2006 International Joint Conference on Neural Networks
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

System-Type Neural Network Architectures for Power Systems

Kwang Y. Lee, *Fellow, IEEE*

***Abstract*—Neural networks have been applied in various new ways to the manifold problems in power systems. The great majority of neural network designs attempt to model a dynamic mapping with one neural network. Recently, attempts have been made at using system-type neural networks for distributed parameter systems, where the system dynamics is distributed over a spatial-temporal domain. In this paper, system-type neural networks is illustrated, which are designed using semigroup theory. The objective will be either to achieve extrapolation of functional patterns along one axis, or to achieve a forecasting of functional patterns in multiple axes.**

I. Introduction

NOWADAYS, a shift has occurred in the overall architecture of neural networks from simple or component-type networks to system-type architectures for distributed parameter systems. There are three reasons for the shift in emphasis from component-type to system-type neural networks. First, in the interests of advancing science, system-type neural networks are seen as the next step. Second, very concrete flaws have been seen in component-type neural networks [1]. Third, there is a need for learning data in multidimensional space or spatial-temporal domain. In such cases, it is necessary to use a system-type neural network where one or more components learn individual functions, and another component synthesizes their contributions. Other classes of neural networks have their own problems, which also motivate a move to system-type neural networks.

The first attempts at system-type neural networks used a variety of ad-hoc approaches based mainly on intuition. Very recently, efforts have been made to develop a disciplined approach in this area. The most popular architecture seems to be the one advocated by Jacobs and Jordan [2], called the "Modular Connectionist Architecture", which is shown in Fig. 1. It consists of a collection of expert components, each being trained independently, tied together by a component called the "gating logic" element, whose function is to decide on the relative contributions to be made by each expert component, such that when they are added, they provide the correct output for a given input. The most serious flaw in system-type neural networks is the lack of a cohesive discipline in the architectural design and in the design of the learning algorithm. Virtually, the entire design is done on an intuitive basis. As a contrast to intuition, the proposed method relies on semigroup theory. To illustrate the lack of a cohesive

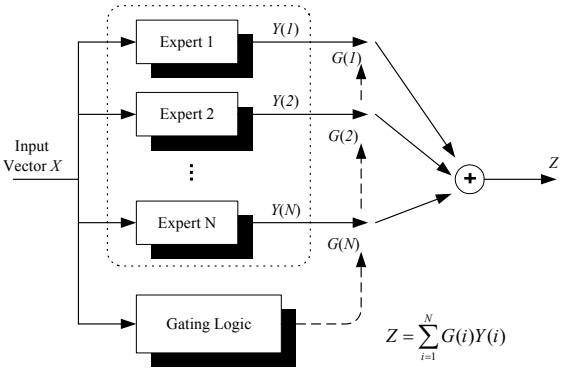

Fig. 1. Modular connectionist architecture.

discipline, in [3], the partitioning of components corresponds to separation of variables, which works if the variables are separated and does not work if the variables are not separated. As another example, in [4], after experiencing failures with one neural network emulating a complex chemical system, the authors use a multi-component architecture in which switching logic simply selects which particular component's output to use. As still another example, in [5], the author explicitly raises the issue of whether or not it is possible to develop a coherent basis for the architectural design of a multi-component neural network system, and then proceeds to improvise a multi-component solution for modeling the dry and viscous torque within a clutch by mapping components to subsets of the inputs, where the segregation process is based on first principles. In both of these cases, the design is based primarily on the designer's intuition.

In recent years, among many other applications, semigroup theory has been widely used in the study of control and stability of distributed parameter systems, or systems governed by differential equations on an abstract Banach space. It is well known that differential equations form a major tool in the study of pure and applied sciences including engineering and many areas of social sciences. Depending on the problem, these equations may take various forms, such as functional differential equations, partial differential equations (PDE's), and sometimes combination of interacting systems of ordinary and partial differential equations. In general, under broad assumptions, many of these equations can be reformulated as ordinary differential equations on abstract spaces, for example, Banach spaces [4]. This is where semigroup theory plays an important role and provides a unified and powerful tool for the study of existence, uniqueness, and continuous dependence of solutions on parameters and their regularity properties. Semigroup theory has also found extensive applications in the study of Markov process, ergodic theory, approximation theory and control and stability theory [6].

This work was supported in part by the U.S. National Science Foundation under Grant ECS-0501305.

K. Y. Lee is with the Electrical Engineering Department, the Pennsylvania State University, University Park, PA 16802 USA (phone: 814-865-2621; e-mail: kwanglee@ psu.edu).

0-7803-9490-9/06/$20.00/©2006 IEEE

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

In Section II, we describe the system-type neural networks. In Section III, we consider applications to the following areas: temperature distribution in the boiler furnace, enthalpy extrapolation in the delivery section of a power plant, prediction of mass unbalance at different speeds for a rotating machine, and short-term load forecasting. The first three applications involve extrapolation and the fourth application involves a forecasting. In Section IV, the simulation results are shown. Finally, we make conclusions in Section V.

