Deep Neural Networks for Maximum Stress Prediction in Piping Design

Sang-jin Oh¹, Chae-og Lim¹, Byeong-choel Park¹, Jae-chul Lee², and Sung-chul Shin¹

¹Department of Naval Architecture and Ocean Engineering, Pusan National University, Busan, Korea

²Department of Naval Architecture and Ocean Engineering, Gyeongsang National University, Tongyeong, Korea

ljfis

Abstract

Piping design mainly consists of design, modeling, and analysis steps. Once all processes of the design and modeling steps are completed, the maximum stress values obtained in the analysis step are compared with those prescribed by the regulations to complete the piping design. If these values do not satisfy those provided by the regulations, the entire design must be modified. In the analysis step, bottlenecks occur because both design and modeling must be re-performed. This requires considerable time and effort from the designer, and it is a major factor lowering designer productivity. To achieve efficiency, the required maximum stress value should be considered in the initial step itself. In this study, a deep neural network was used to predict the maximum stress. Based on the accuracy of the predicted analysis results, it was possible to shorten the design time while improving the piping design.

Keywords: Neural network, Deep learning, Maximum stress, Piping design

1. Introduction

Piping design mainly consists of design, modeling, and analysis steps. Generally, the piping and instrumentation diagram and material properties are determined in the design step, and the detailed pipe arrangement is determined in the modeling step. Once all processes of the design and modeling steps are completed, the values obtained in the analysis step are compared with those prescribed by the regulations, thus leading to the completion of piping design. However, many modifications are required in the analysis step before the piping design can be finalized. For the design and modeling steps, these modifications are made through mutual complementation, and the analysis step can be initiated only after completing the previous steps. Therefore, the analysis step requires considerable time and loss compared to the other steps. Piping design is classified as levels A, B, and C, as determined by the business owners; the time required to complete the piping design inevitably increases from levels A to C. In particular, since high-temperature piping has a level C classification and requires in-depth examination using a dedicated program even in the analysis step, a bottleneck occurs due to requirement for repeated modifications and long time.

Piping design should be performed by considering the maximum stress in the initial design step to reduce the number of iterations and the correction time required in the analysis step and to optimize the design. The maximum stress prediction can also consider the regulations

Received: Mar. 12, 2019

Revised : Aug. 17, 2019

Accepted: Sep. 18, 2019

Correspondence to: Sung-chul Shin

(scshin@pusan.ac.kr)

©The Korean Institute of Intelligent Systems

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

in the initial design by early comparison of the values, which would otherwise be compared only in the analysis step. Further, the maximum stress prediction can help with material purchase, as it can identify the initial quantity of material required by estimating the type and number of piping supports. With the recent advent of the fourth industrial revolution, active research on deep learning based on big data has been conducted; in addition, big data and artificial intelligence have been attracting attention in the offshore plant area. However, the use of artificial intelligence and big data for piping design is still very rare. At present, studies to shorten the time spent on piping design by shortening the time required for computer aided design modeling are ongoing. Lee et al. [1] proposed a method to shorten the design time for piping systems by reducing unnecessary corrections through collaborations between designers using rapid modeling technology and basic structural analysis. Deep learning is used in various fields such as speech recognition and image recognition systems. In particular, it is widely used in the field of prediction. Among the various deep learning methods, a deep neural network (DNN) structure has been effectively applied to regression analysis when general analysis procedures cannot be applied due to the complexity of data. The DNN structure can predict nonlinear regression more than linear functions. Also, neural networks can process large amounts of data, since they run in parallel; further, they are highly error-tolerant, since they employ sophisticated statistical systems. Research related to prediction using the DNN structure is ongoing in various areas, such as the prediction of winter power demand, real estate, and runway visibility distance. In this study, we determined the features necessary for pipe stress analysis and generated data by combining different features. We performed structural analysis on the generated data using ANSYS after 3D modeling. We then formulated a neural network model to predict the maximum stress of the pipe based on the values of the generated data and those obtained by structural analysis. A neural network model is an input/output model in which features are entered as input variables of the neural network and the maximum stress on the piping system is obtained as the output variable. The neural network used in this study was built based on a DNN model obtained using Google's TensorFlow. Finally, we obtained the error rates and confirmed the accuracy of the model by comparing the maximum stress estimated through the DNN model and those measured by actual analysis. Section 2 describes the maximum stress prediction features and data generation with respect to related research. Section 3 proposes a deep learning-based maximum stress prediction model, and

Section 4 presents the experimental prediction results. Lastly, Section 5 provides the conclusions of this study.

