M.A. El-Sharkawi and R.J. Marks II, "Can neural networks play a role in power system engineering?",
Proc. NSF/EPRI Workshop on Advanced Computing Applications to Power Engineering, Victoria, B.C., October 16-18, 1991.

Can Neural Networks Play a Role in Power System Engineering?

Mohamed A. El-Sharkawi   Robert J. Marks II
Department of Electrical Engineering
University of Washington
Seattle, WA 98195

Introduction

Neural networks (NNs) have been studied for many years with the hope of achieving human-like performance in solving certain problems in speech and image processing. There has been a recent resurgence in the field of NNs due to the introduction of new network topologies, training algorithms and VLSI implementation techniques. The potential benefits of NNs such as parallel distributed processing, high computation rates, fault tolerance, and adaptive capability have lured researchers from other fields such as controls, robotics, energy systems to seek NN solutions to their complicated problems.

Artificial Neural Networks are a method of synthesizing a mapping between input and output variables by learning a set of arc weights and node thresholds of a connectionist model based on training examples. They have been developed in a wide variety of configurations with some common underlying characteristics. They all attempt to achieve good performance through massive interconnection of simple computational elements [1-5].

An artificial neural network can be defined as a loosely connected array of elementary processors or neurons. Algorithms are then crafted about this architecture. Neurons are linked with interconnects analogous to the biological synapse. This highly connected array of elementary processors defines the system hardware. Commonly used neural networks, such as the layered perceptron, are said to be trained rather than programmed in the conventional sense.

Computationally, neural networks have the advantage of massive parallelism and are not restricted in speed by the von Neumann bottle neck characteristic of conventional computation. Neural networks, in most cases, are significantly fault tolerant.

At this writing, the layered perceptron is receiving the most attention as a viable candidate for application to power systems. The layered perceptron is taught by example, as opposed, for example, to an expert system, which is taught by rules. The preponderance of data typically available from the power industry, coupled with the ability of the layered perceptron to learn significantly nonlinear relationships, reveals it as a viable candidate in the available plethora of solutions for solving significant power systems engineering problems.

Hopfield neural networks have also been proposed for application to combinatorial search problems in the power industry. In Hopfield nets, each neuron is connected to every other neuron, The state of each neuron is determined by the state and interconnects weights of the other neurons. The state of one neuron may change, thereby changing another, etc., until the network reaches a steady state.

There exist problems with Hopfield neural networks. Their capacity has shown to increase less than linearly with the number of neurons. The number of false stable states has been shown to increase greater than linearly with the number of neurons. This, despite the fact the required number of interconnects grows as the square of the number of neurons.

The Layered Perceptron

Neural networks have been found to be effective systems for learning functional mappings from a body of examples. This is done by adjusting arc weights and node thresholds of a set of interconnected neurons according to a specific learning rule. The high degree of connectivity brings about desirable properties such as generalization, fault tolerance and noise rejection that are useful in any model.

Multi-layer perceptron (MLP) is a specific neural network architecture where sets of neurons are arranged in layers. The input and output layers surround hidden layers. Activation signals of neurons in one layer are transmitted to the next layer through a set of links that either attenuate or amplify the signals based on the respective weights [1-5].

The layered perceptron operates in two modes: training and testing. In the training mode, a set of representative *training data* is used to adjust the weights of the network interconnects. Once these weights have been determined, the neural network is said to be trained. In the test mode, the trained neural network is activated by *test data*. The response of the layered perceptron should then be representative of the data by which it was trained. Typically, the test and training data are different sets. However, it should be kept in mind that training a neural network to respond to the same data on which it is trained is not learning, but is, rather, memorization.

A layered perceptron can be used as either a classifier or a regression machine. As a classifier, the layered perceptron catagorizes the input into two or more categories. In power system security assessment, for example, the trained perceptron will categorized the system as either secure or insecure in accordance to the current system states [6,7]. For regression applications, the output of the layered perceptron takes a continuous value. Electric load forecasting is an example of regression application [8-11].

Learning algorithms

A neural network can be treated, from the training sense, as a model with unknown parameters. The existing technologies of parameter identification can be utilized in the learning process. The only modification that might be employed is to tune the identification algorithm to allow for speedy training.

Most of the parameter identification algorithms require a cost index as a measure of the accuracy of model being identified. For neural network, the cost index is usually a least squares' function that represents the accumulated difference between the desired response and the NN output.

Once the cost index is defined, a convergence algorithm can be utilized. The back error propagation technique, which is the most common method, is a derivative of the steepest descent method. This is probably why training a neural network is very fast in the beginning and slow when the cost index is small. Other training algorithms proposed in the literature include the conjugate gradient descent, random search and adaptive learning [12].

