DTU Library

Optimal Brain Surgeon on Artificial Neural Networks in

Christiansen, Niels Hørbye; Job, Jonas Hultmann; Klyver, Katrine; Høgsberg, Jan Becker

Link back to DTU Orbit

Optimal Brain Surgeon on Artificial Neural Networks in Nonlinear Structural Dynamics

Niels H. Christiansen[1,2], Jonas Hultmann Job[3], Katrine Klyver[4], and Jan Høgsbrg[1]

[1]Department of Mechanical Engineering, Technical University of Denmark, Kgs. Lyngby, Denmark
[2]DNV Denmark, Hellerup, Denmark
[3]Rambøll Danmark - Offshore Wind, Copenhagen, Denmark
[4]NOV Flexibles, Broendby, Denmark

Summary. It is shown how the procedure know as optimal brain surgeon can be used to trim and optimize artificial neural networks in nonlinear structural dynamics. Beside optimizing the neural network, and thereby minimizing computational cost in simulation, the surgery procedure can also serve as a quick input parameter study based on one simulation only.

Key words: Nonlinear structural dynamics, Artificial neural networks, Optimal brain surgeon.

Introduction

Time domain simulation of nonlinear systems using finite element method (FEM) analysis can be computationally very expensive. One method that have shown to be very time saving is the use of artificial neural networks (ANN). In the literature it has been shown on various types of structures how this method can reduce calculation time by up to two orders of magnitude [1, 2]. However, a major problem with this method is the difficulty of developing a robust algorithm that automatically generates an optimized ANN for a given structure. In this paper it is shown how a trained ANN can be further optimized and reduced in size by the procedure known as optimal brain surgeon (OBS). This procedure prunes the least salient network weights in an ANN. Besides trimming the network this operation can be used as a quick ranking of the importance of the various inputs that go into the analysis. Dynamic response of structures often depends on several external loads. By applying the OBS procedure to a trained ANN it is possible to determine which of these external force components that play the least significant role and also evaluate if one or more components can be ignored completely in the analysis. Hence, the OBS procedure can serve as an input parameter study based on one simulation only.

Artificial neural networks and optimal brain surgeon

Artificial neural networks are in principal mathematical models that replicate the human brain's impressive ability to do high speed pattern recognition. They serve as tools which perform nonlinear mapping between a given input and a corresponding output without time consuming equilibrium iterations. There exists vast literature on the fundamentals of neural networks and the philosophy behind. Since the ANN is not the main object in this paper it will be left to the reader to consuls e.g. [3] for further information on that topic. Here it will just be mentioned that an ANN with one input layer, one hidden layer with tangent hyperbolic as activation
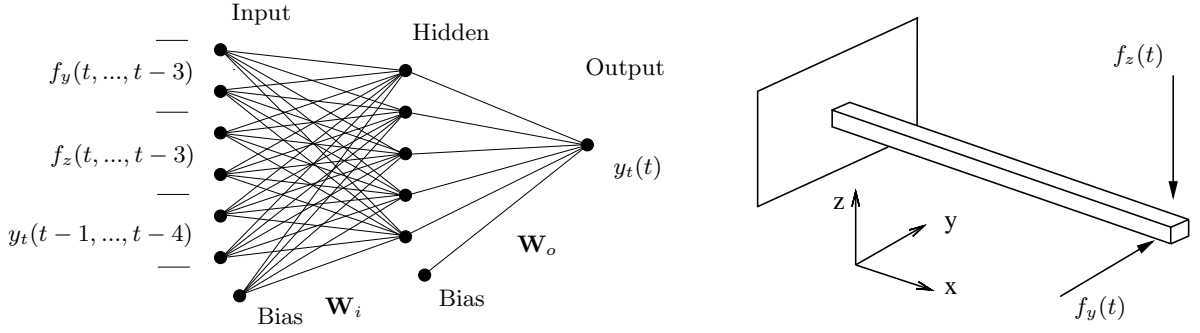
Figure 1. Left: ANN with one hidden layer. Right: Model of cantilevered beam.

function and a liner output layer as sketched in the left part of figure 1 is used in the following study.

OBS is a procedure that prunes the least salient network weights in a trained ANN. When training an ANN the first derivative of the error function is used to find the direction in which to step in weight space in order to minimize the error. The OBS procedure uses the second derivative to find the curvature of the error function in weight space. That information can be used to arrange all network weights and to find out which weight is least salient and thereby 'cheapest' to delete [4]. The saliency of a weight q is calculated by

$$L_q = \frac{1}{2} \frac{w_q^2}{[\mathbf{H}^{-1}]_{qq}}, \tag{1}$$

where $\mathbf{H}^{-1}$ is the inverse of the Hessian matrix ($\mathbf{H} \equiv \partial^2 E/\partial \mathbf{w}^2$) containing all second order derivatives. Having found the least salient weight we can update the rest of the weights using

$$\delta \mathbf{w} = -\frac{w_q}{[\mathbf{H}^{-1}]_{qq}} \mathbf{H}^{-1} \cdot \mathbf{e}_q, \tag{2}$$

where $\mathbf{e}_q$ is the unit vector in weight space. The inverse Hessian is calculated directly using a slightly modified version of the Sherman-Morrison formula (3) and a single sequential pass through the training data $1 \le m \le P$.