2. Prediction Features and Data Generation

2.1 Feature Selection

There are various pipelines, such as steam lines, feed preheater lines, and pump lines, in a plant. In this study, the steam line was considered for the experiments. First, basic shape information is required for piping analysis. Shape information includes the spatial coordinates, length, thickness, and diameter of the pipes. Next, property information and operating conditions are required to determine the characteristics of the pipes. Property information of the pipe includes the properties of the pipe material, such as elastic modulus, Poisson's ratio, density, thermal conductivity, and friction coefficient. Operating conditions include the operating environments of actual plants and consider factors such as internal/external pressures and gravity of pipes as well as the temperature, density, and height of the free surface of fluids inside the pipe. Among these, we selected features based on the properties prescribed by the regulations. The thickness and diameter of the pipes were selected as features for shape information. Overall, there were four features: the thicknesses and diameters of the main and branch pipes. Three features were selected for the operating conditions: temperature, pressure, and wind. The seven features mentioned above were finally used as input variables. For the diameter and thickness of all pipes, the number of variables was set according to the JIS regulations. The diameters were 20", 18", and 16", and the data were generated by changing the thickness with respect to schedule number STD, 40. For temperature, four values, 90°, 120°, 150°, and 180°C, were selected. For pressure, two values, 500 and 700 Pa, were selected. For wind conditions, three values, 900, 1150, and 1550 N/m², were selected. A total of 480 cases could be generated for the training data by combining each variable. The material properties of the pipes were determined as A106B, the material of the pipe, using property information based on ASME codes B31.1 and B31.1.

2.2 Data Generation

Modeling and analysis of the pipes were performed using ANSYS software to generate data. The pipe considered was a steam line consisting of one main pipe and two branch pipes. Figure 1 shows a model of the steam line used in the analysis. The bending point of the pipe was 1.5 times its diameter, and the

Figure 1. Typical steam line model used in ships and offshore plants.

Figure 2. Upper analysis results of a typical steam line model. Analysis results for main pipe and two branch pipe connections are shown.

Figure 3. Results of sub-analysis of a typical steam line model. Analysis results for main pipe and two branch pipe connections are shown.

start and end points of all the pipes were connected to devices. Stress distribution was high in the region where the analysis results were fixed at the top position (Figure 2). In Figure 3, the analysis results were fixed at the bottom position, and the stress distribution in this region was high. To learn the neural network, the overall data was divided into training data, validation data,

Figure 4. DNN model consisting of a fully connected layer.

and test data; as proposed by Beale et al [2], 70% of the total data was used for training, 15% was used for validation while learning, and the remaining 15% was used for testing.

3. Deep Neural Network Model

We proposed a DNN structure (Figure 4), which is the most suitable for regression analysis, to predict the maximum stress. The DNN structure consisted of an input layer, hidden layer, and output layer. The layer depth increased as the number of hidden layers increased. All neurons of the hidden layers together formed a fully connected DNN. We confirmed the results by comparing errors of the test sets while changing the number of hidden layers and neurons. Here, case generation was empirically chosen. Backpropagation was achieved by generalizing the Widrow–Hoff learning rule [3] to the multi-layer perceptrons and the non-linear differential transfer functions. Features and the corresponding target were used to approximate the functions, to associate the features with the target, and to train the neural network until the features were appropriately classified. During the learning, weight and bias in the neural network were iteratively adjusted to minimize the mean square error. Networks with bias and hidden/output layers could approximate all functions with finite numbers of discontinuities. In this study, we used Adam Optimizer as the backpropagation optimization algorithm, and the weight in the neural network was moved, so that the mean square error could be minimized. The properly trained backpropagation neural network provided a more reasonable answer than the untrained data. Adam Optimizer is a training algorithm provided by TensorFlow. It calculates the error function from the given training dataset and corrects parameters in the opposite direction of the slope vector. In addition, its structure automatically adjusts according to the learning rate parameters. Therefore, it is one of the most frequently used algorithms in deep learning, as it performs relatively well and