Supervised versus unsupervised learning

The layered perceptron is trained using supervised learning. The perceptron is told of the desired output for each input pattern. Unsupervised learning, on the other hand, does not require knowledge of the output. The classifier, rather, looks for similarity of structure in input patterns and groups them accordingly. The most visible of neural network's paradigms using unsupervised learning is *adaptive resonance training* that exists in various forms [3]. In general, a classifier will train better when supervised than when not. Unsupervised learning, on the other hand, can be the only available option in some scenarios.

Learning versus memorization

There is a difference between *training* and *memorization* [8]. A trained classifier or regression machine can respond with confidence to a pattern that it has not seen before. The ability to properly classify data that has not been seen before is referred to as *generalization*. Memorization, on the other hand, guarantees that, when presented with a specific element in the training data set, the classifier will respond in exactly the same manner that it was trained. In the case of memorization, the response to data other than training data is not considered in the paradigm.

The ability to interpolate among the training data does not necessarily imply good generalization. A properly trained classifier or regression machine should respond with the same error to training data as to test data. This is a necessary not sufficient condition. If the error from the test data is much higher than that from the training data, then, chances are, the neural system is over determined. In other words, the degrees of freedom in the classifier or regression machine is too high.

Determining the best net size

The degree of freedom of the neural network, equal to the number of interconnects and therefore proportional to the number of hidden neurons, must be matched, in some sense, to the complexity of the classification boundary.

Currently, in the absence of parametric guidance, the only proposed method of determining the best number of hidden neurons is by comparative cross validation among two or more neural networks. Moving from a small number of hidden neurons to a large number must decrease the overall probability of error while maintaining an equivalent error performance for the test and training data. When the perceptron's performance on training data begins to lag, the process of memorization may have started.

Problems with the layered perceptron

Certain problems' characteristics to the layered perceptron are problems of the problem. The challenges would be encountered with all classifiers or regression machines trained by example. Other problems are specific to the layered perceptron such as the following:.

Problems with back error propagation Although back error propagation is the most widely used method to train multi-layer perceptrons, it in not the only nor necessarily the best approach. Indeed, most any algorithm that searches for a minimum can be used to train a layered perceptron. Back propagation is attractive because it can be performed within the neural network structure. However, the technique has a number of limitations. For example, since the back error propagation technique is not designed to be adaptive, all data must be used every time the weights are updated. If a set of old data becomes irrelevant, the NN is retrained by using the entire new data set. Also, when new data is in conflict with old data (data inconsistency), the effect of old data can not be removed unless the NN is retrained without the old data. The importance of some data can not be easily weighted. In addition, If the size of the NN is not adequately selected, or the convergence criterion is not realistic, thousands of iterations can be required to train a layered perceptron on even a simple problem.

Scaling: The scaling problem can be illustrated through the *curse of dimensionality*. Specifically, for a problem of similar partition complexity, the required cardinality of the training data set grows exponentially with respect to the number of input nodes.

Diminished learning *The more you learn, the harder it is to learn.* Indeed, in the absence of data noise, additional learning takes place in a multilayered perceptron only if new data is introduced that the neural network improperly classifies. The closer the representation comes to the concept, the smaller the chance that this happens. This is a characteristic of least squares and steepest descent techniques.

Query and Inverted NN

In supervised learning, each feature vector is assigned a classification (or regression) value or values. There is usually a cost associated with this assignment, such as the cost of performing an experiment, computational overhead or simply time. This process is similar to presenting to an *oracle* the feature vector. For a cost, the oracle will reveal to us the proper classification or regression value associated with that vector.

Another approach to query based learning is, in effect, to ask a partially trained classifier or regression machines "What is it you don't understand?" The response of the classifier or regression machine is taken to the oracle for proper categorization and the result is added to the training data set. The classifier is then further trained and the process repeated.

How might we apply this query approach to, say, a trained layered perceptron classifier with a single output? If the output neuron is thresholded at one half to make the classification decision, the representation boundary in feature vector space is the locus of all inputs that produce an output of one half. This locus of points corresponds to feature vectors of maximum confusion. In other words, when presented with such a vector, the neural network is uncertain to the corresponding classification. If there were a technique to find a number of these points, they could be taken to the oracle to clear the confusion. The data from the oracle could then be used for training data. The perceptron can then be retrained to yield a higher accuracy. The question is, how can the locus of confusion be generated? The answer is through inversion of the neural network [13-15].

