IN THE CHALLENGE OF EARTHQUAKE RESPONSE PREDICTION BY NEURAL NETWORK APPROACH

EN EL DESAFÍO DE LA PREDICCIÓN DE LA RESPUESTA A TERREMOTOS MEDIANTE EL ENFOQUE DE REDES NEURONALES

Carlos Zavala[1,2] , **Miguel Diaz[1,2]** , **Claudia Honma[2]**

[1] *Facultad de Ingeniería Civil, Universidad Nacional de Ingeniería, Lima, Perú*

[2] *Japan-Peru Center for Earthquake Engineering Research and Disaster Mitigation, Lima, Peru*

ABSTRACT

Since the decade of the 1990 neural networks algorithms have been used for compute approximate solutions for different problems in engineering.
In the building behavior against loads is important to know its response. The behavior during earthquakes and the estimation of the response is quite difficult to compute due to the nonlinearity of geometry and material properties. Neural networks approach is a powerful tool for computing the response of structures with an appropriate learning process from big data of structural components. Even if some material parameters are unknown, learning on a neural network will be possible and will provide an estimation using collect information from experience and learning.
To make a learning process in this paper, we present a simple algorithm of back propagation implemented in python programming language where the output shows the decrease of the error and how the response start to learn from the beginning until the end of the process. The results show good agreement between the learning data set and predicted response after the neural network learning.

Keywords: confined masonry, neural networks, building response, dynamic earthquake response.

RESUMEN

Desde la década de 1990, los algoritmos de redes neuronales se han utilizado para calcular soluciones aproximadas para diferentes problemas de ingeniería.

En el comportamiento de la edificación frente a las cargas es importante conocer su respuesta. El comportamiento durante los terremotos y la estimación de la respuesta es bastante difícil de calcular debido a la no linealidad de la geometría y las propiedades del material. El enfoque de redes neuronales es una herramienta poderosa para calcular la respuesta de estructuras con un proceso de aprendizaje apropiado a partir de grandes datos de componentes estructurales. Incluso si se desconocen algunos parámetros materiales, el aprendizaje en una red neuronal será posible y proporcionará una estimación mediante la recopilación de información de la experiencia y el aprendizaje.

Para realizar un proceso de aprendizaje en este artículo, presentamos un algoritmo simple de propagación hacia atrás implementado en el lenguaje de programación python donde la salida muestra la disminución del error y cómo la respuesta comienza a aprender desde el principio hasta el final del proceso. Los resultados muestran una buena concordancia entre el conjunto de datos de aprendizaje y la respuesta prevista después del aprendizaje de la red neuronal.

Palabras clave: mampostería confinada, redes neuronales, respuesta de edificios, respuesta dinámica a terremotos.

1. INTRODUCTION

Neural networks started to be implemented in 1957, when Cornell Aeronautical Laboratory in Buffalo New York introduce the project PARA, "The perceptron – A Perceiving and Recognizing Automaton" [1], with the purpose was the designing, fabricating, and evaluating an electronic brain model, the photoperceptron.

This PARA pilot model will be capable of learning response to ordinary visual patterns or forms. The architecture of PARA provides the starting point of the artificial neural network generalize by [2], where the procedure for learning the correct sets of connection for arbitrary networks was for first time presented to the science community, presenting the back propagation method for learning from the current error of the network´s output. Here the rule called generalized delta rule, is a simple scheme for

---

* Corresponding author:
czavala@uni.edu.pe

implementing a gradient descent method for finding weights that minimized the sum squared error of the system´s performance. The big contribution of this method is the procedure called error propagation, whereby the gradient can be determined by individual units of the network based only on locally available information. Since local minimum can be stacked during the process, several variations of the method appear to include a momentum or noise in each process unit or select an appropriated function for improve the minimization process of the paper. In this report we will use the back propagation method for the estimation of cyclic response of a masonry wall.

2. THE NEURAL NETWORK MUTI LAYER BACK PROPAGATION ERROR LEARNING PROCESS

Neural network is a joint of processing units which spreads an input into the net to produce an output. These outputs have an error comparing with some targets or learning parameters. These errors must be minimized through the modification of the network

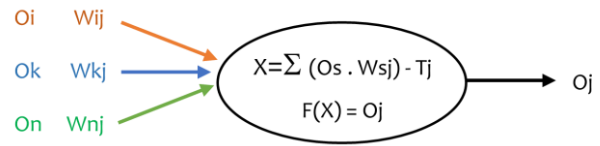components (inputs, outputs, threshold (Tj), weights (Wij).

