APPLICATION OF ARTIFICIAL NEURAL NETWORK MODELING TO THE ANALYSIS OF THE AUTOMATED RADIOXENON SAMPLER-ANALYZER STATE OF HEALTH SENSORS

James C. Hayes, Pamela G. Doctor, Tom R. Heimbigner, Charles W. Hubbard, Lars J. Kangas, Paul E. Keller, Justin I. McIntyre, Brian T. Schrom, and Reynold Suarez

Pacific Northwest National Laboratory

Sponsored by National Nuclear Security Administration
Office of Nonproliferation Research and Development
Office of Defense Nuclear Nonproliferation

Contract No. DE-AC05-RLO1830

ABSTRACT

The Automated Radioxenon Analyzer/Sampler (ARSA) is a complex gas collection and analysis system operating autonomously under computer control. The instruments are part of a network of sensors, some located in remote areas that feed concentration data to a central data center. Because the ARSA instrument is complex, it requires constant monitoring to verify that it is operating according to specifications. System performance monitoring is accomplished by over 200 internal sensors, some of which also send their readings to the data center. Several of those sensors are designated as safety sensors that can automatically shut down the ARSA when unsafe conditions arise, in which case the data center is advised of the shutdown and the cause, so that repairs may be initiated. However, the other sensors, called state of health (SOH) sensors, provide valuable information on the functioning of the ARSA and it would be desirable to detect impending malfunctions before they occur to avoid unscheduled shutdowns. Any of the sensor readings can be displayed by the ARSA Data Viewer, but the interpretation of the data requires specialized technical knowledge that is not routinely available at the data center. Therefore, it would be advantageous to have the sensors automatically monitored and evaluated for the precursors of malfunctions and the results transmitted to the data center. With more automated analysis, the operation of the data center would not require as much detailed technical expertise. Artificial Neural Networks (ANN) are data analysis methods that have shown wide application to monitoring systems with large numbers of information inputs, such as the ARSA. Structured and unstructured ANN methods were applied to ARSA SOH data recording during normal operation of the instrument.

OBJECTIVES

The objectives of the study are first to select and evaluate data analysis methods that are capable of automated real-time monitoring and analysis of the output of the SOH sensors to provide early warning of a potential malfunction before the system is forced to shut down. Second, benefits of this research would mean the ARSA could be operated by less technically trained personnel. The third objective was to develop an automated method to identify the fewest number of sensors that contain the most information for determining the state of health of the ARSA instrument.

The ARSA is a complex gas collection and analysis system operating autonomously under computer control (Hayes et al., 1999; Heimbigner et al., 2002). There are in excess of 80 digital outputs used to control valves, apply power to various pumps and compressors, reset various components, and shuttle calibration sources in and out of the system's nuclear detector. The 10 analog output signals are used to adjust three mass flow controllers as well as control the system's seven heaters.

The ARSA requires constant online monitoring of system operations and overall system health (Heimbigner et al., 2004**).** The software control system records and monitors over 200 different system sensors (temperature, pressures, voltages, etc.). Approximately 20 digital input signals are used to monitor source transfer and safety related sensors, such as heater over-temperature signals. A real-time record of the system state allows the system to monitor for unsafe conditions and maintain a safe state regardless of external or internal failures (e.g., vacuum pump, valve, or power failures and runaway temperatures). If the readings from safety-related sensors deviate from safe levels, the system shuts down in an orderly fashion, and a message is sent to the data center describing the cause of the shutdown, which initiates repair activities.

Figure 1 shows the capability of the ARSA system data viewer to display the values of multiple sensors simultaneously. When viewing SOH data, the user can select any combination of analog and digital sensors for simultaneous display. This capability allows a user who is familiar with the operations of the ARSA to evaluate the health of the system. With the inherent zoom capability, the user can examine specific regions of the data in more detail. Another function of real-time monitoring allows the user to troubleshoot the system when a problem arises should a minor sensor or a major system failure occur.

