A hierarchical hybrid neural model in short-term load forecasting

Otávio A. S. Carpinteiro Agnaldo J. R. Reis
Paulo S. Quintanilha Filho
Research Group on Computer Networks and Software Engineering
Federal University of Itajubá
37500–903, Itajubá, MG, Brazil
{otavio,agnreis,pquinta}@iee.efei.br

Abstract

This paper proposes a novel neural model to the problem of short-term load forecasting. The neural model is made up of two self-organizing map nets — one on top of the other —, and a single-layer perceptron. It has application into domains in which the context information given by former events plays a primary role. The model was trained and assessed on load data extracted from a Brazilian electric utility. It was required to predict once every hour the electric load during the next six hours. The paper presents the results, and evaluates them.

1 Introduction

With power systems growth and the increase in their complexity, many factors have become influential to the electric power generation and consumption (e.g., load management, energy exchange, spot pricing, independent power producers, non-conventional energy, generation units, etc.). Therefore, the forecasting process has become even more complex, and more accurate forecasts are needed. The relationship between the load and its exogenous factors is complex and nonlinear, making it quite difficult to model through conventional techniques, such as time series and linear regression analysis. Besides not giving the required precision, most of the traditional techniques are not robust enough. They fail to give accurate forecasts when quick weather changes occur. Other problems include noise immunity, portability and maintenance [11].

Neural networks (NNs) have succeeded in several power system problems, such as planning, control, analysis, protection, design, load forecasting, security analysis, and fault diagnosis. The last three are the most popular [17]. The NN ability in mapping complex non-linear relationships is responsible for the growing number of its application to the short-term load forecasting (STLF) [14, 1, 16, 19]. Several electric utilities over the world have been applying NNs for load forecasting in an experimental or operational basis [11, 17, 1].

So far, the great majority of proposals on the application of NNs to STLF use the multilayer perceptron (MLP) trained with error backpropagation. Besides the high computational burden for supervised training, MLPs do not have a good ability to detect data outside the domain of the training data.

This paper introduces a new hierarchical hybrid neural model (HHNM) to STLF. The HHNM is an extension of the Kohonen's original self-organizing map [12]. Several researchers have extended the Kohonen's self-organizing feature map model to recognize sequential information. The problem involves either recognizing a set of sequences of vectors in time or recognizing subsequences inside a large and unique sequence.

Several approaches, such as windowed data approach [9], time integral approach[1] [6], and specific approaches [8] have been proposed in the literature. Many of these approaches have well-known deficiencies [4]. Among all, loss of context is the most serious.

The proposed model is a hierarchical model. The hierarchical topology yields to the model the power to process efficiently the context information embedded in the input sequences. The model does not suffer from loss of context. On the contrary, it holds a very good memory for past events, enabling it to produce better forecasts. It has been applied to load data extracted from a Brazilian electric utility.

This paper is divided as follows. The second section provides an overview of related research. The third section presents the data representation. The HHNM

[1] Also known as leaky integral approach.

is introduced in the fourth section. The fifth section describes the experiments, and discusses the results. The last section presents the main conclusions of the paper, and indicates some directions for future work.

2 Related research

The importance of short-term load forecasting has been increasing lately. With deregulation and competition, energy price forecasting has become a valuable business. Bus-load forecasting is essential to feed analytical methods utilized for determining energy prices. The variability and non-stationarity of loads are becoming critical owing to the dynamics of energy prices. In addition, the number of nodal loads to be predicted does not allow frequent interactions with load forecasting experts. More autonomous load predictors are needed in the new competitive scenario.

Artificial neural networks (NNs) have been successfully applied to short-term load forecasting (STLF). Many electric utilities, which had previously employed STLF tools based on classical statistical techniques, are now using NN-based STLF tools.

Park et al. [19] have successfully introduced an approach to STLF which employs a NN as main part of the forecaster. The authors employed a feed-forward neural network trained with the standard error back-propagation algorithm (EBP). Three NN-based predictors have been developed and applied to short-term forecasting of daily peak load, total daily energy, and hourly daily load, respectively. Three months of actual load data from Puget Sound Power and Light Company have been used in order to test the aforementioned forecasters. Only ordinary weekdays were taken into consideration for the training data.

