International Journal of Engineering & Technology, 7 (3.34) (2018) 163-167

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Analysis Methodology of Inelastic Constitutive Parameter Using State Space Method and Neural Network

[1]**JoonSeong Lee**

[1]*Department of Mechanical System Engineering, Kyonggi University, 154-42, Gwanggyosan-ro, Yeongtong-gu, Suwon, Gyeonggi-do, 16227, Korea*

[*]*Corresponding author E-mail: jslee1@kyonggi.ac.kr*

Abstract

Background/Objectives: In this paper, we present a method for describing a set of variables of an inelastic constitutive equation based on state space method (SSM) and neural network (NN). The advantage of this method is that it can identify the appropriate parameters.
Methods/Statistical analysis: Two NNs based on SSM are proposed. One outputs the ratio of inelastic strain for the internal parameters of the material, and the other is the following state of the inelastic strain ratio and material internal variable. Both NNs were trained and successfully collected using input and output data generated by Chaboche 's model.
Findings: As a result, previous NNs have demonstrated their validity as a powerful material model. However, the training data for the proposed NN can't be easily obtained from actual experimental data. Previous neural networks can reproduce the original stress-strain curves. The NNs also produced untrained curves to demonstrate interpolation capabilities. It was also found that the NNs can be estimated to be close to training data. The author defines the implicit constitutive model and proposes the implicit viscous constitutive model using NNs. In modeling, inelastic behavior is generalized in state space representation, and the state space form is constructed by NNs using an input-output data sets. The proposed model was first created from the pseudo-experimental data generated by one of the commonly used configuration models and has been found to be a good replacement for the model. The actual experimental data was then tested, and the proposed model showed the accuracy of its superiority over all existing specified models because the amount of model errors was negligible.
Improvements/Applications: The comparison between the NN constitutive laws with the Chaboche's model indicates that the NN constitutive law generated curves with less model errors than the experimental data, thereby indicating the superiority of the neural constitutive law to explicit constitutive laws as a material model.

Keywords: *Chaboche's Model, Inelastic Constitutive, Multilayer Neural Network, Ramberg Osgood Model, State Space Method*

1. Introduction

Problems using explicit constitutive equations are difficult to determine the inaccuracies and appropriate parameters of the model itself. The former problem overcomes the same parameter approach mentioned in the Refs.[1,2]. Also, we need to introduce a model that is replaced by a more complex explicit model or an implicit constitutive equation.

In the multilayer Ns, we proposed a material modeling by couples as described in the Ref. [3]. Yamamoto's model[4] is not as strong as the other two models. Both models describe the results of the neuro-based model created with the help of Ramberg-Osgood model. Nevertheless, it does not mean that other methods are best suited. One of the drawbacks of Ghaboussi's model [5] is that path dependence is achieved by taking only the past three points. Needless to say, even if the number increases, the size of the input space increases, the number of past points must be increased to account for the hysteresis characteristic of the data. Miyazaki's architecture [6] is rather a linear imitation of Ghaboussi's model. Therefore, to illustrate path dependency, the architecture used two components in duplicate internal variable and last three points. Another serious problem is that both common models use $\Delta\sigma$ and ΔY increments as inputs and are very sensitive to the experimental data. This can produce unstable NNs, especially if the measurement is not suitable for the error. A common NN configuration model is presented with a state space method that can describe all dynamic systems.

What is to be asked before the capability of the parameter identification technique to be discussed is required is "Is there sufficient experimental data to uniquely determine all parameters?" An experimental design suitable for parameter identification of the material model by Chaboche has been considered first and will be described in the next section. In addition to the results, we selected the actual experiment data needed to identify the parameters in the benchmark project and confirmed the parameters of the Chaboche model. The same parameter identification problem has been addressed by two other existing technologies for comparison. Accordingly, the author defines an implicit configuration model, unlike all existing configuration models, and then use the state space method (SSM) based NN to propose a new implicit point viscosity model.