Examples of Reported Applications of NN to Power System

Neural network applications that have been proposed in the literature up to date can be broadly categorized under three main areas: Regression, Classification and Combinatorial Optimization. The applications involving regression includes Transient Stability[16-18], Load forecasting [8-11], Synchronous machine modelling [19], Contingency screening [20,21], Harmonic evaluation [22] and control [23-26]. Applications involving classification include Harmonic load identification [27,28], Alarm processing [29-31], Static security assessment [6,32] and Dynamic security assessment [7]. In the area of combinatorial optimization, there is topological observability [33-35] and capacitor control [36] Most of these, and other applications, are also reported in the *First International Forum on Applications of Neural Networks to Power System* [37]. The Forum proceedings include other promising applications such as fault diagnosis [38], protection [39,40], dynamic load modeling [41], power quality [42,43], training simulator [44], unit commitment [45], economic dispatch [46] and reactive power control [47].

References

[1] Mohamed A. El-sharkawi, Robert J. Marks and Siri Weerasooriya, *Neural networks and their application to power engineering*, in Advances in Control and Dynamic Systems, ed C.T. Leondes, vol 41, Academic press, (in print).

[2] Y.H. Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison Wesley, 1989.

[3] J.A. Anderson and E. Rosenfeld, Eds., *Neurocomputing: Foundations of Research*, MIT Press, Cambridge, MA, 1988.

[4] R.C. Eberhart and R.W. Dobbins, *Neural Network PC Tools: a Practical Guide*, Academic Press, 1990.

[5] J. Dayhaff, *Neural Network Architectures, An Introduction*, Van Nostrand Reinhold, 1990.

[6] Siri Weerasooriya and M. A. El-Sharkawi, "Use of Karhunen-Loe've Expansion in Training Neural Networks for Static Security Assessment," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991, pp. 59-64.

[7] A. B. R. Kumar, A. Ipakchi, V. Brandwajn, M. A. El-Sharkawi and G. Cauley, "Neural Networks for Dynamic Security Assessment of Large-Scale Power Systems: Requirements Overview," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991, pp. 65 - 71.

[7] M. Aggoune, M. A. El-Sharkawi, D.C. Park, M.J. Damborg and R.J. Marks II, "Preliminary Results of Neural Networks for Security Assessment," IEEE Transactions on Power Systems, pp.890-896, May 1991.

[8] M. A. El-Sharkawi, S. Oh, R. J. Marks, M. J. Damborg and C. M. Brace, "Short Term Electric Load Forecasting Using an Adaptively Trained Layered Perceptrone," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991, pp.3-6.

[9] D. C. Park, M. A. El-Sharkawi, R. J. Marks, L. E. Atlas and M. J. Damborg, "Electric Load Forecasting Using An Artificial Neural Network," IEEE Transactions on Power Systems,, pp. 442-449, May 1991.

[10] D.C. Park and O. Mohammed, M.A. El-Sharkawi and R.J. Marks II, "An Adaptively Trainable Neural Network Algorithm and Its Application to Electric Load Forecasting," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[11] J.T. Conner, L.E. Atlas and D. Martin, "Recurrent Neural Networks and Load Forecasting,," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[12] D. C. Park, M. A. El-Sharkawi and R. J. Marks, "Adaptively Trained Neural Network," IEEE Transactions on Neural Networks, May 91, pp. 334-345.

[13] Seho Oh, Robert J. Marks II and M. A. El-Sharkawi, "Query Based Learning in a Multilayered Perceptrone in the Presence of Data Jitter," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991, pp. 72-75.

[14] J.N. Hwang, S. Oh, J.J. Choi & R.J. Marks II, "Classification boundaries and gradients of trained multilayer perceptrons", *Proc. 1990 IEEE International Symposium on Circuits and Systems*, 1-3 May, 1990, New Orleans, Louisiana.

[15] J.N. Hwang, J.J. Choi, S. Oh and R.J. Marks II, "Query based learning applied to partially trained multilayer perceptrons", *IEEE Transactions on Neural Networks* (in press).

[16] D. J. Sobajic and Y. Pao, "Artificial Neural Net Based Dynamic Security Assessment for Electric Power System," IEEE Transactions on Power Systems, Feb, 1989, pp. 220-228.

[17] D. Lubkeman, C. Fallon and A. Girgis, "Unsupervised Learning Strategies for the Detection and Classification of Transient Phenomena on Electric Power Distribution Systems," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[18] H. Mori, "An Artificial Neural-Net Based Method for Estimating Power System Dynamic Stability Index," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[19] M. Chow and R.J. Thomas, "Neural networks synchronous machine modeling," *Proc of the 1989 ISCAS* vol. 1, pp 495-498, Portland, OR, May, 1989

[20] R. Fischl, M. Kam, J.C. Chow, and S. Ricciardi, "Screening power system contingencies using a back-propagation trained multiperceptron," *Proc of the 1989 ISCAS*, vol. 1, pp 486-489, Portland, OR, May, 1989.