II. System Type Neural Network

A. Proposed System Type Neural Network Architecture

Neural networks are being used for systems described by PDE's [7]. The system-type attribute of the neural network architecture is shown in Fig. 2, implementing an arbitrary function $T(z,r)$. Unlike conventional neural network architectures that would attempt to achieve the mapping $T(z,r)$ with one neural network, the proposed architecture reflects a system-type approach using two neural network channels, a Function Channel and a Semigroup Channel, in an adaptation of the connectionist architecture, which implements a mapping of the following form:

$$T(r,z) = T_z(r) = C(z)^T E(r) \tag{1}$$

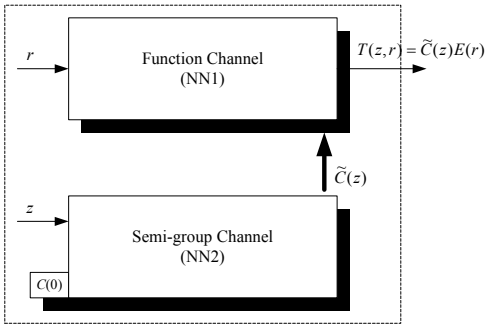

Fig. 2. System-type architecture.

During use, the semigroup channel supplies the function channel with a coefficient vector $C(z)$ as a function of the index z. The coefficient vector, when applied to the basis set $E(r)$ of the function channel, causes the function channel to operate as one specific function from within a vector space of functions. Jointly, these two channels realize a semigroup-based implementation of the mapping $T(z,r)$. The similarity between the proposed architecture (Fig. 2) and that of Fig. 1 arises from the fact that the Function channel is implemented as N "expert" systems.

The function channel can have a Radial Basis Function (RBF) architecture [8]. It consists of n RBF networks, each one of which implements one orthonormal vector of an n-dimensional basis set of vectors $E(r)$. The outputs of the orthonormal vectors are (internally) linearly summed so that the channel spans an n-dimensional function space. The coefficients which determine the linear sum and thereby define the specific function being implemented is supplied by the semigroup channel. Up to this point, the operation of the RBF channel parallels the idea used by Phan and Frueh [9]. One of the essential differences between their approach and the present proposed approach is that the former requires prior engineering knowledge for selecting the basis vectors, and the latter approach requires no such knowledge. One advantage that RBF networks have over other architectures is that their functionality can be given an explicit mathematical expression in which the neuron activation functions act as Green's functions [10]. Another advantage is that they function as universal approximators [8]. Still another advantage that RBF networks have is that they can be designed rather than trained.

The semigroup channel can be adapted from the Diagonal Neural Network (DRNN) [11] or the Simple Recurrent Network (SRN) architecture [12], in which the input is split into a dynamic scalar component z and one static vector component, the vector $C(0)$. The output is a vector $C(z)$, which is related to the dynamic input z and to the static input $C(0)$ by the semigroup property:

$$C(z) = \Phi(z)C(0) \tag{2}$$

where

$$\Phi(z_1 + z_2) = \Phi(z_1)\Phi(z_2) \tag{3}$$

B. Learning Algorithm of Proposed System-Type Neural Network

The first component of the system, namely the Function Channel, can be designed, rather than trained. The design consists of determining the algebraic dimensionality of $E(r)$, say n, of then choosing n sample functions from the given data, of then orthonormalizing the n functions, and finally of training n neural networks to emulate those n functions. For each z, a coefficient vector $C(z)$ is then formed which expresses the linear dependence of the z^{th} sample function on the basis set.

The second component, the Semigroup Channel, can be trained in the new way illustrated below. During training, the semigroup channel receives as input a preliminary coefficient vector $C(z)$ and produces a smoothened coefficient vector $\tilde{C}(z)$. That is, the primary objective of training is to replicate (and, if necessary, to smoothen) the vector $C(z)$ with a vector $\tilde{C}(z)$ which has the following semigroup property [13]:

$$\tilde{C}(z) = \Phi(z)\tilde{C}(0) \tag{4}$$

where $\tilde{C}(z) \equiv [\tilde{c}_1(z), \tilde{c}_2(z), ..., \tilde{c}_N(z)]^T$, $\Phi(z)$: an nxn matrix that satisfies (3).

However, there is a secondary objective of training; the channel must also "replicate" the semigroup property of the trajectory by gradually acquiring a semigroup property of its

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

own, in weight space. In terms of control theory, the idea is to get the semigroup channel to gradually behave as an observer of the reference model, which is producing the coefficient trajectory. When this observer behavior finally evolves, the output of the semigroup channel then tracks the output of the reference model. (The existence of this acquired semigroup property in weight space becomes the basis for extrapolation.) In order to elicit this gradual acquisition of the semigroup property, it is necessary that the training in this second step (semigroup tracking) occur in a graduated manner, as shown in Fig. 3, where l is the length of coefficient vector. It must be noted that there are two concepts of convergence that occur. First, to acquire a given weight, for example, weight W_3, requires conventional training convergence, which in turn may require 500 training iterations. Second, after all weights (W_1, W_2, ..., W_n) have been obtained, a search begins for a convergence within this weight stream alone.

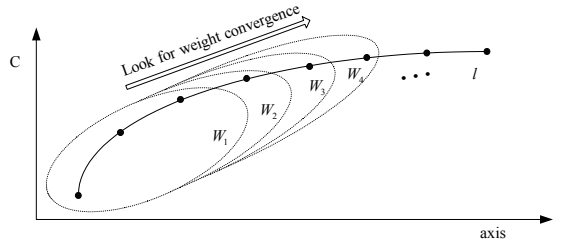

Fig. 3. Overview of new training algorithm.

C. Extrapolation

Extrapolation involves only the coefficient vector and the Simple Recurrent Network (the semigroup channel). At the uppermost level, the idea is to train the neural network to replicate the coefficient vector (produced by the previous system modeling effort) in such a way that it is additionally replicating the semigroup property, which is responsible for generating the coefficient vector by acquiring a semigroup property of its own in weight space. This idea requires a gradual training approach in which the sequence of weight changes reaches a point of weight convergence, after which only that portion of the weight changes that are connected to the extrapolating variable experience any subsequent changes. This convergence is accompanied by a linearization of the original nonlinear SRN behavior in which the rule for weight change (for the extrapolating variable) generates a semigroup property.