Another successful example of NN-based STLF can be found in Lee et al. [13]. The authors employed a multilayer perceptron (MLP) trained with EBP to predict the hourly load for a lead time of 1–24 hours. Two different approaches have been considered, namely one-step ahead forecasting (named static approach), and 1–24 steps ahead (named dynamic approach). In both cases, the load was separated in weekday (Tuesdays through Fridays) and weekend loads (Saturdays through Mondays).

Bakirtzis et al. [1] employed a single fully connected NN to predict, on a daily basis, the load along a whole year for the Greek power system. The authors made use of the previous year for training purposes. Holidays were excluded from the training set and treated separately. The network was retrained daily using a moving window of the 365 most recent input/output patterns. More, the paper proposed another procedure to 2–7 days ahead forecasting.

Papalexopoulos et al. [18] compared the performance of a sophisticated regression-based forecasting model to a newly developed NN-based model for STLF. It is worth mentioning that the regression model had been in operation in a North-American utility for several years, and represented the state-of-art in the classical statistical approach to STLF. The NN-based model has outperformed the regression model, yielding better forecasts. Moreover, the development time of the neural model was shorter, and the development costs lower in comparison to the regression model. As a consequence, the neural model has replaced the regression model. This report is important, for it evaluates the operation of a neural model in a realistic electrical utility environment.

Khotanzad et al. [10] describe the third generation of an hourly short-term load forecasting system, named artificial neural network short-term load forecaster (ANNSTLF). Its architecture includes only two neural forecasters — one forecasts the base load, and the other predicts the change in load. The final prediction is obtained via adaptive combination of these two forecasts. A novel scheme for forecasting holiday loads is developed as well. The performance on data from ten different utilities is reported and compared to the previous generation forecasting system.

Finally, a comprehensive review of the application of NNs to STLF can be found in Hippert et al. [7]. The authors examine a collection of papers published between 1991 and 1999.

3 Data representation

The input data consisted of sequences of load data extracted from a Brazilian electric utility. Seven neural input units are used in the representation, as shown in table 1. The first unit represents the load at the current hour. The second, the load at the hour immediately before. The third, fourth and fifth units represent respectively the load at twenty-four hours behind, at one week behind, and at one week and twenty-four hours behind the hour whose load is to be predicted. The sixth and seventh units represent a trigonometric coding for the hour to be forecast, i.e., $sin(2\pi.hour/24)$ and $cos(2\pi.hour/24)$. Each unit receives real values. The load data is preprocessed using ordinary normalization (minimum and maximum values in the [0,1] range).

Table 1. Input variables for the HHNM model

Input	Variable name	Lagged values (h)
1–5	Load(P)	1, 2, 24, 168, 192
6	HS	0*
7	HC	0*

*Lag 0 represents the hour to be forecast

4 The HHNM

The hierarchical hybrid neural model (HHNM) is shown in figure 1. It is made up of two distinct neural models — a hierarchical self-organizing model (HSOM), and a single-layer perceptron (SLP). The HSOM, in its turn, is made up of two self-organizing maps (SOMs). The HSOM features, performance, and potential are better evaluated in [2, 3].

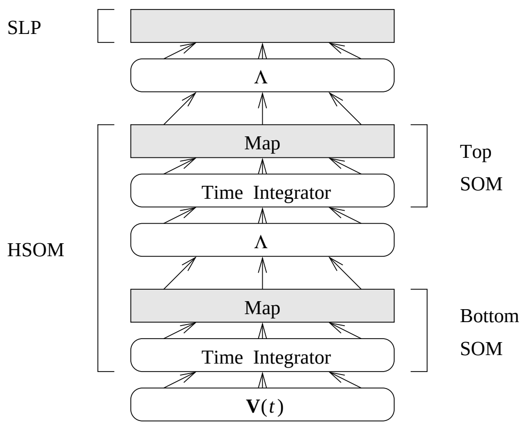

Figure 1. HHNM

The input to the model is a sequence in time of m-dimensional vectors, $\mathbf{S_1} = \mathbf{V}(1), \mathbf{V}(2), \ldots, \mathbf{V}(t), \ldots, \mathbf{V}(z)$, where the components of each vector are real values. The sequence is presented to the input layer of the bottom SOM, one vector at a time. The input layer has m units, one for each component of the input vector $\mathbf{V}(t)$, and a time integrator[2]. The activation $\mathbf{X}(t)$ of the units in the input layer is given by

$$\mathbf{X}(t) = \mathbf{V}(t) + \delta_1 \mathbf{X}(t-1) \quad (1)$$

where $\delta_1 \in (0,1)$ is the decay rate. For each input vector $\mathbf{X}(t)$, the winning unit $i^*(t)$ in the map is the unit which has the smallest distance $\Psi(i,t)$. For each output unit i, $\Psi(i,t)$ is given by the Euclidean distance between the input vector $\mathbf{X}(t)$ and the unit's weight vector $\mathbf{W}_i$.

