NASA Contractor Report 195418

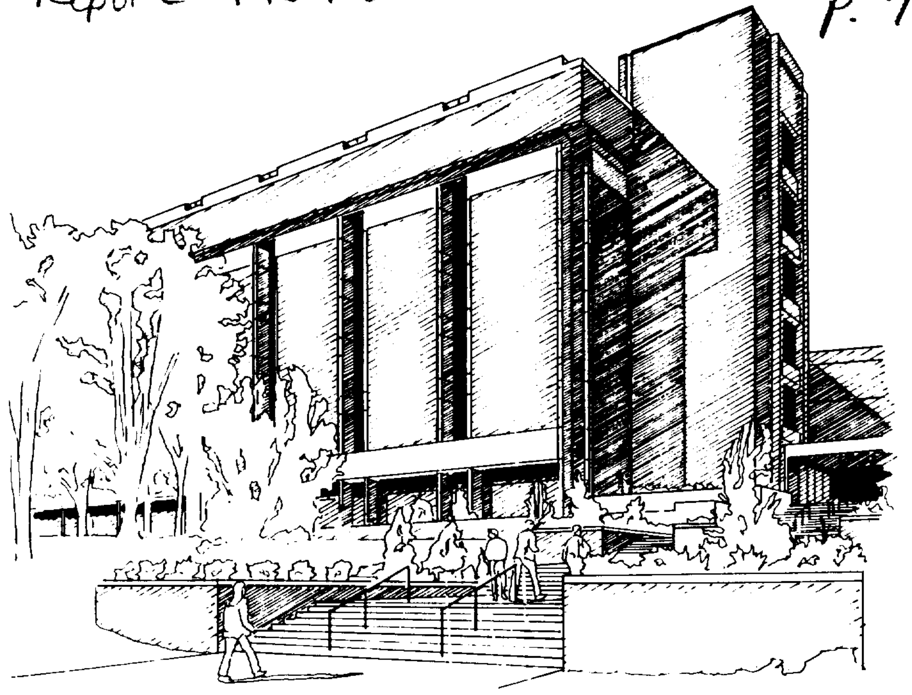
p. 79

N95-18045

Unclas

G3/15 0034969

(NASA-CR-195418) LVQ AND BACKPROPAGATION NEURAL NETWORKS APPLIED TO NASA SSME DATA Final Report (Cincinnati Univ.) 79 p



UNIVERSITY OF CINCINNATI
COLLEGE OF ENGINEERING

LVQ and Backpropagation Neural Networks Applied to NASA SSME Data

Technical Report: TR_156/6/93/ECE

Timothy F. Doniere and Atam P. Dhawan

This work was partially supported by grants from the NASA Lewis Research Center, and the NASA Space Engineering Research Center for System Health Management Technology at the University of Cincinnati

Department of Electrical and Computer Engineering

LVQ and Backpropagation Neural Networks Applied to NASA SSME Data

Technical Report # TR_156/6/93/ECE [1]

Timothy F. Doniere and Atam P. Dhawan
Department of Electrical and Computer Engineering
University of Cincinnati
Cincinnati, Ohio 45221 ML 30

[1]This work was partially supported by grants from the NASA Lewis Research Center, and the NASA Space Engineering Research Center for System Health Management Technology at the University of Cincinnati.

Abstract

Feedfoward neural networks with Backpropagation learning have been used as function approximators for modeling the Space Shuttle Main Engine (SSME) sensor signals. The modeling of these sensor signals is aimed at the development of a sensor fault detection system that can be used during ground test firings. The generalization capability of a neural network based function approximator depends on the training vectors which in this application may be derived from a number of SSME ground test-firings. This yields a large number of training vectors. Large training sets can cause the time required to train the network to be very large. Also, the network may not be able to generalize for large training sets. To reduce the size of the training sets, the SSME test-firing data is reduced using the Learning Vector Quantization (LVQ) based technique. Different compression ratios were used to obtain compressed data in training the neural network model. The performance of the neural model trained using reduced sets of training patterns is presented and compared with the performance of the model trained using complete data. The LVQ can also be used as a function approximator. The performance of the LVQ as a function approximator using reduced training sets is presented and compared with the performance of the backpropagation network.

1 INTRODUCTION

There is considerable interest within the space industry in improving the fault detection and isolation capabilities of rocket engine condition monitoring systems, both real-time and post-test. This requires developing accurate models of engine parameters based on other measured parameters. Developing accurate models is particularly difficult due to the highly complex, non-linear nature of rocket engines, the limited suite of measured parameters, and the large variability of behavior among engines of the same design.

It has been shown that neural networks can be used to approximate continuous functions [1, 2, 3]. Furthermore, neural networks are well-suited for problems in which the exact relationships between inputs and outputs are complex or unknown [4, 1]. These conclusions may be applied to dynamic systems if the system state is sufficiently represented in the inputs of the neural network. For these reasons, feedforward neural networks have been used to model critical parameters of the Space Shuttle Main Engine (SSME) during the start-up transient and they have been shown to be effective [4].