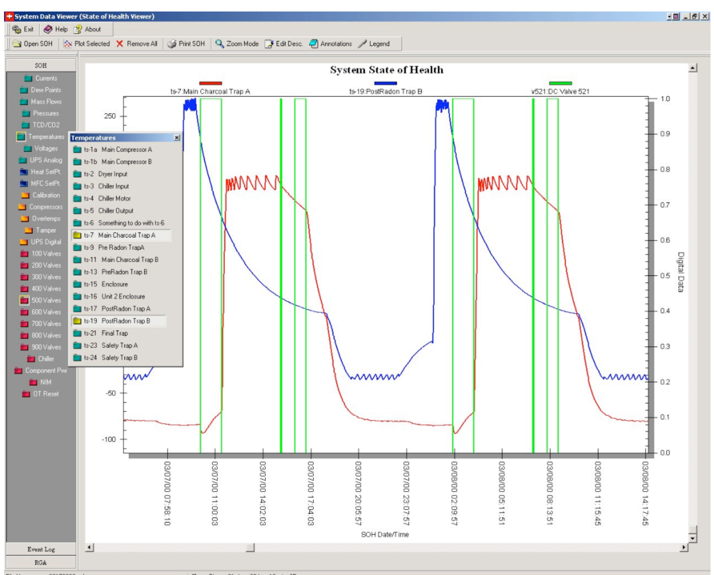

Figure 1. Three of the 46 ARSA analog SOH traces as seen in the ARSA data viewer.

In Figure 1, the behavior of the three sensors represents normal operations. The figure shows that sensor behavior repeats from one cycle to the next with some sensors more consistent than others. These cycles vary at 8-, 16-, and 32-hour intervals for different subsystems (and sensor behavior). Due to the complexity of the plots, as in this example, only three sensor outputs can be represented clearly, so it is difficult to monitor a large number of sensors simultaneously. It would be advantageous to have software algorithms that could continuously monitor the sensors simultaneously to identify subtle changes in multiple sensors that can be indicative of impending malfunctions but are difficult to see by viewing the behavior of one or a few sensors. When detected, the system could send a message to the data center describing the impending malfunction and request service before the system is shut down. The added advantage of an automated SOH monitoring system is that there would be less need to rely on trained personnel to monitor the ARSA operations.

In addition, the data transmission infrastructure restricts the throughput of SOH sensors readings to the data center on a continuous basis. Currently, the data transmission limit is six SOH sensors that can be sent to the data center. Therefore, it would be desirable to have an analytical method to determine which of the sensors provides the most information for predicting failures and transmit only those data to the data center.

RESEARCH ACCOMPLISHED

In this study, we used the feed-forward ANN as an Auto-Associative (AA) ANN to model the relationship between the ARSA sensor values by reducing the values to principal components (PCs) and then expanded the same components to predict sensor values. ANNs are a class of analytical techniques that can potentially address our objectives and will be the focus of this report. The ability of the ANN methodology to address the three objectives was evaluated by using several ANN modeling approaches. The data used in the evaluation of ANN methodologies consisted of sensor readings taken every two minutes from 51 SOH sensors over a three-week period in 2001. Each two-minute reading had a date/time stamp. During this period the ARSA instrument was considered to be operating normally.

Supervised Artificial Neural Networks

ANN modeling is an equation-free, data-driven modeling technique that tries to emulate the learning process in the human brain by using many examples. It is an information processing paradigm that is different from conventional computer programs in that it learns by example rather than by following instructions. It is thus a data-driven methodology that learns to model a system from historical data.

An ANN consists of simple mathematical “neurons” connected by weights. Figure 2 shows the structure of a simple ANN, in this case a feed-forward ANN composed of a set of highly interconnected neurons or nodes, and these neurons work in parallel to solve a complex problem. One input layer, zero or more hidden layers, and one output layer make up the architecture. Each layer contains nodes (circles in the figure), and the nodes in each layer are fully or partially connected to the nearest layers above or below by the lines. A weight is associated with each connecting line. The nodes in the input layer receive the input vectors, and nodes in the output layer produce the output vectors in response to the input vectors. The layers are connected through the weighted connections. Nodes in hidden layers and the output layer perform two calculations: they sum the products of connection weights and the signals from the previous layer, and pass that sum through a transfer function—often a sigmoid function.