[2]Time integrators act as memories for past events.

Each output unit i in the neighbourhood $N^*(t)$ of the winning unit $i^*(t)$ has its weight $\mathbf{W}_i$ updated by

$$\mathbf{W}_i(t+1) = \mathbf{W}_i(t) + \alpha \Upsilon(i)[\mathbf{X}(t) - \mathbf{W}_i(t)] \quad (2)$$

where $\alpha \in (0,1)$ is the learning rate. $\Upsilon(i)$ is the *neighbourhood interaction function* [15], a Gaussian type function, and is given by

$$\Upsilon(i) = \kappa_1 + \kappa_2 e^{-\frac{\kappa_3[\Phi(i,i^*(t))]^2}{2\sigma^2}} \quad (3)$$

where κ_1, κ_2, and κ_3 are constants, σ is the radius of the neighbourhood $N^*(t)$, and $\Phi(i,i^*(t))$ is the distance in the map between the unit i and the winning unit $i^*(t)$. The distance $\Phi(i',i'')$ between any two units i' and i'' in the map is calculated according to the maximum norm,

$$\Phi(i',i'') = max\left\{|l'-l''|, |c'-c''|\right\} \quad (4)$$

where (l',c') and (l'',c'') are the coordinates of the units i' and i'' respectively in the map.

The input to the top SOM is determined by the distances $\Phi(i,i^*(t))$ of the n units in the map of the bottom SOM. The input is thus a sequence in time of n-dimensional vectors, $\mathbf{S_2} = \Lambda(\Phi(i,i^*(1))), \Lambda(\Phi(i,i^*(2))), \ldots, \Lambda(\Phi(i,i^*(t))), \ldots, \Lambda(\Phi(i,i^*(z)))$, where Λ is a n-dimensional *transfer function* on a n-dimensional space domain. Λ is defined as

$$\Lambda(\Phi(i,i^*(t))) = \begin{cases} 1 - \kappa\Phi(i,i^*(t)) & \text{if } i \in N^*(t) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where κ is a constant, and $N^*(t)$ is a neighbourhood of the winning unit.

The sequence $\mathbf{S_2}$ is then presented to the input layer of the top SOM, one vector at a time. The input layer has n units, one for each component of the input vector $\Lambda(\Phi(i,i^*(t)))$, and a time integrator. The activation $\mathbf{X}(t)$ of the units in the input layer is thus given by

$$\mathbf{X}(t) = \Lambda(\Phi(i,i^*(t))) + \delta_2 \mathbf{X}(t-1) \quad (6)$$

where $\delta_2 \in (0,1)$ is the decay rate.

The dynamics of the top SOM is identical to that of the bottom SOM.

The input to the SLP is also determined by the distances $\Phi(i,i^*(t))$ of the p units in the map of the top SOM. The input is thus a sequence in time of p-dimensional vectors, $\mathbf{S_3} = \Lambda(\Phi(i,i^*(1))), \Lambda(\Phi(i,i^*(2))), \ldots, \Lambda(\Phi(i,i^*(t))), \ldots, \Lambda(\Phi(i,i^*(z)))$, where Λ is a p-dimensional *transfer function* on a p-dimensional space domain, and is given by equation 5 as well.