A task that is critical to the success of neural network modeling of complex, dynamic systems such as the SSME is the choice of input parameters. There are several constraints that complicate this task. First, while the instrumentation of the SSME is extensive, it is not complete. Therefore, it is unlikely that it will be possible to completely describe any subsystem input or output. This makes the extensive use of characteristic equations particularly difficult[5]. Second, as was discussed above, it is necessary to provide enough state information to model the desired parameter. Finally, it is not practical to use a large number of inputs for the following reasons. First, large input sets make training the network considerably more difficult. In addition, the input set should be small to reduce the hardware and/or computational complexity and to reduce the processing delay. This last consideration is especially important if the system is to be used for real-time modeling.

The effect of the input size on the system performance is further exacerbated by the use of multiple past values (or time windows) of each input. A time window of each input parameter is typically used in order to provide time dependent information. The size of the window multiplies the number of inputs to the network. For example, if 10 parameters are chosen as network inputs and a time window of the past ten values is used for each parameter, then the effective length of each input vector is 100.

The size of the training set can be kept small by choosing a small number of inputs. Another method to control the size of the training set is to reduce the number of training vectors by using some data compression scheme, such as the Learning Vector Quantization (LVQ) algorithm.

This paper will first present the design issues and methodology applied to the SSME parameter modeling, and compression of the training data. A presentation and discussion of results will follow.

2 DESIGN ISSUES AND METHODOLOGY

2.1 Learning Vector Quantization

The LVQ, in principle, is an auto-associative, supervised nearest-neighbor classifier. Using competitive learning, this paradigm can classify the input vector as one of the output classes by developing decision surfaces in the multi-dimensional feature space. It is a special case of the 'self organizing feature map' which optimally assigns K reference vectors $m_i \in \Re^n$,i = 1, 2, ... K to x $\in \Re^n$ such that the density of each m_i best represents the density function p(x).

In general the LVQ network can be implemented by a 2-layered feedforward topology consisting of an input and an output layer. The output layer can be initialized using either K random vectors or arbitrary samples taken from the data set. Alternately, certain code book vectors based on *a priori* knowledge can be used as K reference seed vectors. This procedure may provide faster convergence and better performance. In the learning process, the training vectors are iteratively presented to the input layer. When an input vector is presented it is compared with the output vectors. The vector related to the nearest (Euclidean distance) node in the output layer is used to compute the error between the input vector and the nearest node vector. Based on the learning rate, a small part of this error is used to update the output vector such that the output node vector 'more' closely resembles the input vector. The iterative process of training is continued until the error statistics satisfy a predefined convergence criterion. Alternately the number of iterations can be set *a priori* depending upon the application.

The mathematical representation of the learning paradigm described above is outlined as follows:

$$m_c(t+1) = m_c(t) + \alpha(t) * (x(t) - m_c(t)), \tag{1}$$

where $c = \arg_i \min \|x - m_i\|$ and $0 < \alpha(t) < 1$. The learning rate $\alpha(t)$ is linearly decreased as the number of iterations increases.

In the SSME application for data reduction or parameter estimation, we associate input-output vectors through the LVQ topology. Let us assume that there are 'n' SSME parameters to be used as inputs to the neural network model of the selected output SSME sensor variable. Each vector will contain 'n+1' elements, where the (n+1)th element is the 'label' for that vector. The 'label' represents the value of the modeled parameter. The input layer of the LVQ network will contain 'n+1' nodes. The output layer will consist of 'K' nodes depending on the ratio of data compression. For example, during the start-up period the SSME test firing data may have approximately 150 patterns. If three SSME test firings are to be used to train the neural network model of a selected sensor variable, there will be a total of 450 patterns to be used in training the neural network model. To obtain a data compression of 3:1 the output layer of the LVQ network should have approximately 150 nodes, each representing a 'n+1' dimensional vector. Each input vector represents data sampled every 40 milliseconds.

The result of training the LVQ is a set of vectors called the codebook vectors. The density of the set of codebook vectors represents the density of the set of training vectors. The codebook vectors can then be used as a training set for other neural network paradigms or directly used as an estimator.

The codebook vectors can be directly used to estimate the desired parameter. Each input pattern is presented to the network and, as in training, the closest codebook vector is found. The label of the winning vector represents the estimated value of the desired parameter.

Another method for using LVQ as an estimator is to train using only the 'n' dimensional input vectors, that is, leave out the target value. Labels are assigned to each vector after training. For each training vector, the closest or 'winning' codebook vector is found. The codebook vector's label is computed by averaging the target values for all the training vectors which were assigned to that codebook vector.

2.2 Feedforward Neural Networks

Feedforward neural networks with a single hidden layer were used in this paper. Each layer consists of processing elements called nodes.

Each node of a given layer receives input from all the nodes in the previous layer and sends its output to each node in the following layer. The connections are unidirectional and have weights associated with them. There are no connections between nodes within a layer and no connections bridging layers. The relationship between the input and the output of a node in the hidden or output layers is determined by the activation function, a.