A supervised ANN model is trained from known, labeled examples. The network learns a mapping from one vector space to another and is often employed in classification or regression (prediction) problems. Examples of inputs and outputs are presented to the ANN, and the values of the weights between the “neurons” in the hidden layer in the figure are estimated during the training of the ANN by “back-propagation” of the errors between the ANN outputs and the known data output (Werbos, 1974, 1994; Rumelhart et al. 1986). Once the ANN is trained and verified by presenting examples not used in the training but used for validation, it is used to predict the model outputs from the new input examples presented to it.

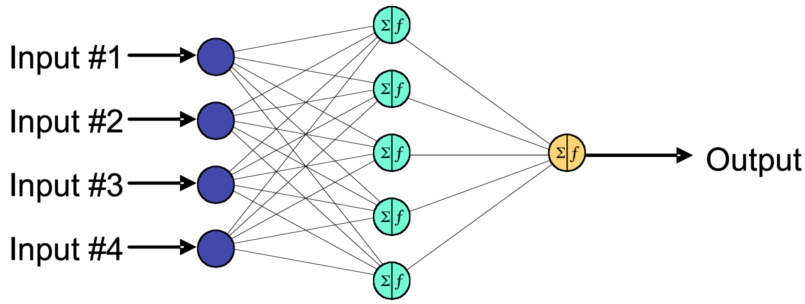

Figure 2. Diagram of a simple feed-forward ANN with a single hidden layer.

Auto-Associative Artificial Neural Networks

AA-ANNs are a special form of ANN models. They are trained to reproduce the input data as an identical vector at the output. To efficiently perform this task, they learn the interrelationships among the input variables. When trained successfully, a small number of "hidden neurons" is sufficient to recreate the input values as the output of the AA-ANN model. Also, through the same mechanism, they have some capability to reproduce data from incomplete data, i.e., to "fill in" missing variables.

Principal Component Analysis Artificial Neural Networks

An AA-ANN can be used to develop an ANN that performs a data reduction similar to that of PC analysis (PCA). When the AA-ANN is used in this manner, it is often called a PCA-ANN. The data are compressed to a few dimensions, and the subsequent prediction of the original values is only successful if the ANN learns the relationship among the values. Thus, the network learns to efficiently represent the sensor values in PCs. The ANN used in the study was not enforced with certain restrictions that would produce "true" PCs, although they are similar. See Karhunen et al., 1995 and 1997 for discussions on using ANNs for PCA.

A PCA-ANN is trained as one network but should conceptually be considered two networks. Once the AA-ANN is trained to reproduce the inputs on the outputs, the ANN is broken apart. The first network compresses the data vectors into PCs, the other expands the PCs to a vector with the original number of dimensions that is a reconstruction of the vector from its PCs. The part that connects the input to the bottleneck layer forms an ANN that maps the input sensor values to a reduced representation analogous to PCs. It also can be considered an encoder network that takes many inputs and compresses them into a few encoded components. The encoded components can then be decoded with the second half of the AA-ANN. This is illustrated in Figure 3. If the PCA-ANN is separated into two networks, as it is when it is used, a single hidden layer network will become two networks with one input layer and one output layer each. A three-hidden-layer PCA-ANN will separate into two networks with one hidden layer each. This type of network (a feed-forward network) requires at least one hidden layer to perform nonlinear mapping from input to output space. Thus, the single hidden layer PCA-ANN is limited to linear PCA, but the three-hidden-layer PCA-ANN performs nonlinear PCA, which captures the nonlinear relationships between the dimensions of the data. We chose to call the common mathematical definition of PCA a "linear" PCA to separate it from the other type, often called nonlinear PCA in the literature.

The PCA-ANN has two or more possible applications for ARSA. First, it allows the data to be compressed into fewer dimensions, benefiting both transmission and storage. Second, the PCA-ANN models the relationship between the different sensors where they exist. As such it can be used to detect when the sensor relationships deviate from the modeled values in sensor validation. In general, the PCA-ANN also functions as an anomaly detector if it is evident that several sensor values differ from those modeled. This latter use can be advanced to different levels of model based reasoning for diagnostics to determine causal origins to anomalies.