D. Forecasting

By replacing the semigroup channel with a comparator, a slight variation of the above functionality is achieved, where the goal of extrapolation is replaced with a new goal of creating an entire new surface based upon a relationship between parameters which is shown in Fig. 4. Considering the family of functions shown in the Fig. 4, if for general m, $F^m(x,y) = C^m(x)E(y)$, and if the basis set $E(y)$ is common to all members of family, and finally if a relationship exists between the parameters ξ_1, ξ_2, ..., and ξ_m, then a relationship

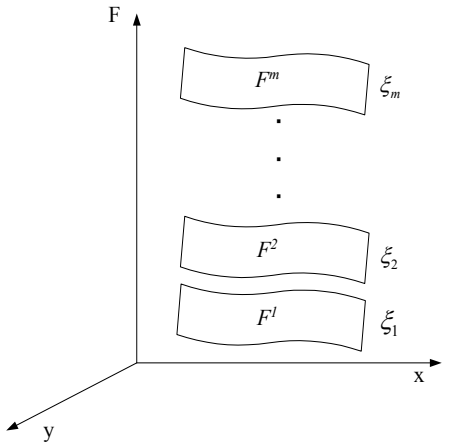

Fig. 4. Family of parameterized functions.

will exist between the coefficient vectors of the family. That is using F^1 as a reference, there is a mapping from the coefficient vector of F^1 to the coefficient vector of F^m:

$$C^m(x) = T(\xi_m)C^1(x) \tag{5}$$

III. Applications

Three applications of extrapolation and one application of load forecasting will be considered.

A. Monitoring of Temperature in Boiler Furnace

The electric utility industry is charged to deliver power as inexpensively and as reliably as possible. Meeting these dual obligations has become increasingly difficult over the past 30 years. Environmental and economic concerns pressed the utility industry to develop clean and efficient ways of burning coal and oil. This has required major improvements in instrument, data management, and control of electric power plant components such as boilers. It has become a challenge to measure high temperature distributions of high-pressure liquids, steam, combustion gases, and heat transfer components in extremely adverse power plant environments. Traditional sensors have not exhibited sufficient stability and long-term accuracy without requiring expensive maintenance

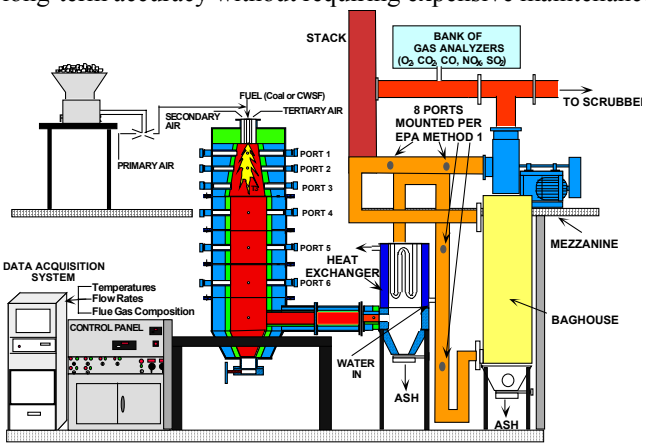

Fig. 5. Schematic diagram of the Penn State 500,000 Btu/h Down-Fired Combustor.

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

and recalibration. Additionally, each sensor only provides one reading so that only a limited number of readings are obtained.

Fig. 5 shows the Penn State down-fired combustor (DFC), which is an advanced pilot-scale furnace designed to evaluate the combustion performance of various fuels (natural gas, coal, coal-water slurry fuel) including emissions monitoring. The combustor has a 20-inch internal diameter, is 10 feet high, and is designed for a thermal input of 350,000 Btu/h (nominal), but this can be varied from 200,000 to 500,000 Btu/h. The proposed boiler furnace-monitoring model addresses the estimation of spatial temperature distribution continuously for any operating condition.

As an alternative to the above model-based estimation techniques, such as infinite dimensional extended Kalman filtering, an intelligent monitoring scheme will be developed for 3D temperature estimation by using the proposed system-type neural networks. An intelligent algorithm will be developed to adaptively tune the monitoring system in real-time to implement in the experimental boilers. The previous emphasis on the application of computational intelligence for control and diagnostic will be shifted to state estimation and prediction problems.

B. Enthalpy

The electric utility industry is confronted with the task of estimating the steam enthalpy at various points in the water-steam cycle in a power plant. The two prominent estimation points are at the boiler, where water is converted into steam, and also in the delivery section which precedes the turbine where the energy of the steam is extracted and converted into mechanical power. It is very important to have functions which are able to accurately describe the correlations between enthalpy and temperature for the water/steam because the enthalpy provides the best description of the energy content for a compressible gas. In the literature, there are numerous works which present mathematical functions of enthalpy vs. temperature, but in many cases they provide insufficient approximation with experimental data or have a good approximation only over a small temperature range [14]. In addition, these methods have no provision for extending (extrapolating) accurate readings into a higher temperature range in which the readings become questionable. In this paper, a new method of extrapolating the enthalpy is proposed using a system-type neural network architecture. Essentially, rather than relying on questionable temperature readings to calculate the enthalpy, this method extrapolates a set of reliable enthalpy readings directly.