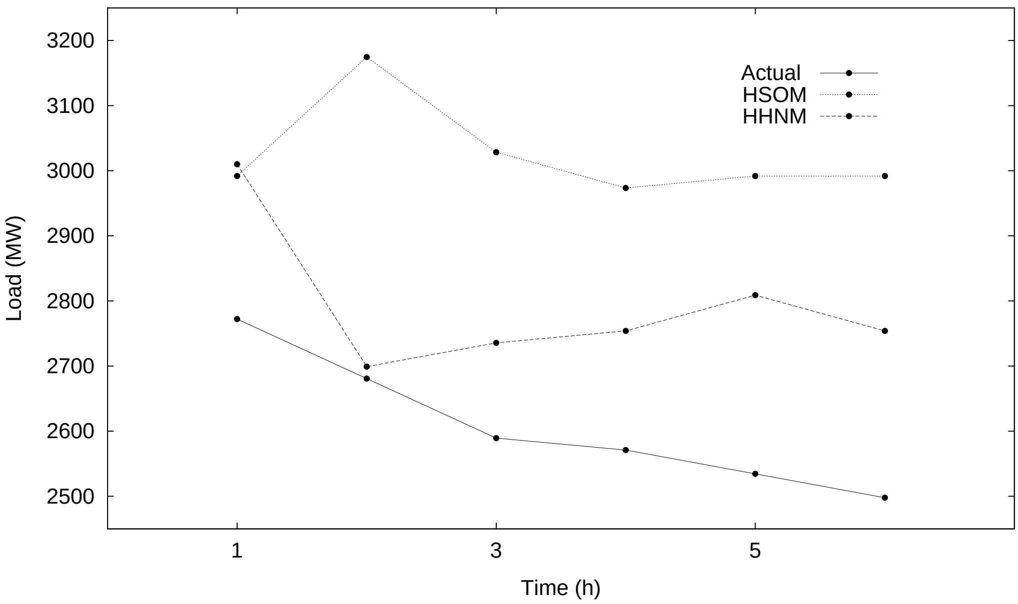

Figure 2. Actual load and forecast loads for February 05, 1995

The sequence $\mathbf{S_3}$ is then presented to the input layer of the SLP, one vector at a time. The input layer has p units, one for each component of the input vector $\Lambda(\Phi(i, i^*(t)))$. The SLP is trained with the usual delta rule [21, 20].

5 Experiments

The HHNM is required to foresee the time horizon from the first to the sixth hour. The training of its two SOMs takes place in two phases — coarse-mapping and fine-tuning. In the coarse-mapping phase, the learning rate and the radius of the neighbourhood are reduced linearly whereas in the fine-tuning phase, they are kept constant. The bottom and top SOMs were trained respectively with map sizes of 15×15 in 700 epochs, and 18×18 in 850 epochs. It was given the values 0.4 and 0.7 to decay rates for the bottom and top SOMs, respectively. Several map sizes and decay rates were tested. The initial weights were given randomly to both SOMs.

The SLP holds a single unit in its output layer. Training was halted when the mean pattern error was 0.005. It was given the values 0.0001 and 0.9 to learning rate and momentum respectively.

The training set comprised 2160 load patterns, spanning ninety days. They were taken from November 1994 to January 1995. The maximum electric load fell around 3900 MWatts. There was no particular treatment for holidays.

Carpinteiro et al. [5] came up with two main conclusions. First, the performance of HSOM[3] in STLF is much superior than that of the multilayer perceptron (MLP). Second, the output mapping process employed on HSOM holds two weaknesses. These weaknesses resulted from the output mapping process employed. In this paper, we propose HHNM to avoid such weaknesses by using a SLP model to perform the output mapping process.

The forecasts were then performed on the HSOM and HHNM models. A comparison of both models was carried out to verify whether or not HHNM outperformed HSOM.

Figures 2 and 3 show the actual load and forecast load for two particular days. The first one — Sunday, February 05, 1995 — is a typical weekday, and the second one — Tuesday, February 07, 1995 — is a special weekday.

A typical weekday is one whose load patterns share some similarity with the load patterns of the same weekdays in former weeks. For instance, the load patterns for Tuesdays tend to display a similar behaviour. Yet, when an unexpected event, such as a holliday, happens on one of those Tuesdays, it changes that fairly stationary behaviour. Such holliday is then said to be a special weekday. Special weekdays break down forecasters, for they perform much better on typical than on special weekdays.

[3] In Carpinteiro et al. [5], HSOM was referred to as HNM.