Nonlinear PCA Application to ARSA Data

A nonlinear PC neural network model was applied to the ARSA data. In this approach, three hidden layers are used in the AA training so that the final PCA-ANN has an input layer, a hidden layer, and an output layer to fully represent any nonlinear relationships in the data.

Figure 4 shows root-mean-square errors (RMSEs) for different numbers of PCs for two configurations of the PCA-ANN used in this study. One had a single hidden layer corresponding to PCs; the other had three hidden layers, with the middle layer corresponding to the PCs. The figure shows that the prediction error for a set of 46 ARSA sensors decreases as the data set is reduced to an increasing number of PCs, as expected. It also shows that the three-hidden-layer PCA-ANN performs significantly better than a single-hidden-layer PCA-ANN. The better PCA-ANN reduces the 46-sensor set to the PCs in two steps and also expands the PCs back to predicted sensor values in two steps; therefore, it captures more of the relations among the sensors. The one-hidden-layer PCA-ANN uses one step each to compress and expand the data vectors.

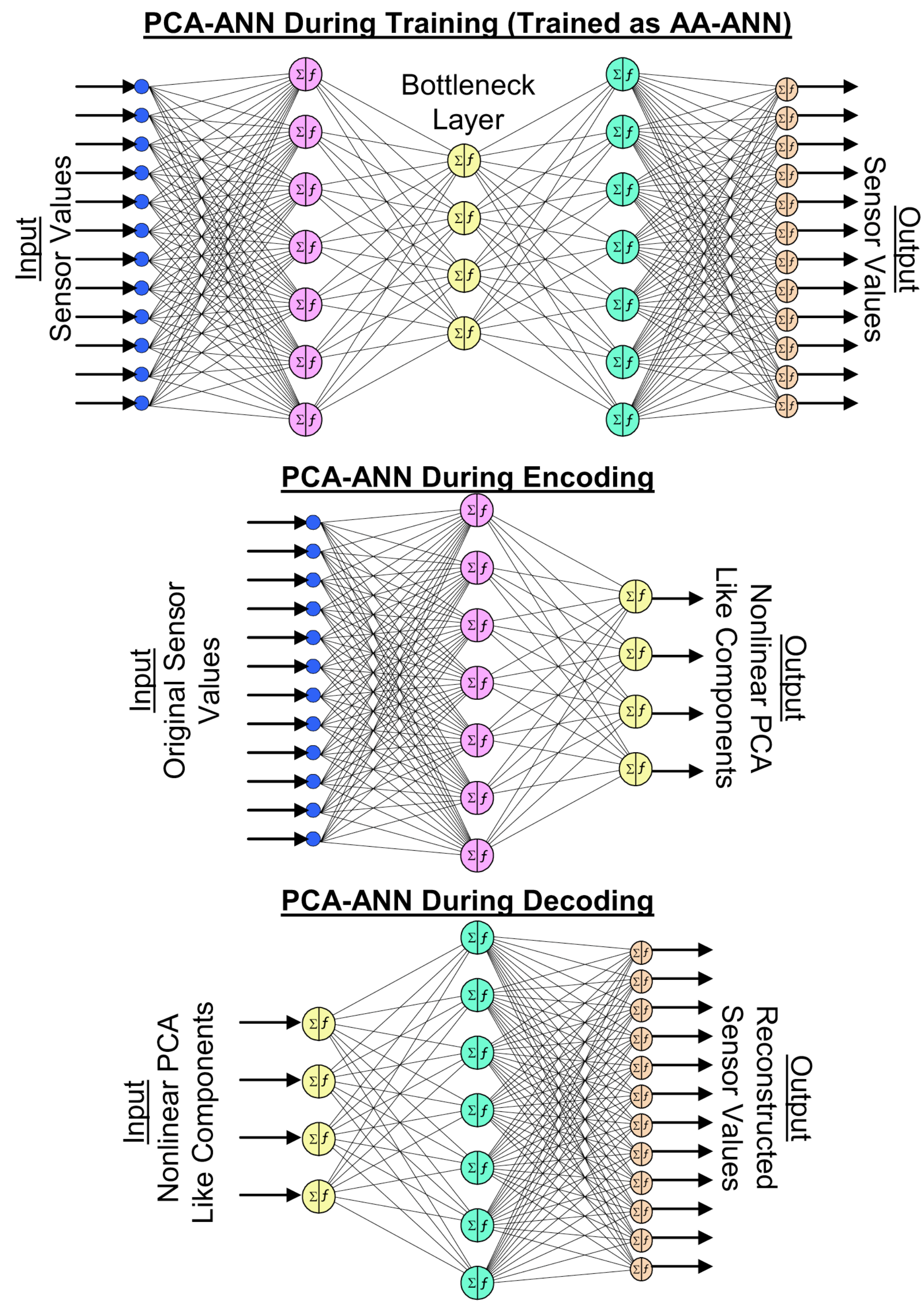

Figure 3. Top—illustration of a PCA-ANN configuration as it is trained to reproduce the input at the output (i.e., AA-ANN); middle—illustration of a PCA-ANN that encodes input sensor values to a set of nonlinear PCA-like components at the bottleneck layer; bottom—illustration of a PCA-ANN that decodes nonlinear PCA-like components and reconstructs the original sensor values.