Using the sigmoid activation function, the output of each node may be represented as:

$$o_i = a(\mathrm{net}_i) = \frac{1}{1 + e^{-\mathrm{net}_i}},$$

where net_i is the net input to node i. More specifically,

$$\mathrm{net}_i = \theta_i + \sum_j w_{ij} \cdot o_j,$$

where θ_i is the bias term for unit i, o_j is the output of node j in the previous layer, and w_{ij} is the weight connected to node j in the previous layer. The values of the weights and biases are adjusted during the learning process.

The multiple-input single-output network can be described as a mapping function which uniformly approximates a function. The approximation is achieved by using the backpropagation algorithm, a supervised learning procedure based on the Generalized Delta Rule. Backpropagation computes the weights and bias terms to minimize the mean squared error between the values predicted by the network and the desired values.

2.3 Normalization

When computing distances between vectors, the data must be normalized in some way. This is necessary to insure that one dimension does not dominate the distance calculation. There are several ways to accomplish this. Two different schemes were investigated here.

One method for normalization is to manipulate the data so that each dimension has zero mean and unit variance. This is accomplished by computing the mean and variance for each dimension in the data. Each data element is normalized by subtracting its corresponding mean and dividing by its standard deviation.

Another method for normalization is scaling. In this method, the data is made to fall within a certain range, for example, (-.5,.5). For each data dimension, a maximum and minimum value are set. Each data element is then mapped to the given range, the maximum data value is mapped to the maximum range value, and the minimum data value is mapped to the minimum range value, with all others mapped within the range.

3 RESULTS

In this paper, the selection of input variables was performed for the start-up period of the SSME test firings. A Genetic algorithm was used to select the input parameters[6]. The list of selected parameters is shown in Table 1.

Table 1: Parameter Descriptions

PID	*Description*
21	Main Combustion Chamber Oxidizer Injection Temperature
58	Fuel Preburner Chamber Pressure
209	High Pressure Oxidizer Pump Inlet Pressure
233†	High Pressure Oxidizer Turbine Discharge Temperature
734	Low Pressure Oxidizer Pump Shaft Speed
951	High Pressure Oxidizer Pump Primary Seal Drain Pressure
1050	Oxidizer Tank Discharge Temperature

† the modeled parameter

The parameter that was modeled is the SSME's High Pressure Oxidizer Turbine (HPOT) discharge temperature, which has a Parameter IDentification (PID) number of 233.

Using the selected inputs, the training data was created for the time period 0.2 to 6.0 seconds. Four SSME ground test-firings b1046, b1060, b1070 and b1077 were used for training. A time window of 5 past values was used for each input variable. Eight SSME test-firings were used as validation sets.

To evaluate the different normalization schemes, the training set was compressed using LVQ to 20% of the data (117 nodes). The performance of the LVQ estimator was then evaluated for each normalization scheme. The results, as represented by the normalized mean square error (NMSE) and the maximum percent error are shown in Table 3 and density of the set of codebook vectors represents the density of the set of training vectors. The codebook vectors can then be used as a training set for other neural network paradigms or directly used as an estimator.

The codebook vectors can be directly used to estimate the desired parameter. Each input pattern is presented to the network and, as in training, the closest codebook vector is found. The label of the winning vector represents the estimated value of the desired parameter.

Another method for using LVQ as an estimator is to train using only the 'n' dimensional input vectors, that is, leave out the target value. Labels are assigned to each vector after training. For each training vector, the closest or 'winning' codebook vector is found. The codebook vector's label is computed by averaging the target values for all the training vectors which were assigned to that codebook vector.

2.2 Feedforward Neural Networks

Feedforward neural networks with a single hidden layer were used in this paper. Each layer consists of processing elements called nodes.

Each node of a given layer receives input from all the nodes in the previous layer and sends its output to each node in the following layer. The connections are unidirectional and have weights associated with them. There are no connections between nodes within a layer and no connections bridging layers. The relationship between the input and the output of a node in the hidden or output layers is determined by the activation function, a.

Using the sigmoid activation function, the output of each node may be represented as:

$$o_i = a(\text{net}_i) = \frac{1}{1 + e^{-\text{net}_i}},$$

where net_i is the net input to node i. More specifically,

$$\text{net}_i = \theta_i + \sum_j w_{ij} \cdot o_j,$$

where θ_i is the bias term for unit i, o_j is the output of node j in the previous layer, and w_{ij} is the weight connected to node j in the previous layer. The values of the weights and biases are adjusted during the learning process.

The multiple-input single-output network can be described as a mapping function which uniformly approximates a function. The approximation is achieved by using the backpropagation algorithm, a supervised learning procedure based on the Generalized Delta Rule. Backpropagation computes the weights and bias terms to minimize the mean squared error between the values predicted by the network and the desired values.