2. State Space Method

The concept of a state space comes from state variable method that describes a differential equation. In this method, the dynamic system is described by a set of differential equations in the first

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

sequence in a variable called "state", where the solution can be visualized as a trajectory through space. This method is especially suitable for performing calculations with a computer.
While the use of state-space approaches is called the modern control theory [7-9], the use of transfer function-based methods such as source trajectories and frequency response has been referred to as classic control design. The benefits of state space design are particularly evident when an engineer designs a controller for a system with one or more control inputs or one or more sensed outputs.
For instance, newton's law of moving under force F in one-dimensional x for a single mass, M is

$$M\,\ddot{x} = F\,. \quad (1)$$

If one state variable is defined as$x = \dot{x}_1$and the other state variable is defined as$x_2 = \dot{x}_1$, this equation can be written as

$$\dot{x}_1 = x_2 \quad (2)$$

$$\dot{x}_2 = \frac{F}{M}\,. \quad (3)$$

Thislinear differential equationcan be simplified using matrix notation. It is possible to mark a state as a column vector**x**, a coefficient for the state equation as a square matrix **A**, and an input coefficientsfor column matrix **B**as a matrix:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}\frac{F}{M} \quad (4)$$

or

$$\dot{x} = \mathbf{A}'x + \mathbf{B}\,u \quad (5)$$

where **B** is the input matrix and **A** is the system matrix.

In conclusion, previous work by the dynamics community has demonstrated that multi-layer NNs can emulate a system where the structure is unknown, but the input-output data can be obtained. Two major NN structures have been intensively used [10,11], although much work has not been done so far.

Case I Output $\dot{x}^k$from x^k and u^k,

Case II Output x^{k+1} from x^k and u^k,

And they are illustrated in figure 1. Clearly Case II is more complex than Case I due to the additional integration terminology. Figure 2 and figure 3 show the training of the NN from the input and output data of the system.

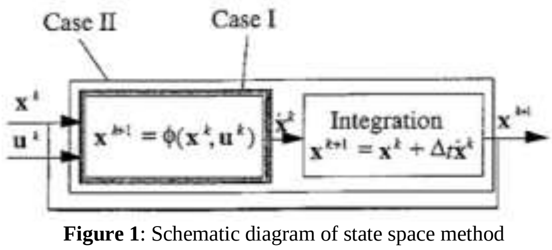
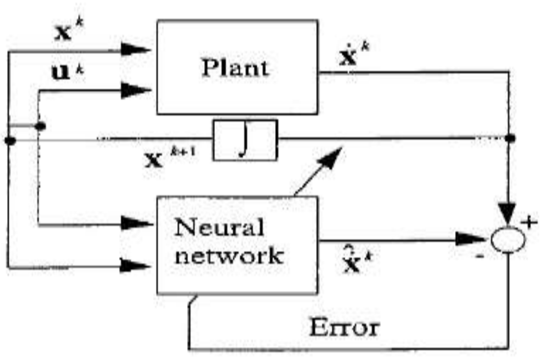
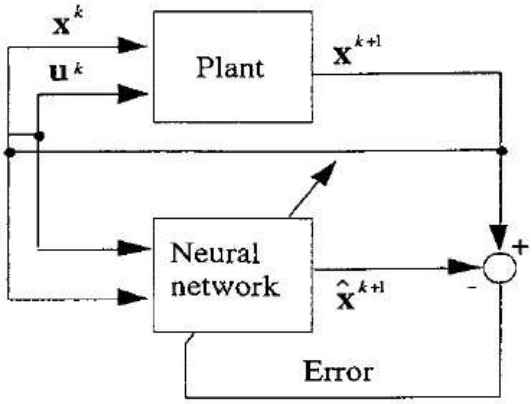


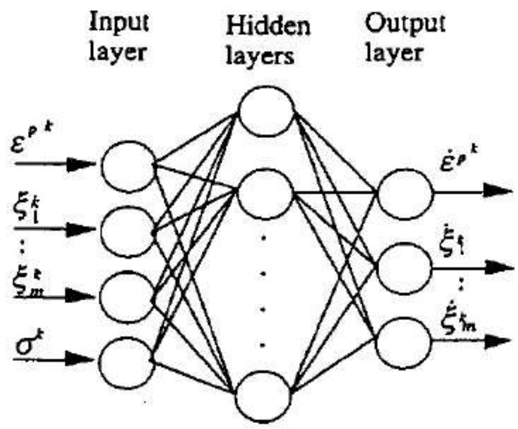
Figure 1: Schematic diagram of state space method