Fig. 1. Processing unit [4]

Between processing units, strength connections exist which are called weights. In the neural network multi-layer back propagation error learning process, the task is to find a set of weights to produce from the input the minimum error [4]. The evaluation of the partial derivatives of the whole system is possible by the back ward propagation of the error, spreading from the upper layer to the lower one, which are used in the computation of the new weights for the next input, in this way the network components learn from the training data generating a learning process.
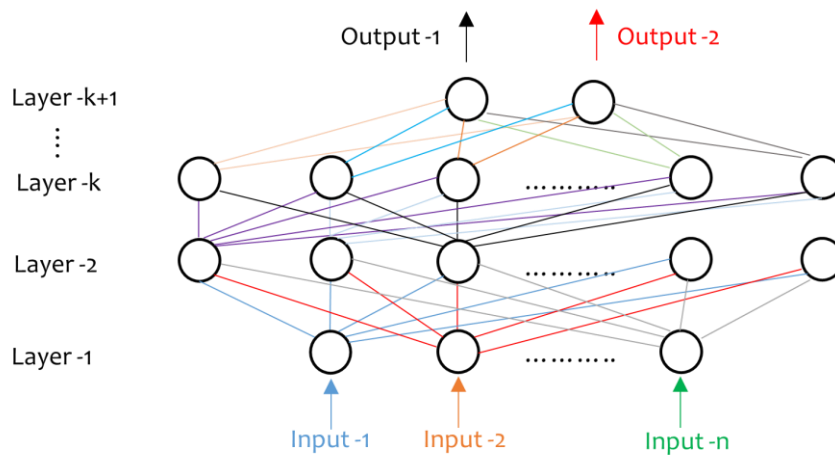

Fig. 2. Multilayer neural network [4]

3. ABOUT CYCLIC RESPONSE OF MASONRY WALLS

One of the first applications on cyclic response of structures was presented on [4] where a neural network model reproduces the cyclic response of an H shape beam, as shown in Figure (3a), which very good agreement in the comparison with a classic multi-spring model of nonlinear behavior of structural beam. Therefore, the application of the neural network model depends on the selection of the inputs as instant load, gradient, incremental step, and other parameters that can be used as input for the neural network approach. In this paper we will use the results of the cyclic test on masonry wall [6] as learning sets for the neural network presents on item 4 (Fig.3b).

Fig. 3a. Cyclic test [4] early test used for neural network simulation hybrid response.

Fig. 3b. Cyclic test [6] used for learning process for neural network simulation response.

As is known 82% of the housing in Lima city are built with masonry and the behavior of housing of this material has been studied for decades [5]. The uncertainty of the quality control during construction, the strength of the masonry and amount of reinforcement produce unknown parameters that can be manage but the non-linearity of a neural network.

The common dynamic model for computing the earthquake response of a building is usually springs masses model with damping factor, which are reduce to one degree of freedom model that reproduce the equivalent force compute from the step-by-step numerical integration model, considering an initial skeleton curve which is updated with the step-by-step response. In a neural network process a learning scheme using the above presented multilayer back propagation error, using a displacement as an input will produce an output with the value of the force. In the following section we present how the network was implemented and the learning process.

A neural network is built as object program considering the definition of each function that we need for the process:

    4.1 Initial of the network: initialize units, weights, and variables.
    4.2 Evaluate Sigmoid functions on the unit.
    4.3 Feedforward the inputs into the network.
    4.4 Compute the error against the Learning output, displaying the media of error of the sets.
    4.5 Backward the error into the network.
    4.6 Modify weights on the networks and units.
    4.7 Training process (repeat sampling times)

In our neural network example, we are considering 16000 sampling learning presentations to get and

4. BUILT A NEURAL NETWORK WITH PYTHON

An easy-to-use programing structural programming language PYTHON, became a source of many intellectual multipurpose applications, from gaming, computing, artificial intelligence, matrix management, numeral integration, sensor management, sensor measurement and IT. Of course, neural networks are included in the generation of algorithms and libraries manage by python. At first, we need to consider the use of three very important libraries for generate the application: Pandas library to manage the data input and out data, NumPy library, that's permit a feedforward and back forward definition, and Matplotlib library, that produce graphic data of the process and evolution of the output.