Figure 5. DNN model composed of input layer, hidden layer, and output layer.

does not require manual adjustment of the learning rate. To perform the learning process more efficiently, the data were normalized to values between 0 and 1, where an activation function occurred. Eq. (1) represents the normalization method.

$$\bar{d} = (d - \min) / (\max - \min), \quad (1)$$

where $\bar{d}$ is the data value after normalization and d is the data before normalization and $\max$ and $\min$ represent the maximum and minimum values of the corresponding features, respectively. The activation function is a function that receives an input signal and outputs it after proper processing. An activation function was assigned to each layer to provide appropriate values to neurons of the next layer after receiving input signals from the neurons of each layer. Among the activation functions, as shown in Figure 5, rectified linear unit (ReLU) was used as a function to replace the sigmoid function with values ranging from 0 to 1 [4]. ReLU is a linear function that assumes 0 for values below 0 and x for values above 0 [5]. It solved the gradient vanishing problem that converged to 0 with increasing layer depth for values ranging from 0 and 1. The initialization by He et al. [2], which is well suited for ReLU, was used to initialize weight.

4. Prediction Results

This study used Google's TensorFlow 1.7.0. The algorithm used included 7 features, and it was configured to generate the maximum stress as a single output. We confirmed the errors of the test set while changing the number of hidden layers and neurons in the proposed model to compare the results. Also, we compared our results with those of commonly used linear regression functions and machine learning. When all cases were learned, we confirmed that the difference in loss was less than 0.0001 and identified overfitting up to that time by substituting

Table 1. Combination of layers and neurons

Case	Layer	Neuron
1	3	8
2	4	16
3	5	32
4	6	64
5	7	128

the validation set. In Figure 5, the number of layers were increased from 3 to 7, and the number of neurons increased by the square of 2 (from 8 to 128) whenever the number of layers increased. Table 1 shows the combination of layers and neurons for each case. When the number of layers was decreased, the number of neurons reduced by the square of 2 and the output layer was configured to converge to 1. We used a multiple linear regression (MLR) model and a support vector machine (SVM) for machine learning.

While conducting the learning on TensorFlow, we ensured that overfitting or underfitting did not occur through the validation of the loss. For all five cases, the results were confirmed in terms of loss-and-error graphs (Figure 6), which were classified by the training and validation sets, to determine whether the learning was completed. Case 1 is a network composed of three layers. At a generation value of 1,700, the loss value and the error of the predicted value converged. In Case 2, consisting of four layers, the loss and error values converged at a generation value of 2,500. Case 3 shows that convergence was constant at a generation value of 2,500, with the loss value converging to a lower value after one more step at a generation value of 9,000. The error value also showed the same tendency. Case 4 shows that the loss and error values converged at a generation value of greater than 100,000, unlike the tendency observed for the cases with 3–5 layers. In Case 5, the learning was completed because the loss and error values converged at a generation value of 2,000. Finally, Table 2 shows the average error values of the training, validation, and test sets for each case along with the average error values of the training and test sets for the MLR model and SVM. Figure 7 shows the prediction results for the test data.

As shown in Table 2, the error values decreased as the number of layers increased and the number of nodes increased. In Cases 1–3, the test set error values decreased greatly from 20.53% to 2.77%, but in Cases 3–5, the test set error value decrease was smaller (from 2.77% to 1.12%). On examining

Figure 6. Loss-and-error graph for (a) Case 1, (b) Case 2, (c) Case 3, (d) Case 4 and (e) Case 5. By setting the batch size to 100, the training (black line) becomes thicker.