Considering the general power plant as shown in Fig. 6, and referring to the delivery section from points 2 to 3, which precedes the turbine in a power plant, there is a need for tracking the steam enthalpy, since it is this function which ultimately determines the mechanical power delivered by the turbine. From the conservation of energy principle, the turbine work per mass of airflow is equal to the change in the enthalpy of the flow from the entrance to the exit of the turbine. Therefore, if we can measure the enthalpy among the delivery section and turbine, we can determine the mechanical power which then becomes the electrical power. The difficulty is that the enthalpy is derived from the temperature and, in the usual cases, these involve very high temperatures, and accurate readings of high temperature steam in the presence of high pressures are very difficult to achieve. At present, various temperature compensation schemes are employed but, even with these, the resulting temperature readings are questionable. Therefore, the resulting enthalpy estimations are questionable [15]. The proposed method suggests an alternative, namely, to obtain a small (sparse) set of reliable temperature-pressure readings at the front end of the delivery section, forming the enthalpy from those readings, and then extrapolating those enthalpy readings directly.

C. Mass Unbalance

Avoiding destructive vibration is of major importance in the turbine-generator of a power plant. Mass unbalance is the most common source of vibration in machines with rotating parts. Balancing of rotors prevents excessive loading of bearings and avoids fatigue failure, thus increasing the useful life of machinery. There are many studies on the vibration subject and most of those studies are based on linear modeling. It was found, however, that linear rotor dynamics cannot account for an unbalance as it had occurred. In addition, a linear model is not sufficient at high frequencies [16]. During transient loads, furthermore, extreme conditions have been observed and efficient methods and tools to analyze such cases are of primary interest to the industry. Vibration has been notoriously difficult to measure, and most reported measurements have used accelerometers attached to a stationary housing carried by a retrofitted bearing mounted on the shaft of interest.

To illustrate the proposed procedure, the most elementary physical model will be assumed. The physical system consists of a shaft carrying a mass at its midspan and having a small mass unbalance, as shown in Fig. 7. In general, the resulting vibrations can be complex, depending primarily on the

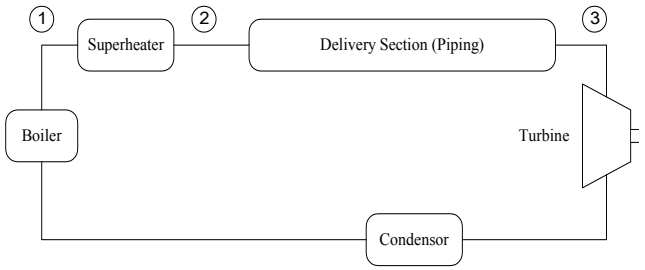

Fig. 6. General power plant.

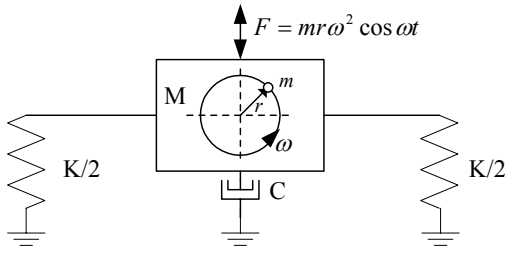

Fig. 7. Simple mass unbalance system.

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

geometry of the system. In this case only the simplest forms of transverse vibrations along the longitudinal axis of the shaft are being considered. In general, it takes a long time to start-up the turbine in a power system. That is, the start-up proceeds in a series of steps, along each of which the speed is held constant. The total start-up time may take up to 18 hours. Therefore, if at any point during the start-up, we can anticipate the mass unbalance at a future step, we can balance it to avoid vibration at higher speeds.

For simulation purposes, the simple mass unbalance system will be considered to be one integral mass, M, which represents the rotor mass, along with a small mass unbalance offset (*mr*) mounted on a stubby shaft and supported at the two ends by its own bearings which have nonlinearity, as shown in the Fig.7. The damping (C) is assumed linear.

D. Load Forecasting

The proposed method uses artificial neural networks with three significant features. One, the influence of the day of the week and of the season of the year are combined into one variable. Further, the recognized influence of weekdays versus weekends must be expanded to a day-by-day influence. Therefore, the approach begins by introducing a new variable for each day of the week and each week of the year. For example, Mondays are grouped into one variable, number of Mondays, which ranges from 1 to 52. The load for a Monday is then represented as a function of the hour of the day and Monday number. The second significant feature of the approach is that this load must be able to be represented as the product of a coefficient vector and a basis set. This, in its general form, has been established [17]-[19]. Therefore, one of the assumptions in this paper is that the general load can be represented in the following form.

$$L(Day, Hour) = C(Day)E(Hour) \qquad (6)$$

The third significant feature of the approach is that the coe fficient vector for the forecasting year is related to the coeffi cient vector from the reference year.

$$C^{forecast}(Day) = T(\alpha, \beta)C^{reference}(Day) \qquad (7)$$

IV. Simulation Results

A. Monitoring of Temperature in Boiler Furnace

The following illustrates simulation results of the application of the proposed method to the prediction (extrapolation) of temperature data from a boiler furnace of dimensions comparable to that found in a power plant. The data represents "raw data" furnished by the Penn State Energy Institute. The geometry of the furnace is cylindrical with the z-axis along the furnace axis, and with r going from one wall to the other wall. (Note that r is a diameter, not a radius.) A simulation will be performed on the configuration below, where there are 25 probes, each one providing 11 readings as shown in Fig. 8. The extrapolation will be simulated in the region occupied by probes 25 to 30. The results of the