2.3 Normalization

When computing distances between vectors, the data must be normalized in some way. This is necessary to insure that one dimension does not dominate the distance calculation. There

Table 2: Error Statistics For Error Due To LVQ Compression

Compression Ratio	*NRMS*	*Max. % Error*
2:1	0.014478	7.869942
3:1	0.019473	15.464482
4:1	0.022115	15.464482
5:1	0.024796	15.464482
6:1	0.046300	22.085990

Table 4. The results indicate that the error performance of the LVQ estimator using the scaling normalization scheme was better than the mean and variance normalization scheme. The scaling normalization is simpler and more universal to implement since the maximum and minimum data values are known the scheme can easily be applied to any new data set. For these reasons scaling was used throughout the study.

To evaluate the different LVQ estimator schemes, the training data was compressed to 20% of the data. The performance of each LVQ estimator schemes was evaluated. The results are shown in Table 5 and Table 6. The labeled LVQ error performance was better than the error performance of LVQ which was labeled after training. For the remainder of this work the labeled LVQ scheme was used.

To evaluate LVQ as a data compressor, the training data was compressed to 50%, 33%, 25%, and 20% of the data. This is accomplished by using 292, 195, 146, and 117 output nodes, respectively. The error introduced by compression is represented by the NMSE, that is, the error between the actual data vector and its corresponding codebook vector. Note that this error is for the entire vector, where all other errors reported in this study are for PID 233. The results are shown in Table 2.

To evaluate the performance of the compressed training data with different compression ratios, feedforward neural networks were trained for 20,000 cycles using the standard back-propagation learning algorithm. Each network had one hidden layer with 10 processing units. For each compression ratio, the neural network model for PID 233 was evaluated using the 8 SSME test-firings. The results are shown in Tables 12, 13, 14, 15, and 16. Table 12 shows the errors for the backpropagation network trained with the full training set. Comparing these results to the errors for the network trained with the reduced training sets, Tables 13, 14, 15, and 16, we see that the ability of the network to generalize is not significantly affected as the training data is reduced. The training time is reduced as the training set is reduced. Therefore, the network training time can be significantly reduced without the network losing its ability to generalize.

The LVQ estimator was also evaluated using the compressed training sets. The results are shown in Tables 7, 8, 9, 10, and 11. Again the ability to generalize was not significantly affected by reducing the training set, while training times were reduced.

Comparing the performance of the backpropagation network to the LVQ estimator, we see that the LVQ learned the training sets better than the backpropagation. But backpropagation was able to better generalize which can be seen by comparing the errors for the validation sets with the LVQ validation set errors.

Plots of the output of the networks are also shown. The dotted line indicates the predicted value and the solid line the actual value. The error plot represents the difference between these two.

4 CONCLUSIONS AND FUTURE WORK

The compression errors show that LVQ can compress the data set and produce a set which reasonably represents the original data. This smaller data set can be used to train other networks or as the codebook set for the LVQ estimator. This work is continuing to standardize the LVQ compression scheme. Data compression will be especially important when modeling parameters during SSME mainstage operation since the data sets for this time period are very large.

5 ACKNOWLEDGMENTS

The authors gratefully acknowledge the help and valuable contributions of Claudia Meyer of Sverdrup Technology Inc.

References

[1] S. Chen, S. A. Billings, and P. M. Grant. Non-Linear Systems Identification Using Neural Networks. Research Report 370, University of Edinburgh, Mayfield Road, Edinburgh, Scotland, August 1989.

[2] G. Cybenko. Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems*, 2:303–314, 1989.

[3] K. Funahashi. On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks*, 2:183–192, 1989.

[4] Claudia M. Meyer and William A. Maul. The Application of Neural Networks to the SSME Startup Transient. AIAA 91-2530, July 1991.

[5] D. K. Makel, W. H. Flaspohler, and T. W. Bickmore. Sensor Data Validation and Reconstruction, Phase 1: System Architecture Study. NASA CR 187122, 1991. Contract No. NAS3-25883.

[6] Charles C. Peck, Atam P. Dhawan, and Claudia M. Meyer. Selection of Input Variables for SSME Parameter Modeling using Genetic Algorithms and Neural Networks. Dept. of Elect. and Comp. Eng., Univ. of Cincinnati

[7] Alianna Maren, Craig Harston, and Robert Pap. *Handbook of Neural Computing Applications.* Academic Press, 1990.

[8] James A. Freeman and David M. Skapura. *Neural Networks Algorithms Applications and Programming Techniques.* Addison-Wesley, 1991.

[9] Tuevo Kohonen. *Self-Organization and Associative Memory.* Springer-Verlag, 1989.

[10] Tuevo Kohonen, Jari Kangas, Jorma Laaksonen, and Kari Torkkola. LVQ_PAK The Learning Vector Quantization Program Package. Helsinki University of Technology, October, 1992.