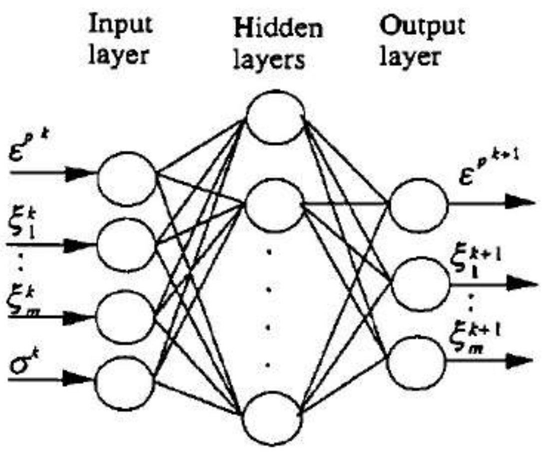
(a) Case I



(b) Case II
Figure 2: Block diagrams of training NN



(a) Case I



(b) Case II
Figure 3: Neural network architectures

If it can describe material behavior in the form of a state space, we can apply NNs to mimic behavior and use the network as a material model. Te next section will suggest a general state space formulation of material behavior.

3. State Space Representation of Material Behavior

In previous paper, we have seen some famous inelastic constitutive equations. When creating a general equation, the first feature we see is that all models consist of equations of plastic deformation and material internal variables. Then the equation is given most commonly with the ratio of inelastic strain and material internal variables. This makes it possible to construct a state space representation of inelastic constructive laws in a general way. The control inputs of known dynamic systems for all t are generally independent of the state variables.

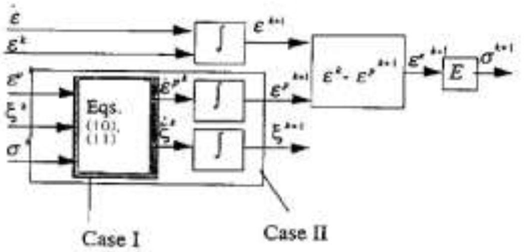
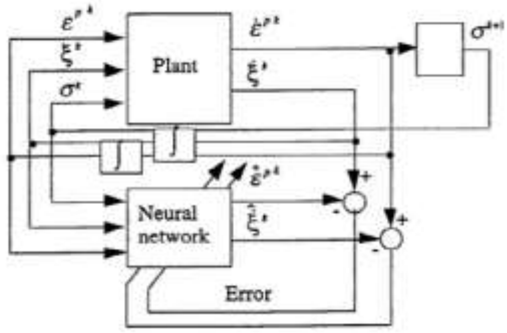


Figure 4: Block diagrams for simulation

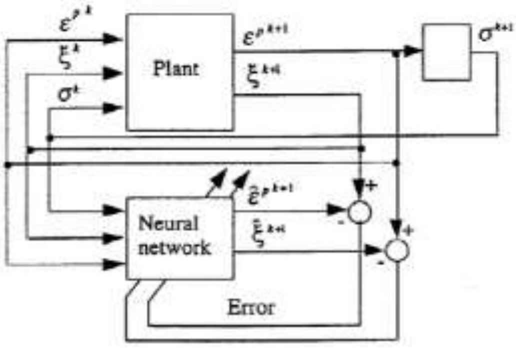
According to the previous section, we can offer two NNs of case I and case II. The proposed NN structure and the block diagram illustrating the induction of stress are shown in figure 3 and figure 4, respectively, and this figure is illustrated in Fig. 5 as a block diagram for network training. The advantage of NN architecture for other networks is obviously the following points.

(a) Simplicity: Compared with Miyazaki's model [6], which uses two NNs independently, only one NN is used. The input layer consists of the internal variables, the strain and stress of the newest information.