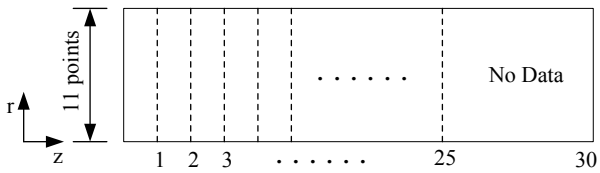

Fig. 8. Temperature probe configuration for the furnace.

extrapolation will be compared to given raw data in that region. The empirical and computed temperature distributions are shown in Fig. 9 and Fig. 10. The preliminary (rough) coefficient vector and the basis vectors produced by the RBF network are shown in Fig. 11. The use of this rough

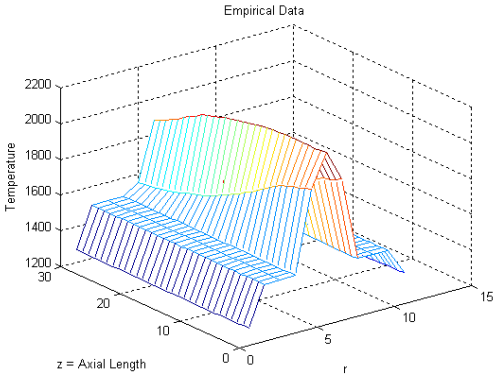

Fig. 9. Temperature distribution for the furnace.

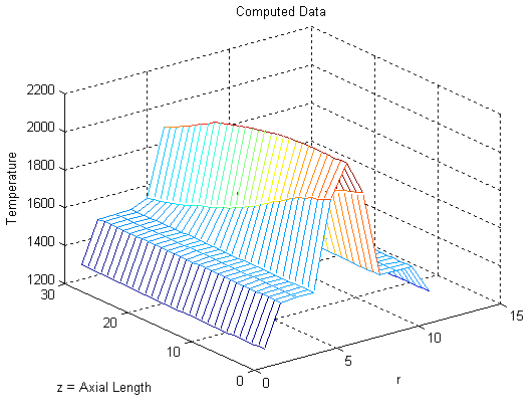

Fig. 10. Computed temperature.

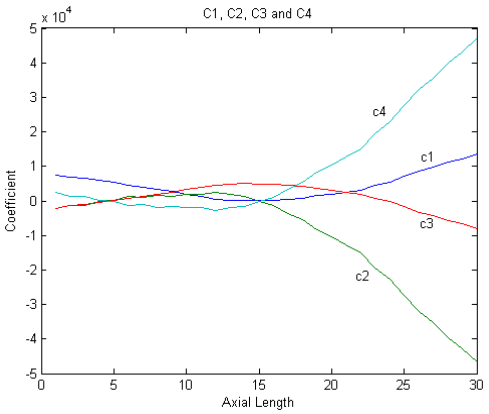

Fig. 11. Preliminary coefficient vector set.

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

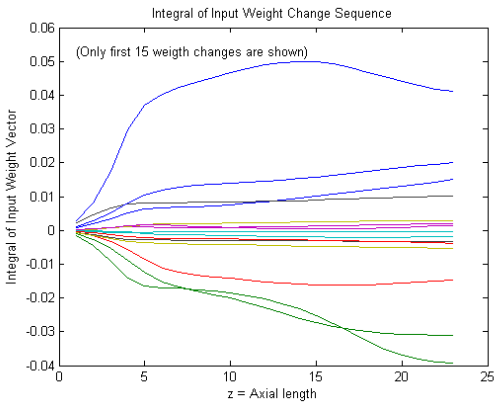

Fig. 12. Integral of input weight change sequence

coefficient vector together with the basis set of vectors can produce the computed temperature distribution shown in Fig. 10. The possibility for extrapolation begins by checking for weight convergence as training is performed along the coefficient vector. In this case, weight convergence occurs as this training is repeated over successively longer intervals (refer Fig. 3). It is this weight convergence, which becomes the basis for extrapolation. These are shown in Fig. 12. In this case, because of the smoothness, the possibility for extrapolation exists and the next step is to apply an extrapolation test in which the trailing end of the weight change sequence (produced by training) is replaced by an

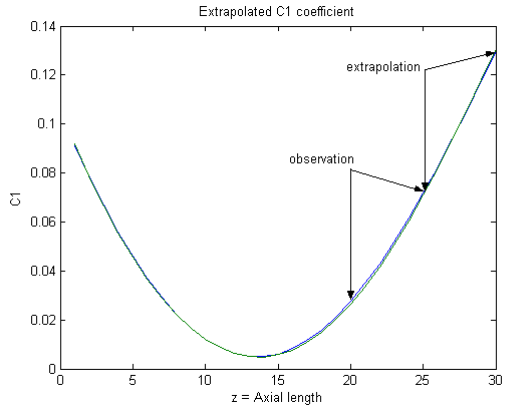

Fig. 13a. Extrapolation results for C1.