Table 3: Error Statistics Using Mean and Variance Normalization

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.012803	5.081546
B1060	T	0.017527	9.557117
B1061	V	0.021687	13.443471
B1062	V	0.039671	18.678014
B1063	V	0.038932	14.067005
B1066	V	0.045194	16.609852
B1067	V	0.038573	14.067005
B1070	T	0.014358	6.802264
B1071	V	0.053139	20.393750
B1072	V	0.026445	8.525152
B1075	V	0.018436	9.557117
B1077	T	0.015274	8.007803

Table 4: Error Statistics Using Scaling Normalization

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.014284	6.583544
B1060	T	0.021480	8.996746
B1061	V	0.017267	9.253903
B1062	V	0.022484	7.258064
B1063	V	0.028766	6.890465
B1066	V	0.025609	8.072915
B1067	V	0.028394	7.329164
B1070	T	0.014477	5.014370
B1071	V	0.040510	15.710223
B1072	V	0.024285	5.342574
B1075	V	0.026553	11.509344
B1077	T	0.008872	4.452689

Table 5: Error Statistics For LVQ Labeling After Training

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.012803	5.081546
B1060	T	0.017527	9.557117
B1061	V	0.021687	13.443471
B1062	V	0.039671	18.678014
B1063	V	0.038932	14.067005
B1066	V	0.045194	16.609852
B1067	V	0.038573	14.067005
B1070	T	0.014358	6.802264
B1071	V	0.053139	20.393750
B1072	V	0.026445	8.525152
B1075	V	0.018436	9.557117
B1077	T	0.015274	8.007803

Table 6: Error Statistics For LVQ Labeled During Training

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.014808	6.857448
B1060	T	0.022716	8.996746
B1061	V	0.017281	9.253903
B1062	V	0.022996	7.603686
B1063	V	0.029167	6.890465
B1066	V	0.026714	8.072915
B1067	V	0.028096	7.329164
B1070	T	0.014349	4.887298
B1071	V	0.040581	15.710223
B1072	V	0.023829	5.342574
B1075	V	0.023829	5.342574
B1077	T	0.008539	4.449154

Table 7: Error Statistics For LVQ As Estimator Trained with Full Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.000000	0.000000
B1060	T	0.000000	0.000000
B1061	V	0.019805	10.890798
B1062	V	0.022290	8.355086
B1063	V	0.030517	9.546048
B1066	V	0.033532	8.623061
B1067	V	0.028992	10.397836
B1070	T	0.000000	0.000000
B1071	V	0.035585	13.344638
B1072	V	0.024342	8.001184
B1075	V	0.023831	11.354581
B1077	T	0.000000	0.000000

Table 8: Error Statistics For LVQ As Estimator Trained with 50% of the Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.005850	2.197425
B1060	T	0.013230	8.183566
B1061	V	0.015815	8.186236
B1062	V	0.020811	8.127593
B1063	V	0.027696	9.187287
B1066	V	0.029318	8.738959
B1067	V	0.026962	9.269014
B1070	T	0.013322	7.529416
B1071	V	0.037448	14.526595
B1072	V	0.025926	8.693025
B1075	V	0.026295	11.256074
B1077	T	0.008661	5.751390

Table 9: Error Statistics For LVQ As Estimator Trained with 33% of the Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.008989	3.371113
B1060	T	0.017070	7.753298
B1061	V	0.018969	10.807628
B1062	V	0.021522	7.519230
B1063	V	0.029663	9.540644
B1066	V	0.028300	9.020437
B1067	V	0.028488	8.318269
B1070	T	0.014406	8.458014
B1071	V	0.038819	15.144536
B1072	V	0.026113	8.369441
B1075	V	0.026796	11.913180
B1077	T	0.009234	5.990783

Table 10: Error Statistics For LVQ As Estimator Trained with 25% of the Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.011926	5.292437
B1060	T	0.021922	8.624715
B1061	V	0.017168	9.632886
B1062	V	0.021273	7.563217
B1063	V	0.029269	8.029670
B1066	V	0.026163	8.887903
B1067	V	0.027523	7.742225
B1070	T	0.015476	6.848912
B1071	V	0.039532	15.650424
B1072	V	0.024587	5.673757
B1075	V	0.026525	11.132799
B1077	T	0.008601	5.009275

Table 11: Error Statistics For LVQ As Estimator Trained with 20% of the Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.014284	6.583544
B1060	T	0.021480	8.996746
B1061	V	0.017267	9.253903
B1062	V	0.022484	7.258064
B1063	V	0.028766	6.890465
B1066	V	0.025609	8.072915
B1067	V	0.028394	7.329164
B1070	T	0.014477	5.014370
B1071	V	0.040510	15.710223
B1072	V	0.024285	5.342574
B1075	V	0.026553	11.509344
B1077	T	0.008872	4.452689

Table 12: Error Statistics For Backprop Trained with Full Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.0124322	3.8644
B1060	T	0.0220291	7.5323
B1061	V	0.0155388	8.1503
B1062	V	0.0281898	8.06953
B1063	V	0.0285942	4.95574
B1066	V	0.0252927	6.51426
B1067	V	0.0291385	6.24563
B1070	T	0.0166826	4.96011
B1071	V	0.0434168	16.2365
B1072	V	0.0212164	3.25108
B1075	V	0.0199444	7.92355
B1077	T	0.0129176	5.65356