$$\mathbf{H}_{m+1}^{-1} = \mathbf{H}_m^{-1} - \frac{\mathbf{H}_m^{-1} \mathbf{X}_{m+1} \cdot \mathbf{X}_{m+1}^T \cdot \mathbf{H}_m^{-1}}{P + \mathbf{X}_{m+1}^T \mathbf{H}_m^{-1} \mathbf{X}_{m+1}}, \tag{3}$$

where P is the number of training data sets.

So the procedure is to take a network that has been trained to a local minimum, calculate $\mathbf{H}^{-1}$ using (3), find the network weight that gives the smallest saliency using (1), and determine the adjustment of the network weights (2). This procedure can in principal be repeated until all weights are deleted. However, after each deletion of the cheapest weight one can evaluate the increase in error and see if it exceeds an acceptable level. In that case the procedure must be stopped.

Structural model

To demonstrate the OBS method a 3D FEM model of a cantilevered beam is set up. The beam has the properties listed in table 1 and is shown to the right in figure 1. The model uses the co-rotational beam element formulation as described in [5]. Hence, it is a nonlinear model able to handle large rotations and deflections. The beam model is divided into five elements of equal length and is damped through Rayleigh damping. The beam is subject to two stochastic load histories working in two different directions at the beam tip. The two load histories are

l	E	I_y	I_z	A	ρ
10 m	$1 \cdot 10^5$ MPa	$1 \cdot 10^4 \ mm^4$	$1 \cdot 10^5 \ mm^4$	$1 \cdot 10^5 \ mm^2$	$1 \cdot 10^5 \ kg/m^3$

Table 1. Beam properties

different but have equal characteristics in terms of mean value, standard deviation and dominant frequency. The load has a mean of 2.7 kN with a standard deviation of 0.3 kN and a dominating frequency around 0.7 Hz. The response of the beam is calculated by Newmark's method of direct integration. The Newton-Raphson method is used to achieve force equilibrium in each time step, i.e. to update system matrices in accordance with the procedures described in [5]. Time step size is 0.1 sec. and equilibrium is assumed when force and displacement residuals are below 1×10^{-6}.

Data generated by the FEM is used to train an ANN to predict the horizontal response $y_t(t)$ of the beam tip. A 400 sec. simulated response history is divided into 3 sets. The first set is used for training the ANN, the second set is used for testing and the last set is saved to demonstrate the accuracy of the pruned ANN on new data. The ANN has 40 neurons in the hidden layer and 13 neurons in the input layer. The network input vector consists of the four previous time steps of the tip response and the two loads, as indicated in figure 1. This gives a $[13 \times 40]$ weight matrix $\mathbf{W}_i$ between input and hidden layer and a $[40 \times 1]$ weight matrix $\mathbf{W}_o$ between hidden and output layer.

One step in the OBS procedure is to evaluate the network error each time a network weight is deleted. Depending on the structure and the type of analysis carried out one must decide an error increase tolerance for when to stop the deletion of weights. Figure 2 shows how the network error increases with the number of deleted weights. It is seen that it is possible to delete about half the weights without increasing the network error considerably.
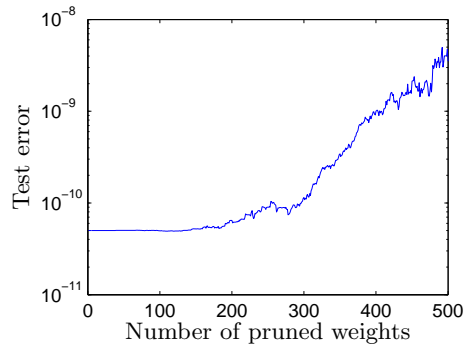
Figure 2. Test error vs. number of pruned weights.

Figure 3 shows response histories calculated by the FEM model and the pruned ANN. It is seen that even though almost half of the network connections have been deleted the ANN still predicts the horizontal deflection very accurately. Note that the ANN in this case simulates about two orders of magnitude faster than the FEM. If we examine the input weight matrix $\mathbf{W}_i$ containing all connection weights between the input layer and the hidden layer it is possible to do some interpretation of the importance of the input variables.

As the procedure deletes the least important weights and since the loads on the beam are arranged in the network input vector it is possible to evaluate the importance of the loads and whether one or more inputs can be neglected. The pruned weight matrix $\mathbf{W}_{i,prune}$ is written below. As actual values of individual weights are unimportant for this evaluation of the pruned matrix positive and negative weights are marked by $+/-$ while deleted weights are denoted 0. It is seen that the load in the $z-$direction is almost completely ruled out by the OBS and thereby

Figure 3. Comparison of calculated response after pruning of 250 network weights.

possibly can be completely neglected in the response simulation. This is as one intuitively would expect since f_z works vertically and the analysis considers the horizontal deflection.