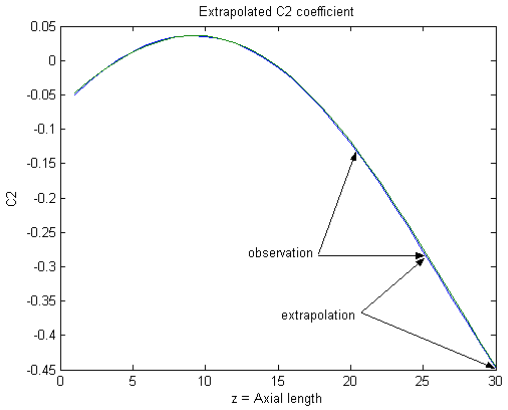

Fig. 13b. Extrapolation results for C2

equivalent weight change sequence based on a rule that generates a semigroup. Based upon an observation of the weight change sequence on the interval from 15 to 20, a semigroup-based rule for weight change is formulated and applied to the interval from 20 to 25, as a test. Extrapolation (to the region where no data were assumed) consists of the autonomous continuation of the rule for weight change, which was derived during the extrapolation test. These results are shown in Fig. 13a and 13b below (only the first two coefficients are shown).

B. Enthalpy

The steam enthalpy corresponding to the temperature and pressure is obtained from the NIST Chemistry WebBook [20]. The range of temperature is 800°F to 1200°F and the range of

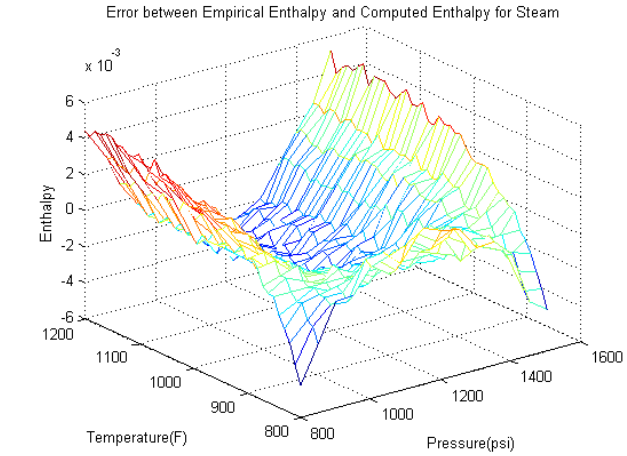

Fig. 14. Error between empirical and computed enthalpies.

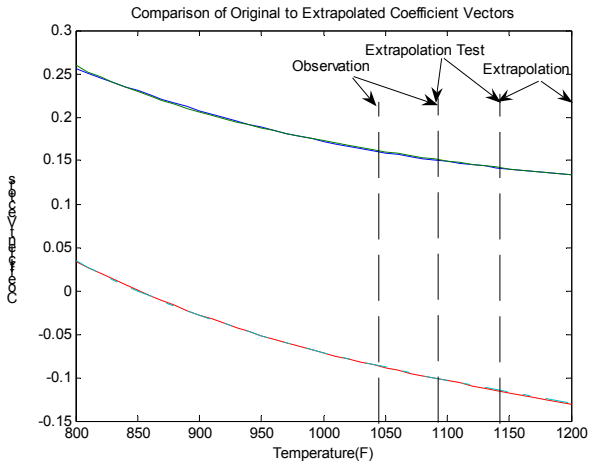

Fig. 15. Extrapolated coefficient vectors.

pressure is 800 psi to 1500 psi. The proposed method will be applied to the extrapolation of the enthalpy of steam for temperature-pressure distributions which typically exist in the power plant after the water has exited the boiler and the superheater and travels through the delivery section to the turbine (dry stream – points 2 to 3 in Fig. 6). The steam enthalpy is first re-expressed as the vector product: $h(T,P) = C(T)E(P)$ and extrapolation is performed along

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

the temperature axis. Fig. 14 displays the error between the given empirical enthalpy and the computed enthalpy. Fig. 15 displays the extrapolated coefficient vector. Based upon an observation of the weight change sequence on the interval from 1040°F to 1090°F, a semigroup based rule for weight change is formulated and applied to the interval from 1090°F to 1140°F, as a test. Extrapolation consists of the autonomous continuation of the rule for weight change, which was derived during the extrapolation test. Extrapolation is performed from 1140°F to 1200°F where no data were assumed

C. Mass Unbalance

In its simplest form, the nonlinear mass unbalance problem can be described by the following classic Duffing equation:

$$\ddot{x} + \gamma\dot{x} + (\alpha x + \beta x^3) = F\omega^2 cos\omega t \tag{8}$$

where ω = engine speed, $F = mr$ = normalized mass unbalance. It can be shown that Eq. (8) has the following approximate solution [21].

$$F^2(\omega, A) = \frac{\left[\left(\alpha - \omega^2\right)A + \frac{3}{4}\beta A^3\right]^2 + \left(\gamma A\omega\right)^2}{\omega^4} \tag{9}$$

where, ω = engine speed, α, β, γ = constant, and A = vibration magnitude. This will be simulated on the domain: $\omega \in [100,150]\times 2\pi$; $A \in [0.01, 0.10]$, using the following parameter values: $\alpha = 900$; $\beta = 0.75$; $\gamma = 0.75$ [22].

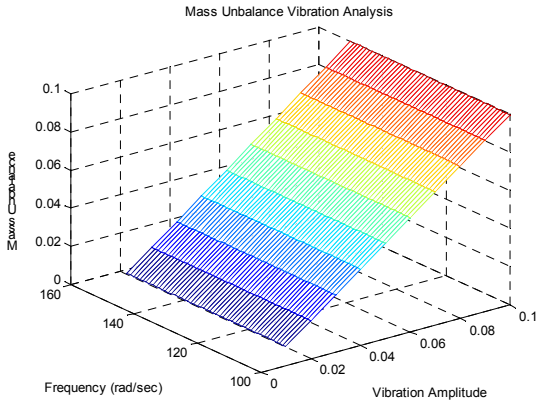

Fig. 16. Mass unbalance profile.