[21] H.H. Yan, J-C Chow, M. Kam, R. Fischl, and C.R. Sepich, "Hybrid Expert System/Neural Network Hierarchical Architecture for Classifying Power System Contingencies," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991,

[22] H. Mori, H. Uematsu, S. Tsuzuki, T. Sakarai, Y. Kojima and K. Suzuki, "Identification of harmonic loads in power systems using an artificial neural network," 2nd Symposium on Expert System Applications to Power Systems, Seattle, July, 1989.

[23] S. Weerasooriya and M.A. El-Sharkawi, "Identification and control of a DC motor using a back-propagation neural network," Paper No. 91 WM 288-1 EC, IEEE-PES Winter Meeting, New York, February, 1991.

[24] R.J. Thomas and E. Sakk, "Dynamic Implications of Using Neural Networks as Controllers," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[25] K. Saitoh and S. Iwamoto, "Application of Neural Network Based Fuzzy Control to Power System Generator," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[26] H. Saitoh, Y. Shimotori and J. Toyoda, "Neural Network Based Preventive Control Support System for Power System Stability Enhancement," First International Forum on

Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[28] H. Mori and S. Tsuzuki, "Comparison between backpropagation and revised GMDH techniques for predicting voltage harmonics," *Proc of the 1990 ISCAS* vol. 2, pp 1102-1105, New Orleans, LA, May, 1990.

[29] P. Chan, "Application of neural-network computing in intelligent alarm processing,"*PICA conference proceedings*, Seattle, WA, May, 1989.

[30] R. Karunakaran and G. Karady, "Artificial Neural Network as a Dispatchers' Aid in Alarm Processing," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[31] R. Adapa and E. Chan, "Diagnostic Alarm Processor," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[32] D. Niebur and A.J. Germond, "Power Flow Classification for Static Security Assessment," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991,

[33] H. Mori and S. Tsuzuki, "Power system topological observability analysis using a neural network model," 2nd Symposium on Expert System Applications to Power Systems, Seattle, July, 1989.

[34] H. Mori and S. Tsuzuki, "Determination of power system topological observability using the boltzman machine," *Proc of the 1990 ISCAS* vol. 3, pp 2938-2941, New Orleans, LA, May 1990.

[35] H. Mori, "Application of a Revised Boltzmann Machine to Topological Observability Analysis," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[36] N.I. Santoso and O.T. Tan, "Neural net based real-time control of capacitors installed on distribution systems," 89 SM 768-3 PWRD, IEEE Power Engineering Society, Summer Meeting, Long Beach, California, 1989.

[37] Proceedings of the First International Forum on Applications of Neural Networks to Power Systems, Seattle, Washington, July 23-26, 1991, IEEE Catalog # 91TH0374-9.

[38] H. Ogi, H. Tanaka, Y. Akimoto and Y. Izui, "Fault Diagnosis System for GIS Using An Artificial Neural Networks," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991,

[39] K. Feser, U. Braun, F. Engler and A. Maier, "Application of Neural Networks in Numerical Busbar Protection Systems (NBPS)," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991,

[40] S.A. Khaparde, P.B. Kale and S.H. Agarwal, "Application of Artificial Neural Network in Protective Relaying of Transmission Lines," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991,

[41] R.J. Thomas and B-Y Ku, "Approximation of Power System Dynamic Load Characteristics By Artificial Neural Networks," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[42] R. Daniels, "Power Quality Monitoring Using Neural Networks," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[43] B. Jayaraman, M. Durham, K. Ashenayi and R. Strattan, "Neural Net Based Correction Of Power System Distortion Caused By Switching Power Supplies," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[44] Y-H Pao, D.J. Sobajic, "A Perspective on Use of Neural-Net Computing in Training Simulator Design," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[45] M.H. Sendaula and S.K. Biswas, A. Eltom and W. Kazibwe, "Application of Artificial Neural Networks to Unit Commitment," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[46] Y. Fukuyama and Y. Ueki, "An Application of Artificial Neural Network to Dynamic Economic Load Dispatching," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.

[47] G. Neily, R. Barone, G. Josin and D. Charney, "Joint VAR Controller Implemented in an Artificial Neural Network Environment," First International Forum on Applications of Neural Networks to Power Systems, Seattle, July 23 - 26, 1991.