It should be noted that two runs with the OBS procedure will not give identical pruned weight matrices. This is due to the fact that the ANN is initiated with random weights before training. However, the simulation will still be as accurate as the one shown in figure 3 and the conclusions drawn from the interpretation of $\mathbf{W}_{i,pruned}$ will be the same despite some minor changes in the matrix.

$$
\mathbf{W}_{i,pruned} \quad \Rightarrow \quad
\begin{bmatrix}
+ & - & + & - & - & 0 & - & - & 0 & - & - & - & + & - & 0 & - & 0 & - & + & + & + & + & - & - & 0 & + & + & - & - & + & - & 0 & - & + & 0 & + & - & 0 & - & - \\
+ & 0 & + & - & 0 & + & + & 0 & + & - & - & 0 & 0 & 0 & + & + & 0 & + & 0 & 0 & 0 & 0 & + & 0 & - & 0 & - & 0 & - & + & 0 & 0 & + & 0 & - & 0 & 0 & - & 0 & 0 \\
+ & - & + & - & - & - & - & - & - & - & - & + & - & - & 0 & 0 & + & + & + & + & - & - & + & 0 & + & - & - & 0 & - & 0 & - & + & + & + & - & + & - & - \\
0 & - & + & 0 & - & - & - & - & - & - & + & - & - & 0 & + & + & + & 0 & + & - & - & 0 & 0 & 0 & - & 0 & 0 & 0 & 0 & - & + & + & + & - & + & - & 0 \\
0 & 0 & 0 & 0 & 0 & + & 0 & 0 & 0 & 0 & - & 0 & - & + & 0 & 0 & - & 0 & 0 & 0 & + & 0 & 0 & 0 & 0 & 0 & - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & + & - & + & + & 0 & 0 & 0 & + & - & + & - & 0 & 0 & - & 0 & - & 0 & - & 0 & + & 0 & 0 & + & + & 0 & 0 & 0 & 0 & 0 & 0 & + & - & 0 & + & + & + \\
0 & + & 0 & 0 & 0 & + & + & 0 & + & 0 & 0 & 0 & 0 & + & + & 0 & + & - & 0 & 0 & 0 & + & + & - & 0 & - & 0 & 0 & 0 & 0 & + & 0 & - & - & 0 & - & + & 0 \\
0 & - & + & 0 & 0 & - & - & - & 0 & 0 & 0 & + & - & - & 0 & - & + & + & 0 & + & - & + & 0 & + & 0 & 0 & 0 & 0 & 0 & - & 0 & + & + & - & + & - \\
0 & + & 0 & 0 & + & + & + & + & 0 & 0 & + & - & + & + & + & 0 & + & - & - & - & - & + & + & - & 0 & - & + & 0 & 0 & + & 0 & + & - & - & - & + & - & + & + \\
+ & - & + & - & - & - & - & - & - & + & - & - & - & + & + & + & + & - & + & + & + & - & - & + & - & - & + & + & + & - & + & - & - \\
0 & - & 0 & 0 & 0 & - & - & 0 & - & 0 & 0 & 0 & 0 & 0 & - & - & 0 & - & 0 & 0 & 0 & 0 & - & - & + & 0 & + & 0 & 0 & 0 & 0 & 0 & - & 0 & + & + & - & + & 0 & 0
\end{bmatrix}
\begin{array}{l}
\left.\vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right\} f_y \\[6pt]
\left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} f_z \\[6pt]
\left.\vphantom{\begin{matrix}a\\a\\a\end{matrix}}\right\} y_t \\[6pt]
\left.\vphantom{a}\right\} Bias
\end{array}
$$

Concluding remarks

The example presented in this paper demonstrates that an trained ANN can be trimmed and reduced considerably in size by the OBS procedure. The above analysis was carried out on a very simple structure and with simple load patterns. However, it may be possible to use this method on more complex structures subject to more complicated loads. In that case the OBS procedure can serve as a quick input parameter study that can help reducing the computational cost on nonlinear simulations.

References

[1] R. Guarize, N.A.F. Matos, L.V.S. Sagrilo, and E.C.P. Lima. Neural networks in dynamic response analysis of slender marine structures. *Applied Ocean Research*, (29):191–198, January 2008.

[2] N. H. Christiansen, J. Høgsberg, and O. Winter. Artificial neural networks for nonlinear dynamic response simulation in mechanical systems. *Nordic Seminar on Computational Mechanics*, (24):77–80, November 2011.

[3] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, October 2006.

[4] B. Hassibi, D.G. Stork, and G.J. Wolff. Optimal brain damage and general network pruning. *1993 IEEE International Conference on Neural Networks, vols 1-3, 293-299*, 1993.

[5] Steen Krenk. *Non-linear Modeling and Analysis of Solids and Structures*. Cambridge University Press, August 2009.