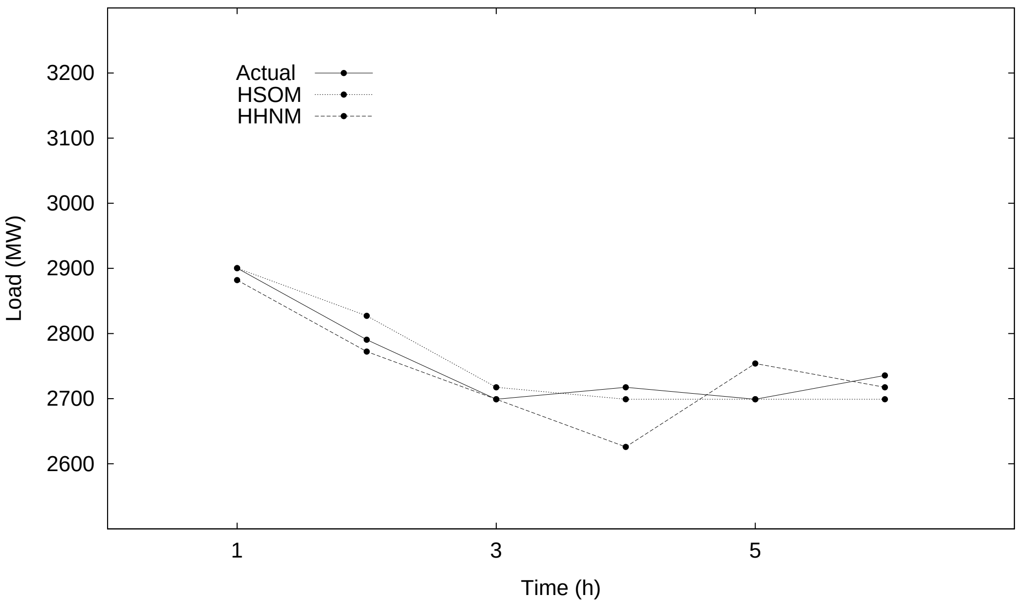

Figure 3. Actual load and forecast loads for February 07, 1995

Tables 2 and 3 present the performance of the forecasters for one to six step ahead predictions on those weekdays, as well as the mean absolute percentage error (MAPE).

Table 2. Hourly percentage error for February 05, 1995

Time (h)	HSOM	HHNM
1	*7.92*	8.58
2	18.43	*0.68*
3	16.96	*5.65*
4	15.65	*7.12*
5	18.05	*10.83*
6	19.77	*10.25*
MAPE (%)	16.13	*7.19*

The results from the HHNM are very promising. On the special day, the performances of HHNM and HSOM are similar. The results presented in figure 3 and table 3 show that HHNM yielded three better hourly percentage errors, and three worse percentage errors than HSOM.

On the typical day, as shown in figure 2 and table 2, the performance of HHNM was significantly superior than that of HSOM. The hourly percentage errors yielded by HHNM were much better than that yielded by HSOM.

The superior performance displayed by HHNM seems to be justified by its superior capacity to map output produced by the top SOM. As subsequent predictions are based on the former ones, the enhanced mapping process provided by SLP yields to HHNM an overall higher performance.

Table 3. Hourly percentage error for February 07, 1995

Time (h)	HSOM	HHNM
1	*0.00*	-0.63
2	1.31	*-0.66*
3	0.68	*0.00*
4	*-0.67*	-3.37
5	*0.00*	2.03
6	-1.34	*-0.67*
MAPE (%)	*0.67*	1.23

The forecasting errors were fairly high, however, even for the HHNM model. The load patterns were divided into seven groups, each one corresponding to a specific weekday. An analysis of those groups of patterns was then performed. It was observed that the training patterns within each group did not share much similarity between themselves. More, the difference was significant when comparing them with the testing patterns. Another Brazilian electric utility was contacted to provide us with more relevant and enlarged sequences of load data.

6 Conclusion

The paper presents a novel artificial neural model to the problem of short-term load forecasting. The model has a topology made up of two self-organizing map networks — one on top of the other —, and a single-layer perceptron. It encodes and manipulates context information effectively.

Some conclusions may be drawn from the experiments. First, the knowledge representation proposed for the HHNM inputs seems to be adequate. It supplied the model with the necessary information to make it produce correct predictions.

Second, the HHNM performance on the forecasts was better than that of the HSOM, which, in its turn, is much better than that of the MLP. The results obtained have shown that the HHNM was able to perform efficiently the prediction of the electric load in short forecasting horizons.

Third, it is worth mentioning that MLP has been widely employed to tackle the problem of STLF so far. The results obtained thus suggest that HHNM may offer a better alternative to approach such problem.