The mass unbalance raw data profile is shown in Fig. 16. The error between empirical and computed unbalance profile is shown in Fig. 17. Based upon an observation of the weight change sequence on the interval from 125 to 130 rad/sec, a semigroup-based rule for weight change is formulated and applied to the interval from 130 to 135 rad/sec, as a test. Extrapolation consists of the autonomous continuation of the rule for weight change, which was derived during the extrapolation test. These results are shown in Fig. 18.

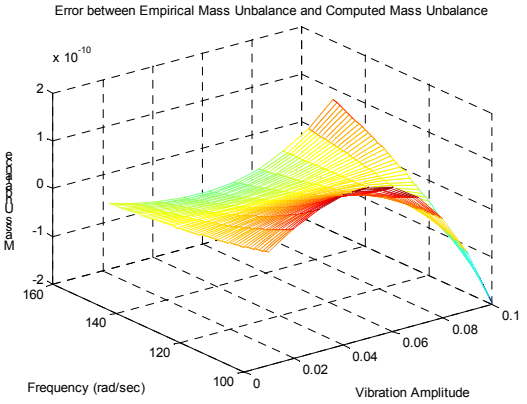

Fig. 17. Error between empirical and computed unbalance profile.

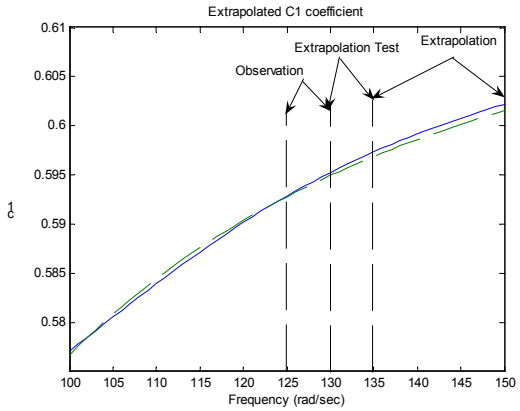

Fig. 18. Extrapolation.

D. Load Forecasting

The proposed forecasting procedure was tested using the past load data obtained from Korea Power Exchange (KPX). For the simulation, 2004 load data was chosen as a forecast year. Also the reference year was arbitrarily chosen as the previous year (2003). The simulation results are shown in Table I. For each day, the percent error is obtained as the average of 52 days for each day of the week (over the entire year). Also, the unit of standard deviation is MW. Among the daily forecasting, Sunday shows the best forecasting results and Thursday shows the worst forecasting results. For Sunday, the average error is 0.89% and the highest error is 1.35% and the lowest error is 0.5%. For Thursday, the average error is 1.79% and the highest error is 2.72% and the lowest error is 1.07%. The total average error for daily forecasting is 1.35% and average standard deviation is 657.6 MW. Monday, Wednesday, Thursday, and Friday errors are above the total average with respective errors 1.61%, 1.54%, 1.79%, and 1.49% and Tuesday, Saturday, and Sunday errors are below the total average with respective errors 1.01%, 1.14%, and 0.89%.

V. Conclusions

In this paper, we investigate a mathematical approach to extrapolation of distributed parameter systems with various examples in power systems, such as boiler facility, steam enthalpy, and mass unbalance problem using a combination

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.

of a modified neural network architecture and semigroup theory. Given a set of empirical data in spatial-temporal domain with no analytic expression, we first develop an analytic description with available data and then extend that model along a single axis for extrapolation. From the results, we conclude that the proposed system-type neural network architecture works well for the examples presented for both extrapolation and load forecasting. The concept of the proposed system-type neural network architecture and training method can be applied to other engineering and non-engineering problems.

REFERENCES

[1] M. Omid, L. Omidvar, and C. Wilson, "Progress in neural networks," in *Architecture*, vol. 5, Ablex Publishing Co, Ed, N.J., 1997.

[2] R. Jacobs and M. Jordan, "A competitive modular connectionist architecture," in *Proc. Advances in Neural Information Processing Systems*, vol. 3, pp. 767-773, 1991.

[3] A. Atiya, R. Aiyad, and S. Shaheen, "A practical gated expert system neural network," *IEEE International Joint Conference on Neural Networks*, vol. 1, pp. 419-424, 1998.

[4] J. P. Velas, "A neural network algorithm for system modeling, global extrapolation, and parameter estimation for acoustical data," Ph.D. dissertation, Department of Acoustics, the Pennsylvania State University, University Park, PA, 2003.

[5] M. Cao, "Grey box neural network and its application to system modeling," M.S. Paper, Department of Electrical Engineering, the Pennsylvania State University, University Park, PA, 2001.

[6] N. U. Ahmed, *Semigroup Theory with Applications to Systems and Control*. Longman Scientific & Technical, Harlow, 1991.

[7] R. Padhi and S. N. Balakrishnan, "Proper orthogonal decomposition based feedback optimal control synthesis of distributed parameter systems using neural networks," *Proceedings of the 2002 American Control Conference*, vol. 6, pp. 4389 – 4394, May, 2002.

[8] S. Haykin, *Neural Networks*. 2nd ed., Prentice Hall, N.J., 1999.

[9] M. Q. Phan and J. A. Frueh, "Learning control for trajectory tracking using basis functions," *Proceedings of the 35th IEEE Conference on Decision and Control*, pp. 2490 - 2492, Dec. 1996.

[10] A.N. Tikhonov, "On solving incorrectly posed problems and method of regularization," *Doklady Akademii Nauk* vol. 151., USSR, 1973.