Table 2. Average error values of training, validation, and test sets (unit: %)

Set	MLR	SVM	Case1	Case2	Case3	Case4	Case5
Training	58.51	13.43	27.26	16.46	1.14	0.77	0.67
Validation			23.75	20.35	2.35	1.08	1.29
Test	68.81	16.08	20.53	18.94	2.77	1.32	1.12

the difference between the regression function and the neural network, we confirmed that the test set error value for the MLR model was 68.81%, which was much higher than that of the neural network-based models. SVM, a major machine learning

technique, showed better performance in terms of error value (16.08%) than the MLR model, but its performance was poorer than that of the neural network-based models with 5 or more layers (Cases 3–5).

Figure 7. Prediction results for test data. In case of neural networks, the larger the number of layers, the smaller the error from the target.

5. Conclusions

This study defined the features affecting maximum stress, built data for learning a neural network, and proposed a neural network model for predicting the maximum stress of a pipe by neural network learning using TensorFlow. When we compared the results from the stress analysis program with the predicted values from the neural network model, we could confirm the smallest average error rate to be 1.12%, and that the neural network models exhibited better performance than the MLR model and SVM. We varied the number of layers and neurons in the neural network model and found that the best prediction rate, i.e., the smallest error value, was achieved when there were 7 hidden layers. In this study, we defined seven features for measuring the stress and derived the results for the steam line. The proposed neural network model considered the maximum stress in the initial design step, thereby reducing the number of iterations and hence the time required for correction and reducing the bottleneck during the design process. The proposed model has the potential to be applied practically if environmental features affecting the maximum stress value, such as the location of valves, nozzles, and pipes, are considered and more data are built in the future.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgements

This work is partly supported by Korea Agency for Infrastructure Technology Advancement grant funded by Ministry of Land, Infrastructure and Transport (18IFIP-B133628-02, Development of on-shore mud mix and treatment system) and National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) through GCRC-SOP (No. 2011-0030013).

References

- [1] G. B. Lee, T. S. Kim, S. Kim, Y. Choi, and S. W. Cho, "Conceptual design and preliminary structural analysis of a traditional plant piping system," in *Proceedings of the the Korean Society of Mechanical Engineers Autumn Conference*, Changwon, Korea, 2012, pp. 2039-2043.
- [2] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Neural Network Toolbox User's Guide*. Natick, MA: The MathWorks Inc., 1992.
- [3] B. Widrow and M. E. Hoff, "Adaptive switching circuits," Stanford Electronics Labs, Stanford University, Stanford, CA, *Technical Report No. 1553-1*, 1960.
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097-1105, 2012.

- [5] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, Haifa, Israel, 2010, pp. 807-814.
- [6] K. He, X. Zhang, R. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on ImageNet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, 2015, pp. 1026-1034. <http://doi.org/10.1109/ICCV.2015.123>

Sang-jin Oh received his B.S. and M.S. degrees in Naval Architecture and Ocean Engineering from Pusan National University, Korea in 2017 and 2019, respectively. His present interests include object detection and image classification using deep learning.

E-mail: osj5588@pusan.ac.kr

Chae-og Lim received his M.S. degrees in Naval Architecture and Ocean Engineering from Pusan National University, Korea in 2016. Currently, he is a Ph.D. candidate of Naval Architecture and Ocean Engineering from Pusan National University, Korea. His present interests include HILS, control system design and artificial intelligence.

E-mail: orc@pusan.ac.kr

Byeong-choel Park received his M.S. degrees in Naval Architecture and Ocean Engineering from Pusan National University, Korea in 2015. Currently, he is a Ph.D. candidate of Naval Architecture and Ocean Engineering from Pusan National University, Korea. His present interests include risk/reliability analysis and fire explosion analysis.

E-mail: bcpark@pusan.ac.kr

Jae-chul Lee received his M.S. and Ph.D. degree in Naval Architecture and Ocean Engineering from Pusan National University. He is currently assistant professor at the Naval Architecture and Ocean Engineering from Gyeongsang National University, Tongyeong, Korea. His current research interest includes artificial intelligence and production.

E-mail: j.c.lee@gnu.ac.kr

Sung-chul Shin received his B.S., M.S. and Ph.D. degree in Naval Architecture and Ocean Engineering from Pusan National University. He is currently a professor at the Naval Architecture and Ocean Engineering from Pusan National University, Busan, Korea. His current research interest includes artificial intelligence and deep learning.

E-mail: scshin@pusan.ac.kr