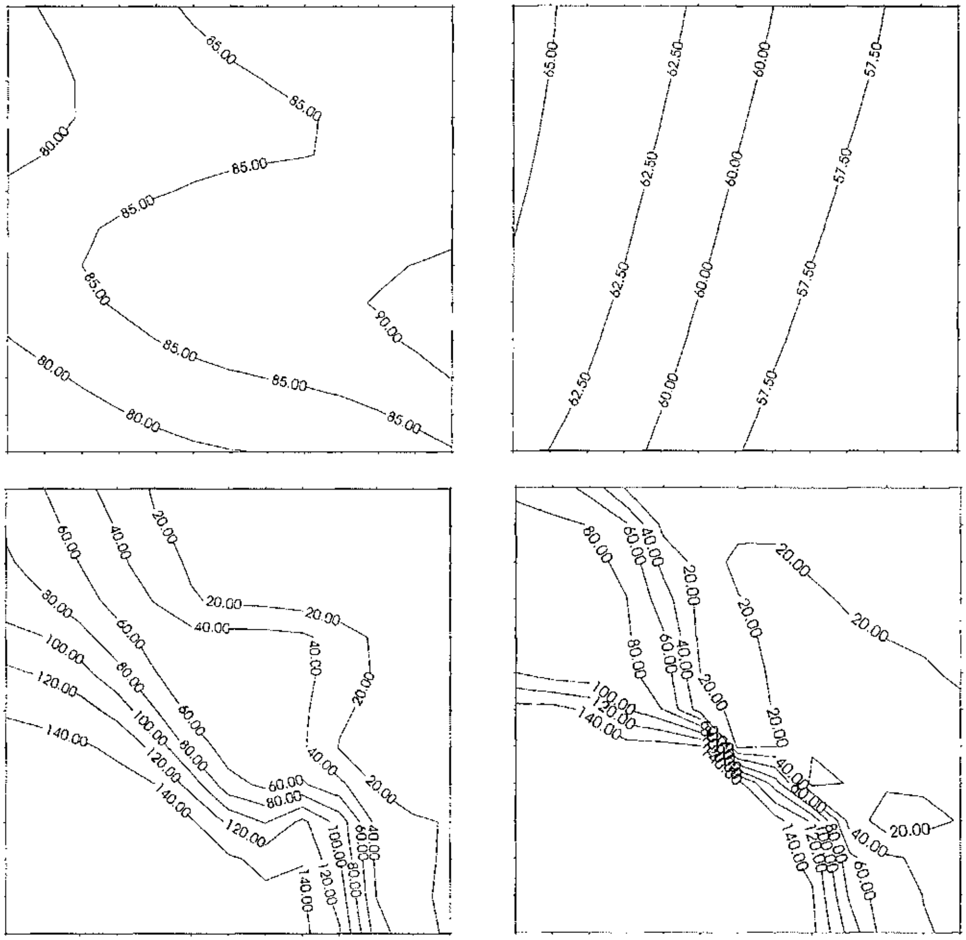

Fig. 5 Results of two (left and right) typical runs. Top figures show initial transmissivity fields, bottom figures optimized fields. The sum of squared errors for the initial left run is 4766 m^2, for the right run 3287 m^2, and for the optimized fields 0.017 m^2 and 0.025 m^2 respectively.

More work has still to be done:

- to effectively solve the optimization problem, inspecting many local minima,
- to incorporate other criteria or prior knowledge into the construction,
- to optimize several fields of hydraulic properties simultaneously.

REFERENCES

Barron, A. R. (1993) Universal approximation bounds for superposition of a sigmoidal function. *IEEE Trans. Information Theory* **39**, 930–945.

Bishop, C. M. (1998) *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, UK.

Carrera, J. & Neuman, S. P. (1986) Estimation of aquifer parameters under transient and steady state conditions. *Wat. Resour. Res.* **22**(2), 199–242.

Chen, A. M., Lu, H., Hecht-Nielsen, R. (1993) On the geometry of feedforward neural network error surfaces. *Neural Computation* 5, 910–927.

Doherty, J. (1998) *PEST: Model Independent Parameter Estimation Software Package.* Watermark Computing, Brisbane, Australia.

Hemker, C. J. & Nijsten, G. J. (1997) *Groundwater Flow Modelling Using MicroFEM version 3.* Amsterdam.

Lapedes, A. & Farber, R. (1988) How Neural Nets work. In: *Neural Information Processing Systems* (ed. by D. Z. Anderson), 442–456. :American Institute of Physics, New York.

Lippmann, R. P. (1987) An introduction to computing with Neural Nets. *IEEE ASSP Magazine*, 4–22.

Masters, T. (1995a) *Neural Novel and Hybrid Algorithms for Time Series Prediction.* John Wiley & Sons, Chichester, UK.

Masters, T. (1995b) *Advanced Algorithms for Neural Networks.* John Wiley & Sons, Chichester, UK.