[11] C. C. Ku and K.Y. Lee, "Diagonal recurrent neural networks for dynamic systems control," *IEEE Trans. on Neural Networks,* vol. 6, pp. 144-156, Jan. 1995.

[12] J. Elman., "Finding structure in time," *Journal of Cognitive Science*, vol. 14, pp. 179-211, 1990.

[13] I. Miyadera, *Nonlinear Semigroups*. American Mathematical Society, Providence, R.I., 1992.

[14] R Lanzafame and M. Messina, "A new method for the calculation of gases enthalpy," *Energy Conversion Engineering Conference and Exhibit*, vol.1, pp. 318~328, July 2000.

[15] M. A. Tahani and C. Lucas, "Development of expert controller for steam temperature regulation in power plants," *IEEE/RSJ International Workshop on Intelligence for Mechanical Systems*, vol.3, pp. 1333 – 1337, Nov. 1991.

[16] S. F. Stefan, L. Noel, and R. Thomas, "Mechanical behavior of an industrial gas turbine under fault conditions, a case history," *Proceedings of ISROMAC-7 Conference*, vol. **A**, pp. 373-382, 1998.

[17] K. Y. Lee, J. P. Velas, and B. H. Kim, "Development of an intelligent monitoring system with high temperature distributed fiberoptic sensor for fossil-fuel power plants," *IEEE Power Engineering Society General Meeting*, pp. 1350-1355, June 2004.

[18] B. H. Kim, J. P. Velas, and K. Y. Lee, "Development of intelligent monitoring system for fossil-fuel power plants using system-type neural networks and semigroup theory," *IEEE Power Engineering Society General Meeting*, pp.2949-2954, 2005.

[19] B. H. Kim, J. P. Velas, and K. Y. Lee, "Semigroup based neural network architecture for extrapolation of enthalpy in a power plant," *Proceedings of the ISAP*, pp. 291-296, 2005.

[20] *NIST Chemistry WebBook*. Available: http://webbook.nist.gov/chemistry/fluid/

[21] J. J. Stoker, *Nonlinear Vibrations*. Interscience, N.Y., 1950.

[22] J. J. Vance, *Rotordynamics of Turbomachinery*. Wiley, N.Y., 1988.

TABLE I
STATISTICS OF DAILY FORECASTING RESULTS

	Mon.		Tue.		Wed.		Thr.		Fri.		Sat.		Sun.	
Hour	Err.	Std. Dev.	Err.	Std. Dev.	Err.	Std. Dev.	Err.	Std. Dev.	Err.	Std. Dev.	Err.	Std. Dev.	Err.	Std. Dev.
1	2.1	901	1.37	654	1.39	691	1.57	752	1.49	712	1.42	717	1.35	630
2	2.26	887	1.13	503	1.65	772	1.88	900	1.6	721	1.34	599	1.05	466
3	2.58	979	1.17	515	1.87	838	2.18	1042	1.84	794	1.46	603	1	414
4	2.37	996	0.92	433	2.02	871	2.36	1107	1.81	757	1.2	522	0.7	288
5	2.52	1055	0.93	446	2.27	982	2.61	1212	1.73	797	1.09	481	0.52	233
6	2.51	1083	1.17	494	2.4	1063	2.72	1289	2	885	1.39	583	0.96	375
7	2.32	998	1.24	536	2.37	1091	2.62	1190	1.96	901	1.55	630	1.15	446
8	1.87	809	0.89	420	1.62	789	1.89	829	1.51	687	1.38	580	0.93	362
9	1.16	608	0.85	415	1.17	674	1.21	727	1.05	479	1.13	604	0.87	333
10	1.44	799	0.91	505	1.44	769	1.32	856	1.28	631	1.05	607	0.8	318
11	1.47	785	0.96	617	1.72	839	1.82	915	1.62	823	0.95	509	0.71	283
12	1.44	803	1.05	705	1.79	916	1.82	913	1.53	786	0.86	445	0.59	258
13	1.48	684	1.4	672	1.38	684	1.72	808	1.51	708	1.22	552	0.76	343
14	1.13	561	0.96	475	1.04	549	1.23	643	0.68	362	0.83	396	0.83	338
15	0.85	479	0.65	345	1.11	595	1.07	617	1.3	638	0.68	373	0.84	357
16	0.84	443	0.68	356	1.11	551	1.23	676	1.33	656	0.67	347	0.65	277
17	1.07	536	0.9	519	0.99	507	1.27	739	1.2	599	0.9	436	0.76	303
18	1.14	575	1.08	542	0.93	487	2.43	1254	1.57	810	1.23	626	0.5	203
19	1.36	765	1.03	552	1.8	890	1.54	826	1.03	541	1.33	637	1.05	442
20	1.1	592	0.83	500	0.79	436	1.5	804	1.65	866	0.8	400	0.76	364
21	1.26	667	0.69	363	1.55	864	1.78	958	1.5	813	1.22	650	0.79	395
22	1.42	717	0.87	488	1.54	821	1.8	1025	1.51	756	0.86	439	0.74	371
23	1.72	876	1.53	787	1.98	1055	1.95	1041	1.83	946	1.59	769	1.7	754
24	1.29	672	1.12	624	1.11	603	1.34	707	1.22	642	1.13	598	1.33	629
Avg	1.61	761	1.01	519	1.54	764	1.79	909	1.49	721	1.14	546	0.89	383

Authorized licensed use limited to: Baylor University. Downloaded on January 26, 2010 at 20:21 from IEEE Xplore. Restrictions apply.