Technical University of Denmark

Artificial Neural Networks for Nonlinear Dynamic Response Simulation in Mechanical Systems

Christiansen, Niels Hørbye; Høgsberg, Jan Becker; Winther, Ole

Published in:
Proceedings of the 24th Nordic Seminar on Computational Mechanics

Publication date:
2011

Proceedings of the 24$^{\text{th}}$ Nordic Seminar on Computational Mechanics
J. Freund and R. Kouhia (Eds.)

Artificial Neural Networks for Nonlinear Dynamic Response Simulation in Mechanical Systems

Niels H. Christiansen[1,2], Jan Høgsberg[1], and Ole Winther[3]

[(1)]Department of Mechanical Engineering, Technical University of Denmark, Kgs. Lyngby, Denmark, e-mail: Niels.Christiansen@dnv.com
[(2)]DNV Denmark, Hellerup, Denmark
[(3)]Department of Informatics and Mathematical Modeling, Technical University of Denmark, Kgs. Lyngby, Denmark

Summary. It is shown how artificial neural networks can be trained to predict dynamic response of a simple nonlinear structure. Data generated using a nonlinear finite element model of a simplified wind turbine is used to train a one layer artificial neural network. When trained properly the network is able to perform accurate response prediction much faster than the corresponding finite element model. Initial result indicate a reduction in cpu time by two orders of magnitude.

Key words: Nonlinear structural dynamics, Artificial neural networks.

Introduction

Time domain simulation of nonlinear systems using finite element method (FEM) analysis can be computationally very expensive - especially in fatigue calculations where long response histories are needed in order to obtain reliable time series statistics. The use of artificial neural networks (ANN) combined with classical methods have shown promising results in reducing this computational cost [1]. This paper presents a hybrid method for simulation of dynamic response of a simple nonlinear structure. It is shown how an ANN can be trained to predict dynamic response of a simplified model of a wind turbine. FEM models of nonlinear structures often require fine element mesh discretization, small time step size and iterative procedures in order to obtain equilibrium between internal and external forces. For large complex models this can be very time consuming. The ANN's ability to perform nonlinear mapping between a given input and a system output makes it capable of response prediction without equilibrium iterations. Hence, a properly trained ANN can save a lot of computational effort in response prediction.

Artificial Neural Network

The architecture of a typical one layer artificial neural network is shown in Figure 1. The ANN consists of an input layer, a hidden layer and an output layer. Each connection between two neurons in two neighboring layers has a weight. Training of an ANN is optimization of these weights for a given data training set.

Following [2] the ANN set up and training procedure can be written as follows. The ANN output is calculated by

$$\mathbf{y} = \mathbf{W}_o^\top \mathbf{z}, \qquad \mathbf{z} = \tanh\left(\mathbf{W}_i^\top \mathbf{x}\right), \qquad x_0 \equiv z_0 \equiv 1, \tag{1}$$

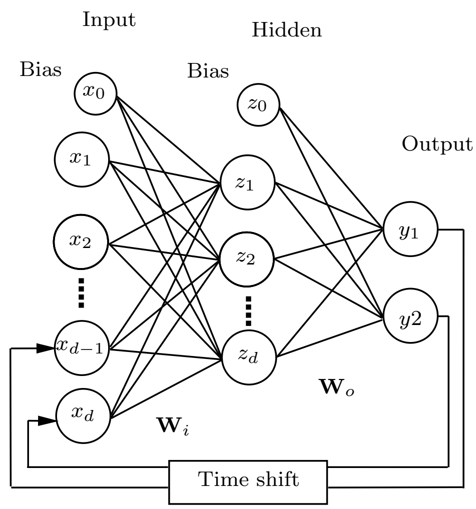

Figure 1. Sketch of artificial neural network.

where $\mathbf{x}$ is input vector, $\mathbf{y}$ is output vector and $\mathbf{W}_i$ and $\mathbf{W}_o$ are neuron connection weights between input and hidden layer and hidden and output layer, respectively. The tangent hyperbolic is used as activation function between input and hidden layer.

The error function which is minimized during training can be written as

$$E(\mathbf{W}) = \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{c}\{y(\mathbf{x}^n;\mathbf{W})_k - t_k^n\}^2 + \frac{1}{2}\alpha\mathbf{W}^2, \tag{2}$$

where y is the ANN output, t is the target value and α is the weight decay that controls the value of the weights and prevents the ANN from overfiting to noise in the training data. Optimal weights are found with an iterative procedure stepping in weight space towards minimal error. The weight update is done by gradient decent going the opposite direction of the cost functions gradient as in (3).

$$\mathbf{W}_{new} = \mathbf{W}_{old} + \Delta\mathbf{W}, \qquad \Delta\mathbf{W} = -\eta\frac{\partial E(\mathbf{W})}{\partial\mathbf{W}}, \tag{3}$$

where η is the learning step size parameter. This parameter can either be constant or updated during the training of the ANN. For this application the dynamic learning step size parameter is adjusted for each iteration so that it is increased if the training error is decreased compared to previous iteration step and reduced if the training error increases.