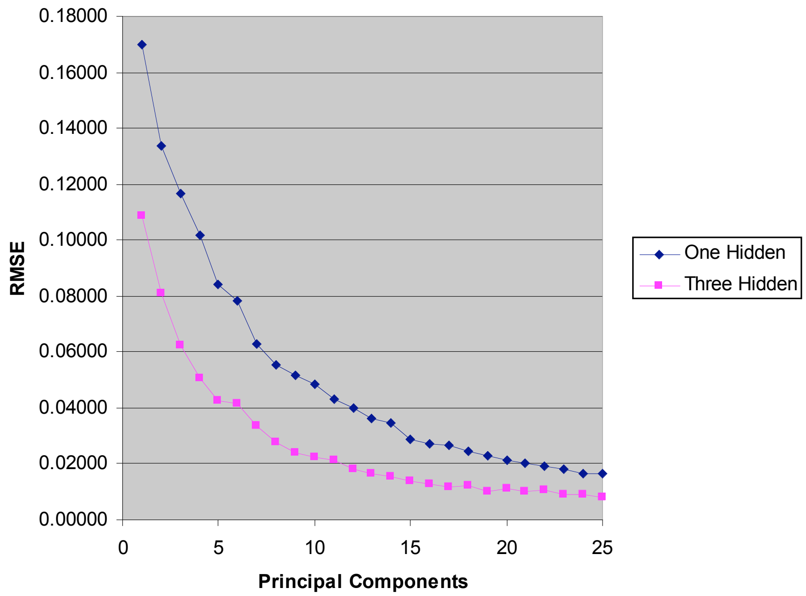

Figure 4. Two configurations of PCA-ANN to compress a set of 46 ARSA sensors to PCs. One ANN has a single hidden layer; the other has three hidden layers for more efficient data compression.

Figure 5 illustrates the ability of a nonlinear PCA-ANN to reproduce the sensor values at the output. The actual values of two cycles of the ps-5 sensor are shown with the decoded sensor values from the nonlinear PCA-ANN. The plots were staggered and offset to facilitate the comparison of the fine detail.

The prediction errors for a PCA-ANN, i.e., how well the network reproduces the data after it has been represented in a few PCs, was calculated for 46 ARSA sensors with six PCs. The errors were calculated for the set from approximately 1,700 test samples. In this case, 14,800 samples were used in developing the PCA-ANN. The mean error for all sensors as a percentage of their individual ranges is 3.8%.

As discussed above, deviations between actual and predicted individual sensors or groups of sensors can be incorporated in sensor validation and model based diagnostics. Most sensors in this study were modeled with high accuracy, such as the pressure sensor shown in Figure 5, making these models attractive in a health monitoring system for ARSA.