In this application we consider two inputs (displacement X and gradient Z) to produce one out (response force for the current displacement Y). Here, the output sets are compared with the learning sets Y_{LEARN} [6] to evaluate the error $(Y_{LEARN} - Y)/Y_{LEARN}$ of each presentation of data set. A very important task is the normalization of the involve process of the input variables and out variables, so prior the use of the network algorithm and the spreading of data into, we need to avoid the saturation of the Sigmoid function in the processing unit, presented in item 2.

$$F(X) = \frac{1}{1+e^{-X}} \qquad (1)$$

So, we found the absolute maximum / minimum to normalize each of the inputs and output variables to avoid the saturation of the Sigmoid function, therefore:

$$Xnorm = \frac{X}{aXmax} \qquad (2)$$

$$Ynorm = \frac{Y}{aYmax} \qquad (3)$$

$$Znorm = \frac{Z}{aZmax} \qquad (4)$$

acceptable response of our data and reduce it neglected level.

5. NEURAL NETWORK RESPONSE PREDICTION

As linearization is used in the common skeleton curves on wall cyclic loading, such first crack appearance, initial yielding of reinforce, maximum demand and ultimate demands, and the bias value (initial value) it means at least 5 pairs of data for the prediction of the capacity curve, in this network we consider 20 sets of data, to produce the nonlinearity of the capacity curve from a wall test. Figure (5) presents the initial output where currents weights start as a set random number. It is possible to see that the output force (orange) is very far from the learning data (blue).
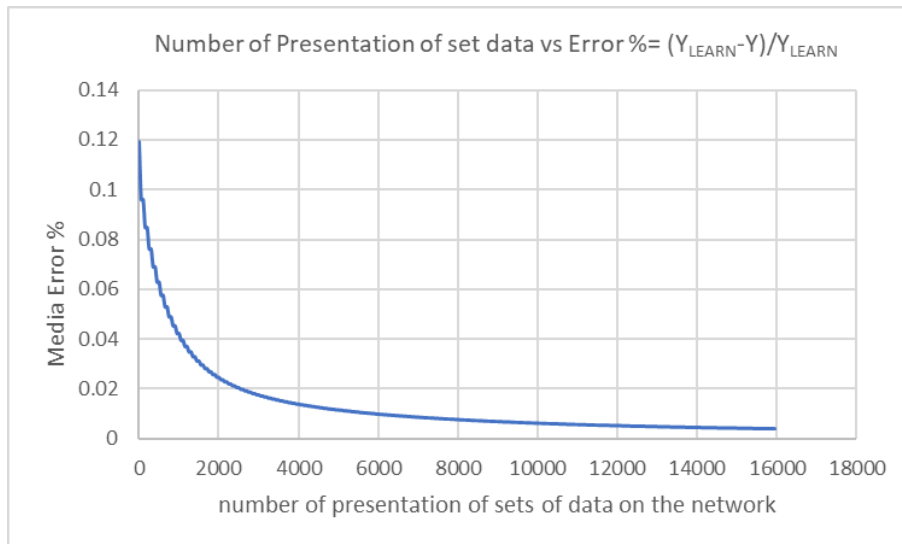
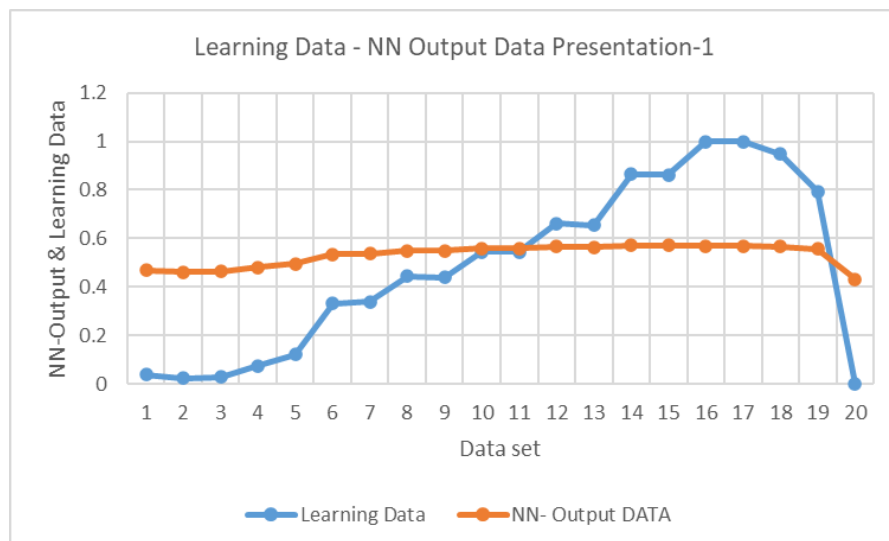Fig. 4. Media error of presenting sets evolution on the example neural network