(b) Generality: Depending on the material selected, all kinds of internal variables can be used if the material have a state space representation.



(a) Case I



(b) Case II

Figure 5: Block diagrams of training NNs

4. Numerical Examples

This section uses computer-generated pseudo-experimental data to investigate the performance of the proposed NNs. For example, the NN is determined to use two internal variables, namely back stress Y, and the isotropic hardening parameter R (called yield stress, which begins to increase at the yield surface), to be used in Chaboche models. Thus, the network consists of four inputs and three outputs.

The model used to generate the virtual experiment data is also the model of Chaboche. Because the same internal variables are used, you can check your network performance directly to find out about each model equation. Table 1 lists the material parameters used to generate training data was 500, and Table 2 shows that the test was taken periodically in the first five cycles of the inverse loading test at a constant strain rate. Each verification data was represented by two training data.

Table 1: Material parameters for generating training and validation data

H	n	K	d	h	R^0	D
6,000	4	60	0.5	400	60	200

Table 2: Reverse cycle load test parameter

ε_{max}(%)	$\|\varepsilon\|^{s-1}$	# of validation sets	# of training sets
0.03	8.0 x10^{-3}	490	500

4.1. Case I

The errors in the training and validation sets are shown in figure 6. Obviously, the error is approaching to zero. This means that the NN is learning the law of material.

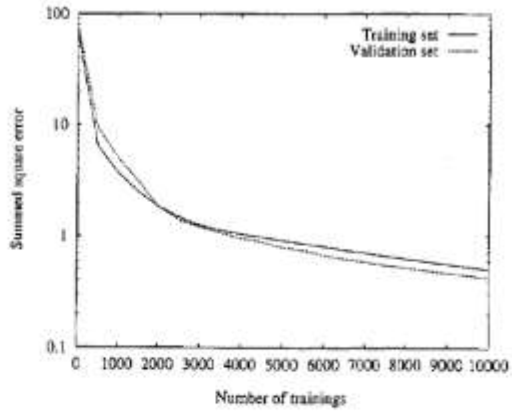


Figure 6: Error of training set and validation set (Case I)

As shown in figure 7 and figure 8, the NN curves with ±0.072% were far from the correct curve. It can be clearly seen that the NN curve in the first cycle deviates from the Chaboche curve after a 0.036% strain. NN training was not experienced. There are significant errors in the stress and strain curves.

The ability of the NN was also investigated at different strain rates of 0.8%, and figures 9 and 10 show the results. In figures 9 and 10, we can easily see that the NN curve exhibits very inelastic behavior. NNs no longer represent material behavior.

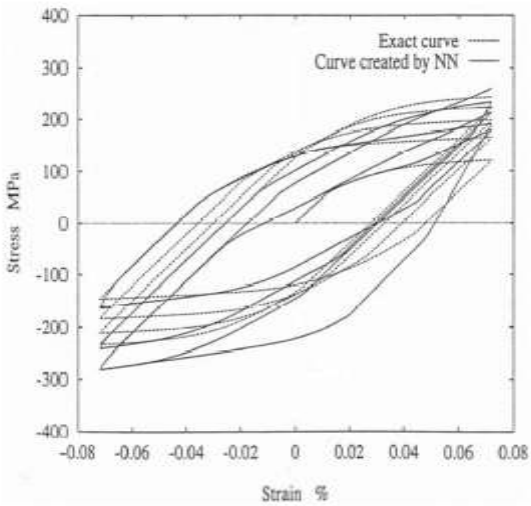


Figure 7: Exact stress-strain curve generated by NN (Max. strain range: 0.072%)

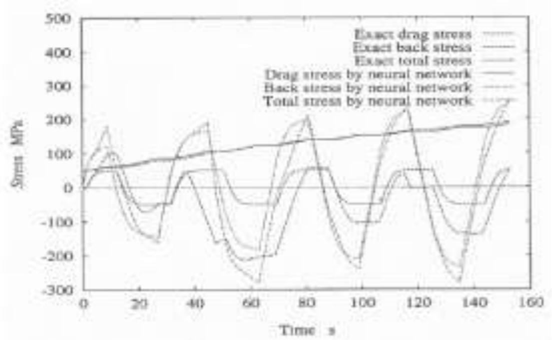


Figure 8: Exact stress curve generated by NN (Max. strain range: 0.072%)