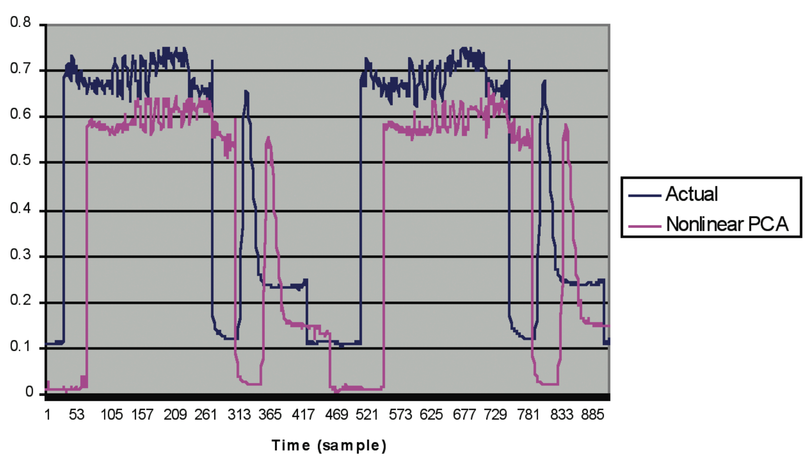

Figure 5. Actual values for sensor ps-5 and those decoded from the 46-60-6-60-46 five-layer PCA-ANN (the two traces are staggered for clarity).

CONCLUSION(S) AND RECOMMENDATIONS

The PCA–ANN successfully compressed the information from over 40 ARSA sensors into 6 nonlinear PCs that were then expanded into reasonable estimates of the original sensor data values with high accuracy. This research indicates that it would be possible to compress all of the ARSA SOH data into as few as six PCs that can then be sent to the data center where an accurate representation of the SOH data can be viewed. Current methodology relies on trained analysts to determine the six best sensors to be sent to the data center, losing most of the SOH information.

Monitoring systems with neural network algorithms can also be used to monitor the system real time for failures as they occur. Currently, most system failures are recognized days after the failure has occurred.

REFERENCES

Hayes, J. C., K. H. Abel, T. W. Bowyer, T. R. Heimbigner, M. E. Panisko, P. L. Reeder, J. I. McIntyre, R. C. Thompson, L. C. Todd, and R. A. Warner (1999). Operations of the Automated Radioxenon Sampler/Analyzer—ARSA, in *Proceedings of the 21st Seismic Research Symposium: Technologies for Monitoring the Comprehensive Nuclear-Test-Ban Treaty*, LA UR-99-4700, Vol. 2, pp. 249–260.

Heimbigner, T. R., J. I. McIntyre, T. W. Bowyer, J. C. Hayes, and M. E. Panisko (2002). Environmental monitoring of radioxenon in support of the radionuclide measurement system of the International Monitoring System, in *Proceedings of the 24th Seismic Research Review—Nuclear Explosion Monitoring: Innovation and Integration*, LA-UR-02-5048, Vol. 2, 694–700.

Heimbigner, T. R., T. W. Bowyer, J. C. Hayes, C. R. Hubbard, J. I. McIntyre, M. Panisko, M. Ripplinger, and R. Suarez (2004). State-of-health software for the Automated Radioxenon Sampler/Analyzer, in *Proceedings of the 26th Seismic Research Review: Trends in Nuclear Explosion Monitoring*, LA-UR-04-5801, Vol. 2, 710–718.

Karhunen, J. and J. Joutsensalo (1995). Generalizations of principal component analysis, optimization problems, and neural networks, *Neural Networks* 8: 549–562.

Karhunen, J., E. Oja, L. Wang, R. Vigaro, and J. Joutsensalo (1997). A class of neural networks for independent component analysis, *IEEE Trans. Neural Networks*, 8: 486–504.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986). *Learning Internal Representations by Error Propagation, Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, Vol. 1, Foundations, D. D. Rumelhart and J. L. McClelland (Eds.). Cambridge, Massachusetts; MIT Press, pp. 318–362.

Werbos, P. J. (1974). Ph.D thesis, Harvard University, Cambridge, Massachusetts.

Werbos, P. J. (1994). *The Roots of Back-Propagation*. New York: John Wiley & Sons.