A research and development project for a Brazilian electric utility is under course. The research will focus on the effects of the HHNM time integrators on the predictions in order to produce a better adaptability. Besides, it will focus on the study of its performance on larger load databases. The forecasts should also span a larger number of days in order to be more significant statistically.

Acknowledgements

This research is supported by CNPq, Brazil.

References

[1] A. Bakirtzis, et al. A neural network short-term load forecasting model for the Greek power system. *IEEE Trans. on Power Systems*, 11(2):858–863, May 1996.

[2] O. A. S. Carpinteiro. A hierarchical self-organizing map model for pattern recognition. In *Proceedings of the Brazilian Congress on Artificial Neural Networks (CBRN)*, pages 484–488, Florianópolis, Brazil, 1997.

[3] O. A. S. Carpinteiro. A hierarchical self-organizing map model for sequence recognition. In *Proceedings of the ICANN*, pages 815–820, Skövde, Sweden, 1998.

[4] O. A. S. Carpinteiro. A hierarchical self-organizing map model for sequence recognition. *Pattern Analysis & Applications*, 3(3):279–287, 2000.

[5] O. A. S. Carpinteiro, A. Reis, and A. da Silva. A hierarchical neural model in short-term load forecasting. *Applied Soft Computing*, 4:405–412, 2004.

[6] G. J. Chappell and J. G. Taylor. The temporal Kohonen map. *Neural Networks*, 6:441–445, 1993.

[7] H. Hippert, C. Pedreira, and R. Souza. Neural networks for short-term load forecasting: A review and evaluation. *IEEE Trans. on Power Systems*, 16(1):44–55, Feb 2001.

[8] D. L. James and R. Miikkulainen. SARDNET: a self-organizing feature map for sequences. In *Proceedings of the NIPS*, volume 7, 1995.

[9] J. Kangas. *On the Analysis of Pattern Sequences by Self-Organizing Maps*. PhD thesis, Helsinki University of Technology, Finland, 1994.

[10] A. Khotanzad, R. Afkhami-Rohani, and D. Maratukulam. ANNSTLF – artificial neural network short-term load forecaster – generation three. *IEEE Trans. on Power Systems*, 13(4):1413–1422, Nov 1998.

[11] A. Khotanzad, R. Hwang, A. Abaye, and D. Maratukulam. An adaptive modular artificial neural hourly load forecaster and its implementation at electric utilities. *IEEE Trans. on Power Systems*, 10(3):1716–1722, August 1995.

[12] T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, third edition, 2001.

[13] K. Lee, Y. Cha, and J. Park. Short-term load forecasting using an artificial neural network. *IEEE Trans. on Power Systems*, 7(1):124–132, Feb 1992.

[14] K. Liu, S. Subbarayan, R. Shoults, M. Manry, C. Kwan, F. Lewis, and J. Naccarino. Comparison of very short-term load forecasting techniques. *IEEE Trans. on Power Systems*, 11(2):877–882, May 1996.

[15] Z. Lo and B. Bavarian. Improved rate of convergence in Kohonen neural network. In *Proceedings of the IJCNN*, volume 2, pages 201–206, July 8–12 1991.

[16] O. Mohammed, D. Park, R. Merchant, T. Dinh, C. Tong, A. Azeem, J. Farah, and C. Drake. Practical experiences with an adaptive neural network short-term load forecasting system. *IEEE Trans. on Power Systems*, 10(1):254–265, February 1995.

[17] H. Mori. State-of-the-art overview on artificial neural networks in power systems. In *A Tutorial Course on Artificial Neural Networks with Applications to Power Systems*, chapter 6, pages 51–70. IEEE Power Engineering Society, 1996.

[18] A. Papalexopoulos, S. Hao, and T. Peng. An implementation of a neural network based load forecasting model for the EMS. *IEEE Trans. on Power Systems*, 9(4):1956–1962, Nov 1994.

[19] D. Park, M. El-Sharkawi, R. Marks II, L. Atlas, and M. Damborg. Electric load forecasting using an artificial neural network. *IEEE Trans. on Power Systems*, 6(2):442–449, May 1991.

[20] R. S. Sutton and A. G. Barto. Toward a modern theory of adaptive networks: Expectation and prediction. *Psychological Review*, 88:135–170, 1981.

[21] G. Widrow and M. E. Hoff. Adaptive switching circuits. *Institute of Radio Engineers, Western Electronic Show and Convention, Part 4*, 1960.