Fig. 5. Presentation of NN output sets against Learning Data at initial presentation

Using the backpropagation error learning process, the units and weights are change every swapping of the sampling data, decreasing the error to neglected values as was presented in Figure (4), learning as child can learn after repeat and repeat a visual pattern to know that is a cat or dog, or repeat the name of his father or mother, in such way after the units are charge with the backward error and forward values, generating an artificial nonlinearity from the learning data. After 16000 presentations of forward and backward the sampling sets, a good agreement is produced as we can see in Fig. (6).

Representation of the normalize data as displacement force capacity curves are presented in Fig. (7) for the initial stage of the learning process and final learning at Fig. (8) with good agreement between the learning data and the output data from the artificial neural network using the backpropagation learning procedure.
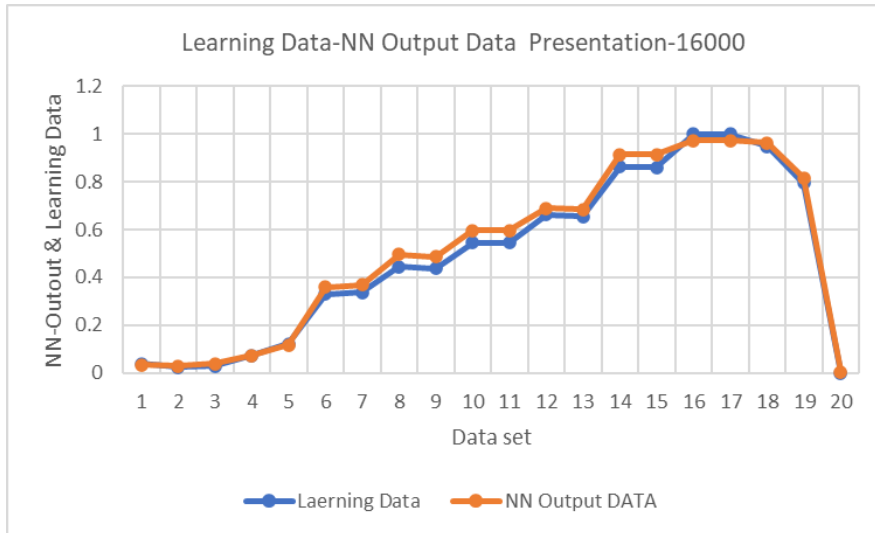
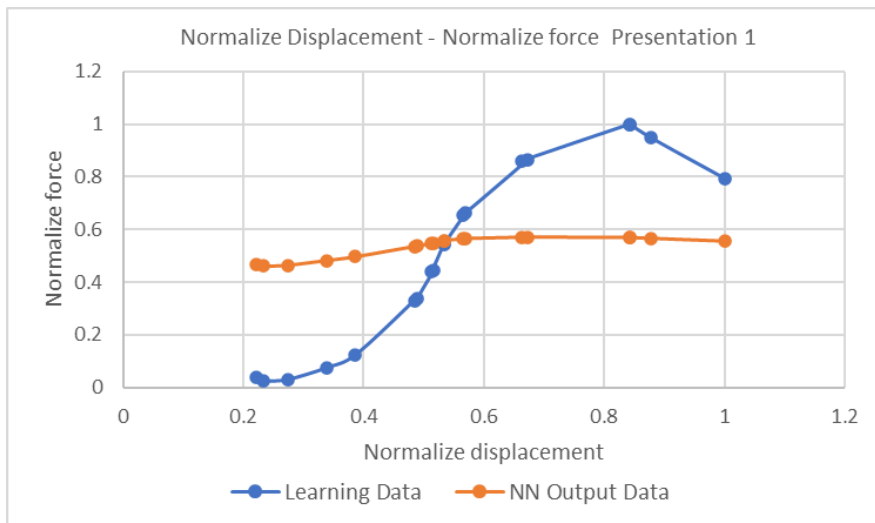Fig. 6. Predicted out from neural network at 16000 sampling training.