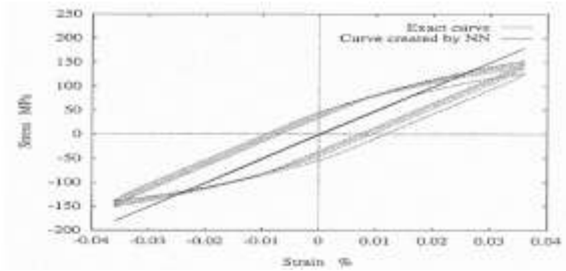


Figure 9: Exact stress-strain curve generated by NN (Max. strain range: 0.036%)

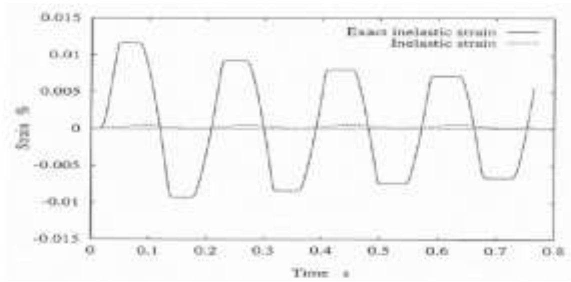


Figure 10: Exact stress curve generated by NN (Max. strain range: 0.036%)

4.2. Case II

The validation and training data depicted in figure 11 was used for training, but the results of the trained network are not very good. Therefore, the network was trained with the training data obtained from all computer simulations during the material tensile behavior. Error propagation of training and verification is set up until 60,000 training sessions shown in figure 12. You can see that the NN can learn the curve. However, figure 12 shows that the NN curve deviates significantly from the correct curve. The reason for the deviation can be easily explained in the curve of back stress and the inelastic strain of figure 13, where the deviation from the correct curve increases over time.

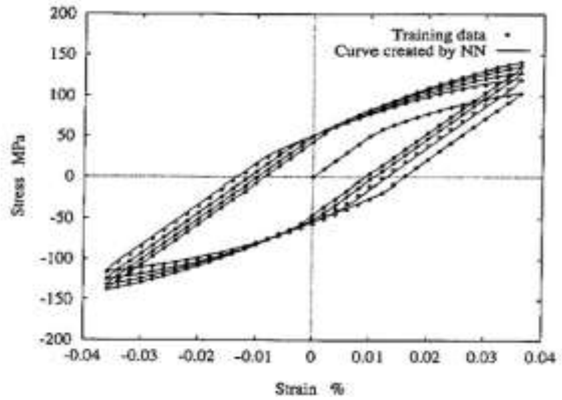


Figure 11: Stress-stain plots produced by training data and the curves produced by NN

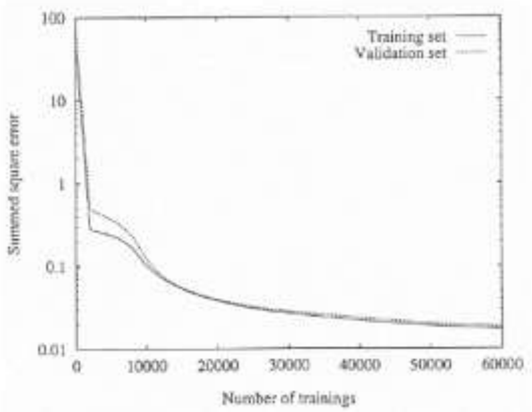


Figure 12: Errors of training and validation data (Case II, Max. strain range: 0.036%)