Structural Model

To illustrate the hybrid method a simplified model of a wind turbine is set up, see Figure 2. The height of the wind turbine is 100 m. The diameter of the turbine steel tower is 4 m with a wall thickness of 0.1 m. At the top of the tower a $100 \cdot 10^3$ kg mass is placed to represent the nacelle and turbine blades. The lowest eigenfrequency of the structure is 0.3 Hz. The load $\mathbf{f}(t)$ applied to the structure corresponds to the horizontal load on a 100 m diameter rotor in a 15 m/s wind. The mean wind load is 284 kN with a standard deviation of 44 kN.

The FEM model of the structure uses co-rotational beam element formulation as described in [3]. This formulation separates beam motion into two parts: a rigid body motion associated with a local frame of reference, and a deformation of the beam within this frame. The local deformation of the beam element implies geometrical stiffness contributions which depend on the deformation. Thus, we have a nonlinear model which can handle large deformations, when the governing equation is given as

$$\mathbf{M}\ddot{\mathbf{r}} + \mathbf{C}\dot{\mathbf{r}} + \mathbf{K}(\mathbf{r})\mathbf{r} = \mathbf{f}(t). \tag{4}$$

The lumped masses and beam stiffness contributions are assembled in a mass matrix $\mathbf{M}$ and a stiffness matrix $\mathbf{K}$, respectively. The beam is Rayleigh damped through damping matrix $\mathbf{C}$ introducing 3 % damping to the lowest vibration mode of the wind turbine. The force vector $\mathbf{f}$ consists of external force components at each degree of freedom (DOF) for each time step and the vector $\mathbf{r}$ contains the degrees of freedom (DOF) of the structure. Note that the components in the stiffness matrix $\mathbf{K}$ dependent on the deflection $\mathbf{r}$ of the structure. In the FEM model the turbine tower is divided into 10 elements.

The response of the structure is calculated by Newmark's method of direct integration. The Newton-Raphson method is used to achieve force equilibrium in each time step and the update of the stiffness matrix follows the procedures described in [3]. The time step size is 0.1 s and equilibrium is assumed when force and displacement residuals are below 1 %.

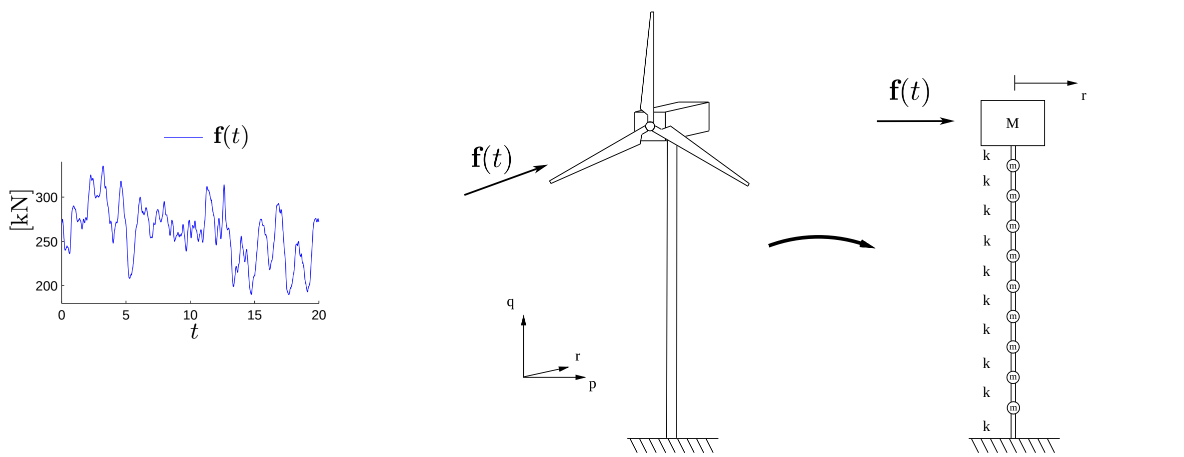

Figure 2. a) Harmonic wind load, b) Sketch of wind turbine, c) Simple turbine FEM model.