Fig. 7. Normalize displacement and force output at the initial of learning.
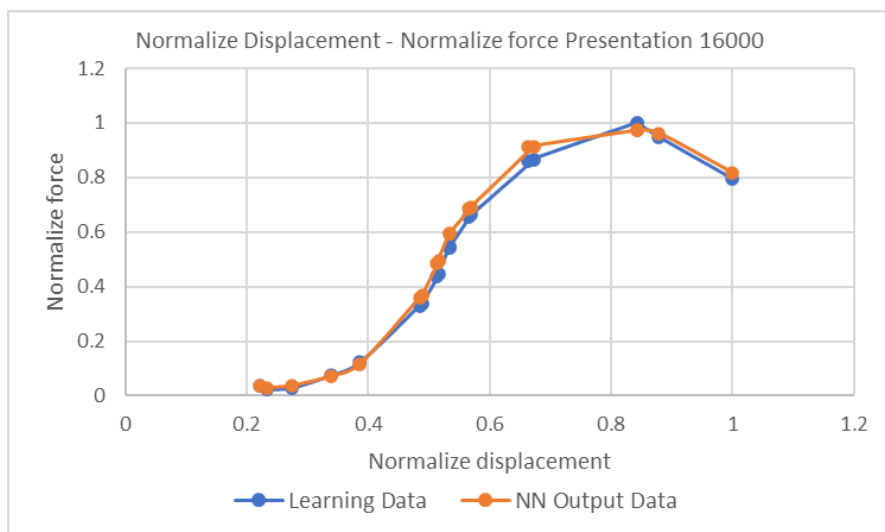

Fig. 8. Normalized displacement and normalized force output after 16000 presentation of training process.

After the learning procedure after 16000 presentations of the data set to the neural network the normalized output represents the NN output data. Now we need to convert this normalized output data into the standard representation of the capacity of a wall. For that purpose, the normalize displacement and force are computed using the $aXmax$ and $aZmax$ factors used on the normalization process presented in item 4. Here, the displacement is converted into a drift ratio (displacement/height) using the geometry of the specimen of H shape masonry wall [6] with a height of 2300 mm.

In Fig. 9. Is presented the NN output in a standard format of capacity curve for building structural investigations. This data corresponds to the first presentation of the learning data [6] to the network where we can read that at this presentation the NN data output is still in the early learning stage.

After the 16000 presentations of the learning data [6] to the artificial neural network the normalize data is converted into standard data representation and the capacity curve for the final stage of learning is presented on Fig.10, where a good agreement has been reached.
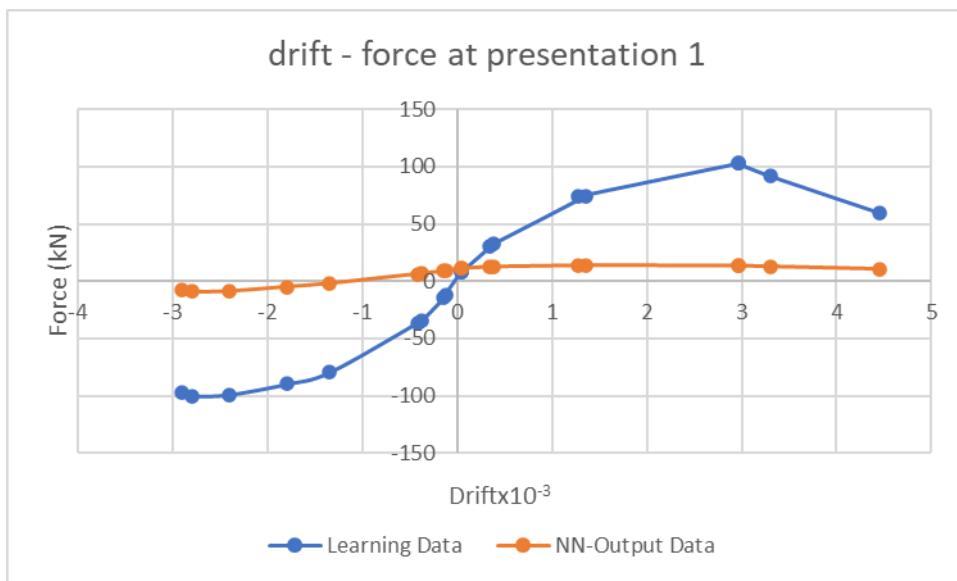