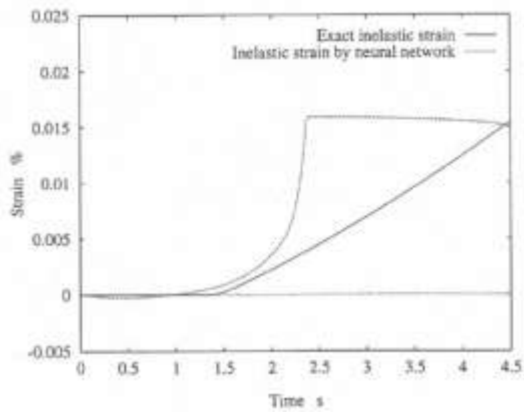


Figure 13: Exact inelastic strain curve generated by NN (Case II, Max. strain range: 0.036%)

5. Conclusion

Two neural networks are proposed, a material model based on state space methods. One outputs the inelastic strain rate with respect to the internal parameters of the material, and the other shows the next state of the inelastic strain and the internal variable of the materials. Both neural networks were trained and successfully converged using input and output data generated by Chaboche's model. Previous neural networks can reproduce the original stress-strain curves.

Techniques have been developed to decompose experimental data into learning data on the constitutive laws of neural networks. The neural network construction law is composed of the same experimental data as that used for the most suitable Chaboche curve. Comparing the Chaboche’s model with the NN constitutive law, the neural net constitutive law shows that the neural constitutive law expresses superiority to the explicit constitutive law as a material model by creating curves with less model errors than experimental data. Nevertheless, the training data for the proposed neural network can’t be easily obtained from actual experimental data. The next version refers to a strategy for extracting training data from an experiment.

Acknowledgment

This work was supported by the GRRC program of Gyeonggi province.[GRRC 2017-B02, Research on Innovative Intelligent Manufacturing System]

References

[1] J.S. Lee and B.G. Bae (2011). Evolutionary Analysis for

Continuous Search Space. Journal of Fuzzy Logic and Intelligent Systems, 21, 206-211.

[2] T. Hurukawa, J.S. Lee and E.C. Lee (2009). Constitutive Parameter Identification of Inelastic Equations Using Evolutionary Algorithm. Journal of Fuzzy Logic and Intelligent Systems, 19, 96-101.

[3] Furukawa, T., Okuda, H. and Yagawa, G. (1995). On General Formulation of Neural Networks as a Material Model. Proceedings of the AnnualMeeting of JSME/MMD, Morioka, A, 417-418.

[4] Yamamoto, K. (2002). Modeling of Hysteretic Behavior with Neural Network and its Application to Non-linear Dynamic Response Analysis.Applications of Artificial Intelligence in Engineering, Computational Mechanics Publications and Elsevier Applied Science, 475-486.

[5] Ghaboussi, J., Garrett Jr., J.H. and Wu, X. (1991). Knowledge-Based Modeling of Material Behavior with Neural Networks. Journal of Engineering Mechanics, 117(1), 132-153.

[6] Miyazaki, H. (1995). *Inelastic Analysis Using a Neural Network Constitutive Law*, The University of Tokyo, Japan.

[7] Franklin, G.F., Powell, J.D. and Emami-Naeini (2008). *Feedback Control of Dynamic Systems*, Addison Wesley.

[8] S.F. Su, Y.C. Hsueh, C.P. Tseng, S.S Chen and Y.S. Lin. (2015). Direct Adaptive Fuzzy Sliding Mode Control for Under-actuated Uncertain Systems. International Journal of Fuzzy Logic and Intelligent Systems, 15(4), 240-250.

[9] J.Y. Park and J.H. Kim. (2004). Controller Design for Fuzzy Systems via Piecewise Quadratic Value Functions. International Journal of Fuzzy Logic and Intelligent Systems, 4(3), 300-305.

[10] M. Khayet (2011). Artificial neural network modeling and response surface methodology of desalination by reverse osmosis. Journal of Membrane Science, 368 (1), 202-214.

[11] H.W. Lee, N.R. Kim and J.H. Lee (2017). Deep Neural Network Self-training Based on Unsupervised Learning and Dropout. International Journal of Fuzzy Logic and Intelligent Systems, 17(1), 1-9.