Table 13: Error Statistics For Backprop Trained with 50% of the Data

Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.0135083	4.88156
B1060	T	0.0225959	8.42676
B1061	V	0.0155132	8.48681
B1062	V	0.0268862	7.53272
B1063	V	0.0279912	5.58431
B1066	V	0.0232444	6.49096
B1067	V	0.0284255	5.84923
B1070	T	0.0178881	5.19813
B1071	V	0.0436407	16.4448
B1072	V	0.0221922	3.73636
B1075	V	0.020972	8.95508
B1077	T	0.0123256	5.04442

Table 14: Error Statistics For Backprop Trained with 33% of the Data

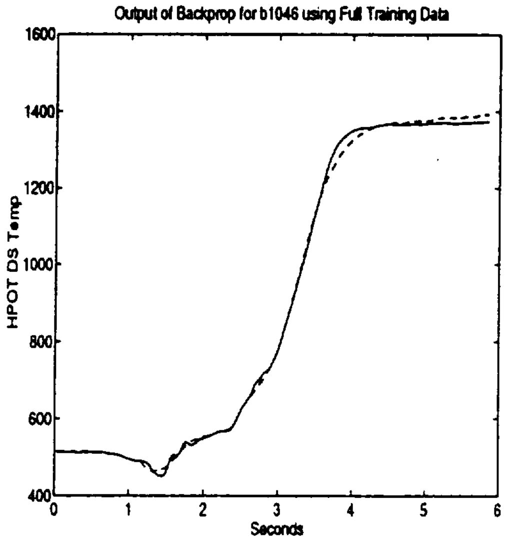
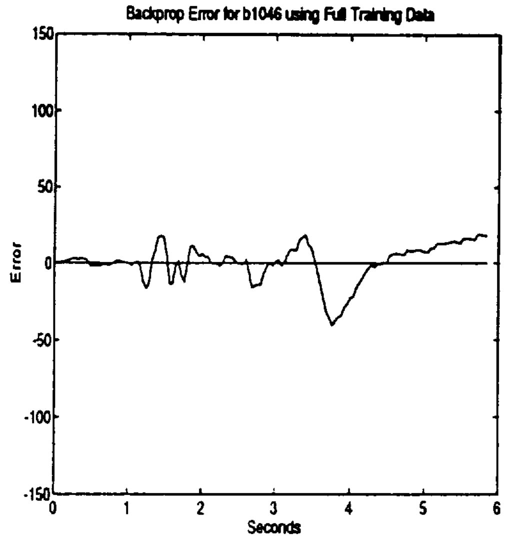
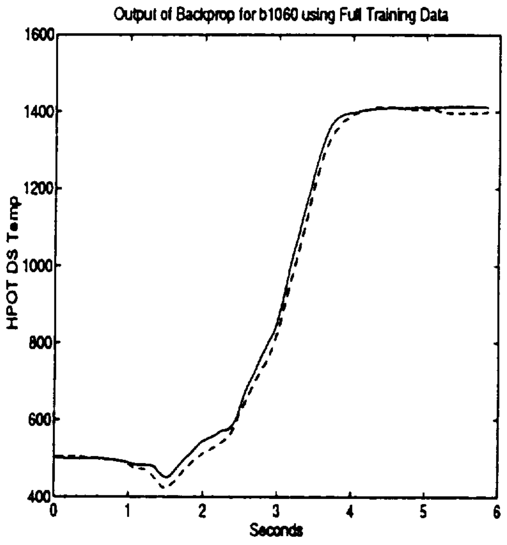
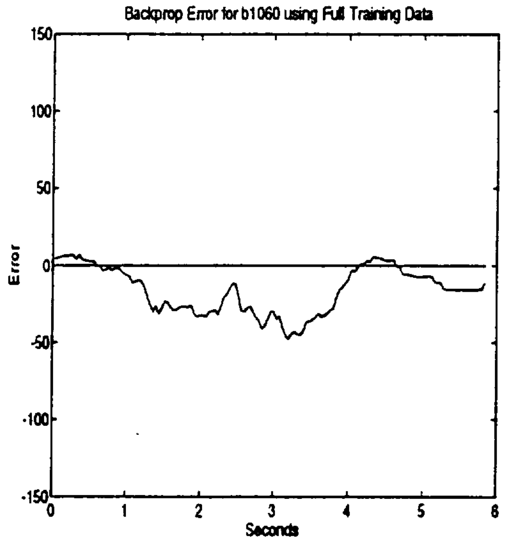
Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.0122172	3.73747
B1060	T	0.0232073	7.97284
B1061	V	0.0159075	8.23784
B1062	V	0.0267634	8.11873
B1063	V	0.0279388	6.01757
B1066	V	0.0234415	6.05602
B1067	V	0.0286324	5.81188
B1070	T	0.0187359	5.49698
B1071	V	0.0442942	16.7554
B1072	V	0.0222597	4.46895
B1075	V	0.0209986	9.3965
B1077	T	0.012065	4.72981