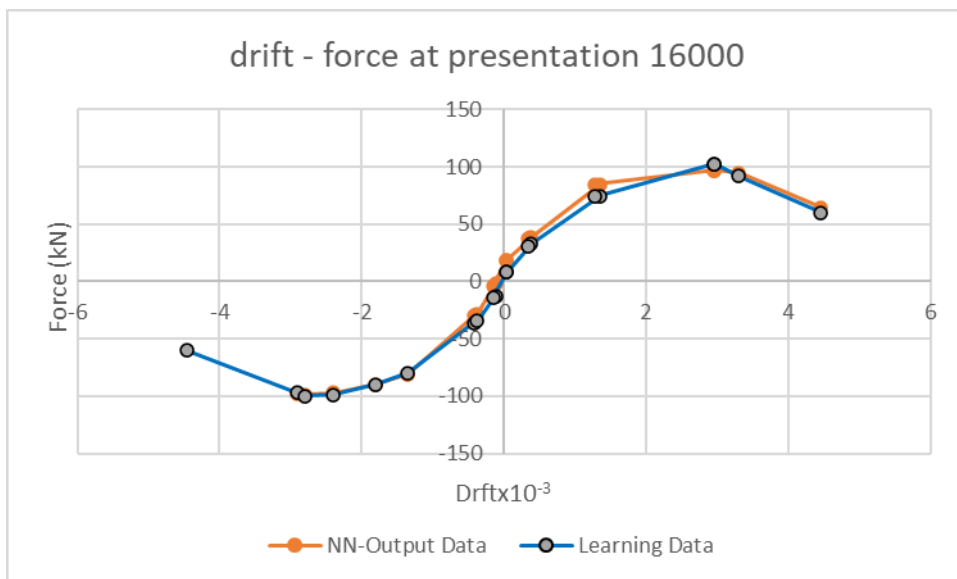Fig. 9. Drift ratio and force output at initial of learning.

Fig. 10. Drift ratio and force output after 16000 presentations of learning data

6. CONCLUSIONS

– The use of simple programing structural language as PYTHON, with libraries that include neural network generation, provides a wonderful tool for the young generation of researchers, especially the ones who try to produce IT software based on cloud or internet development. The example we present only intents to show that method can be used to solve any nonlinearity problem.

– The learning data in our trial program was a capacity curve data or skeleton curve for produce cyclic response from a masonry wall test, with good agreement and very fast time consuming. Old applications we developed in spark stations in the decade if 1990 took 2 days to developed similar process. With Python just less than 2 minutes to produce the learning of the network.

– The prediction of the capacity curve using an artificial neural network provide by a python subroutine, could be combined with other algorithms of artificial intelligence for generation of forest of data, and which is a challenge of the force response of the building against an earthquake scenario. A future study will combine multi-hazard approach (earthquake, tsunami, inundation, massive soil movement, etc.) with artificial neural networks using in benefit of the population of risk areas.

– There are many developments we intend to make; we hope time will help us to reach our goal in the development of an artificial brain to produce earthquake response on masonry buildings.

7. ACKNOWLEDGEMENT

8. REFERENCES

[1] F. Rosenblatt, et al., "The Perceptron – a Perceiving and Recognizing Automaton" (Project PARA), Report 85-460-1 Cornell Aeronautical Laboratory Inc. *Buffalo New York*, Jan. 1957

[2] D. Rumelhart, G. Hinton, and R. Williams, "Learning Internal representations by error propagation", ICS Report 8506, Cognitive Science, University of California San Diego, Sep. 1985 https://doi.org/10.1016/B978-1-4832-1446-7.50035-2

[3] D. Ackley, G. Hinton, and T. Sejnowski "A Learning Algorithm for Boltzmann Machines", *Cognitive Science,* vol. 9, 147-169, 1985 https://doi.org/10.1016/B978-0-08-051581-6.50053-2

[4] C. Zavala, K. Ohi, K. Takanashi, "Neural Network Model in Substructuring Hybrid Simulation", *Annual Meeting of Architectural Institute of Japan*, paper 21672, Mar. 1993.

[5] M. Diaz, et.al. Report on statistical analysis of target study Area in Lima Metropolitan and Callao. Project FONDECYT-CISMID-FIC UNI Development a digital tool for feasibility of confined masonry dwelling retrofitting in multi seismic scenarios based on assessment of vulnerability and risk.

[6] CISMID-UNI Report, Program PPO68, Product 02- Dissemination of progress in the design of earthquake resistant buildings and tsunamis (in Spanish), December 2013