Based on the response history data generated by the FEM model an ANN is trained to predict the future dynamic response of the wind turbine.

Simulation

The ANN is designed and trained to predict the horizontal deflection of the wind turbine at the location of the mass M, see Figure 2. Note that the ANN output only gives the horizontal response of top and hence, as oppose to the FEM model, not the response of all model DOFs. The load together with the state space variables $(r, \dot{r})$ of previous time steps are used as ANN input

$$\mathbf{x}_t = [f(t) \ldots f(t-d) \quad r(t-1) \quad \dot{r}(t-1) \ldots r(t-d) \quad \dot{r}(t-d)]^T, \tag{5}$$

where d is number of previous time steps included in the input i.e. the model memory. ANN output is the current deflection and velocity

$$\mathbf{y}_t = [r(t) \quad \dot{r}(t)]^T \tag{6}$$

Since the ANN in this case simulate a theoretical model of a physical system and therefore replicate an exact solution to the system equations of motion there is no noise in the target output data. Therefore the weight decay in (2) is set to zero ($\alpha = 0$) in this example.

The wind turbine response history generated by the FEM model (4) is divided into a training and a test set as shown on Figure 3. Out of the 100 s response history the first 80 s is used

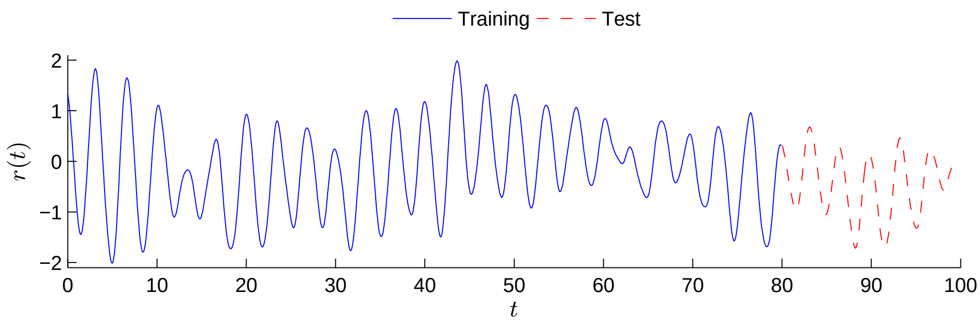

Figure 3. Training and test data.

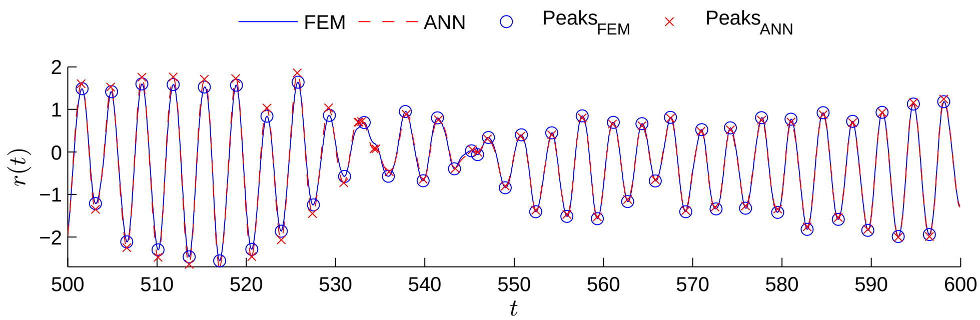

Figure 4. Exact FEM solution together with ANN simulation.

for training the ANN and the last 20 s is used for testing. With a time step size of 0.1 s and a dominating vibration frequency of 0.3 Hz, this gives a training set of 800 points covering about 24 vibration cycles. Parameter investigations not included in this paper show that an accurate, compact and robust ANN is obtained with 50 units in the hidden layer and 10 previous time steps. After training the ANN is able to predict the wind turbine response to a similar wind load history as the one used to generate the training data. The results of simulations using FEM model and ANN is shown in Figure 4. It is seen that the ANN captures the dynamics of the system very well and predicts the deflection quite accurately.

Concluding remarks

In the example presented in this paper the reduction in cpu time spend on a simulation of 10 min. wind is about a factor of 100 when using the ANN compared to the FEM - that is when the ANN is set up and trained. This factor is obviously dependent on the structure, the loading, model configuration etc. However, the examples indicates that the method holds a great potential. Furthermore, the reduction in calculation cost may be increased even further if it is possible to skip a few post possessing steps so that, instead of just predicting structural response, the ANN can be trained to predict material stresses or structural damage directly based on force input.

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

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