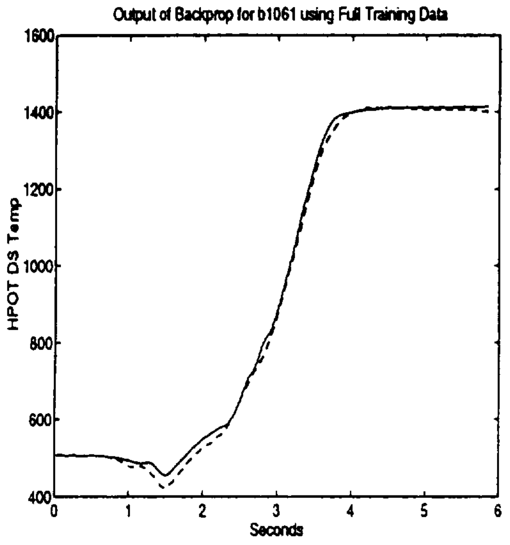
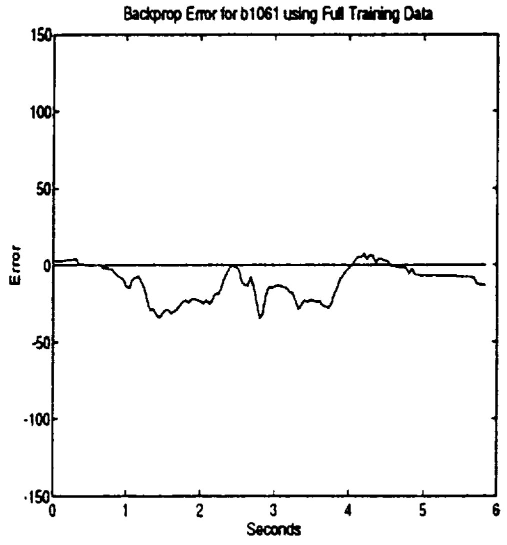
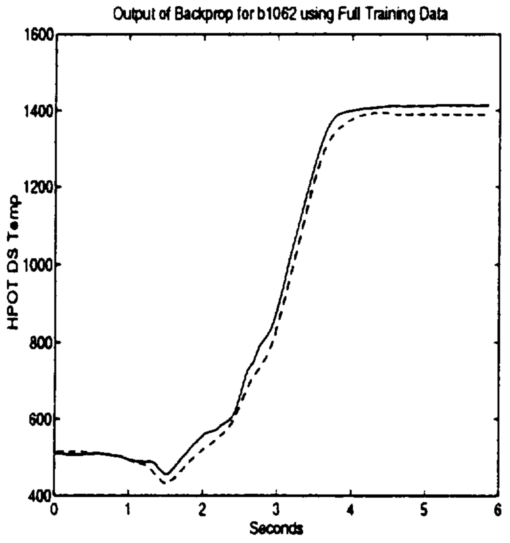
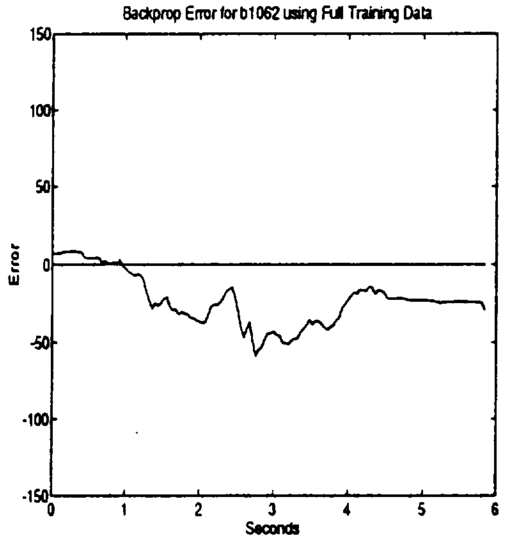
Table 15: Error Statistics For Backprop Trained with 25% of the Data

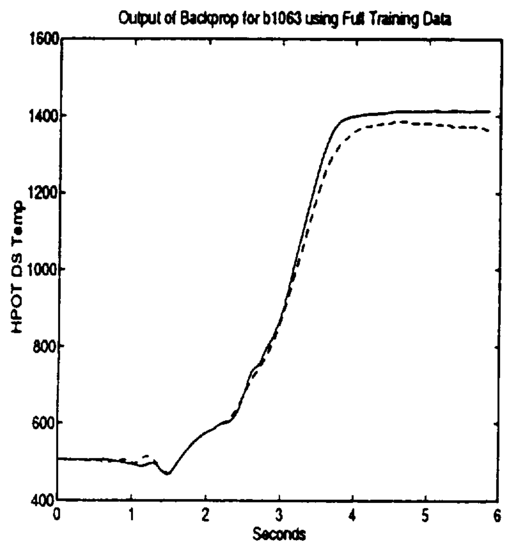
Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.0120872	3.82172
B1060	T	0.0239868	8.04438
B1061	V	0.0164772	8.29851
B1062	V	0.0267587	7.96939
B1063	V	0.0283388	6.17946
B1066	V	0.0236575	6.40781
B1067	V	0.0291795	6.11476
B1070	T	0.0184146	6.09501
B1071	V	0.0452917	16.5497
B1072	V	0.0224161	4.74605
B1075	V	0.0219442	9.42541
B1077	T	0.0117136	4.75593

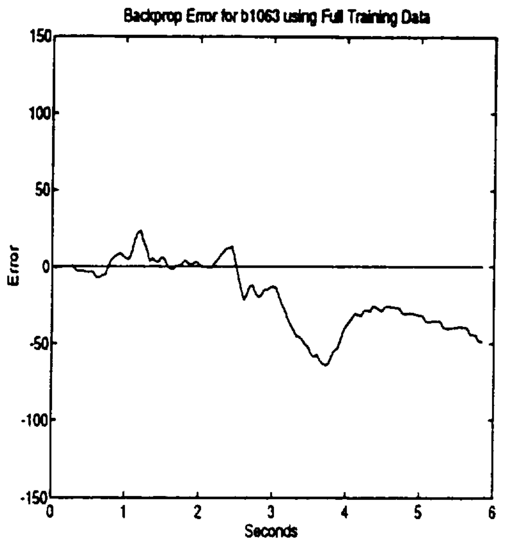
Table 16: Error Statistics For Backprop Trained with 20% of the Data

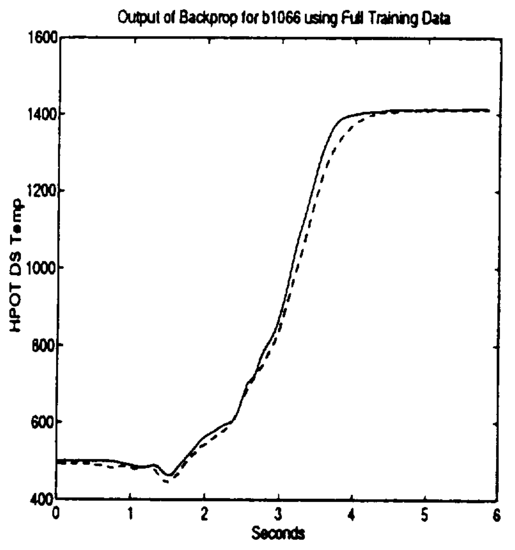
Test Firing	*Training/ Validation*	*NRMS*	*Max. % Error*
B1046	T	0.0129552	3.62103
B1060	T	0.0245237	7.90749
B1061	V	0.0169781	8.23699
B1062	V	0.0274555	7.96652
B1063	V	0.0283145	5.90633
B1066	V	0.0243938	6.18729
B1067	V	0.0293336	6.31275
B1070	T	0.01755	5.233
B1071	V	0.0456178	16.7355
B1072	V	0.022092	3.4244
B1075	V	0.0224649	9.23946
B1077	T	0.0115751	4.67811

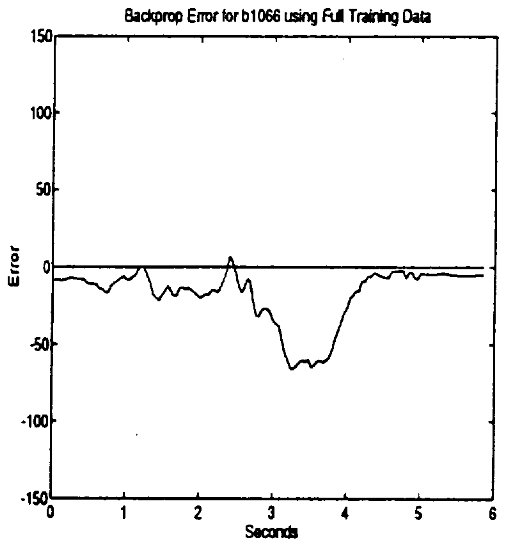

Output of Backprop for b1046 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1046 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

Output of Backprop for b1060 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1060 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

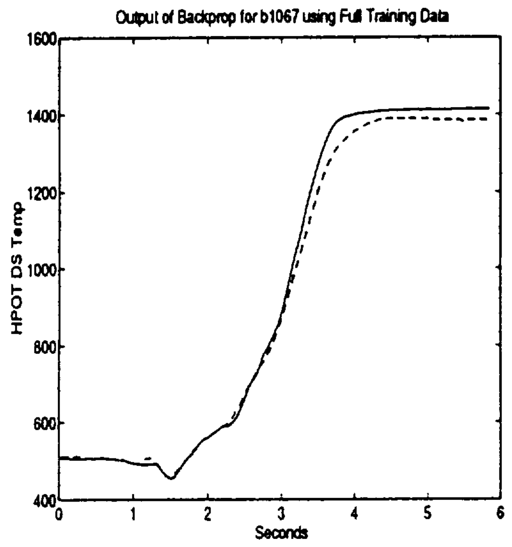

Output of Backprop for b1061 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1061 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

Output of Backprop for b1062 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1062 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

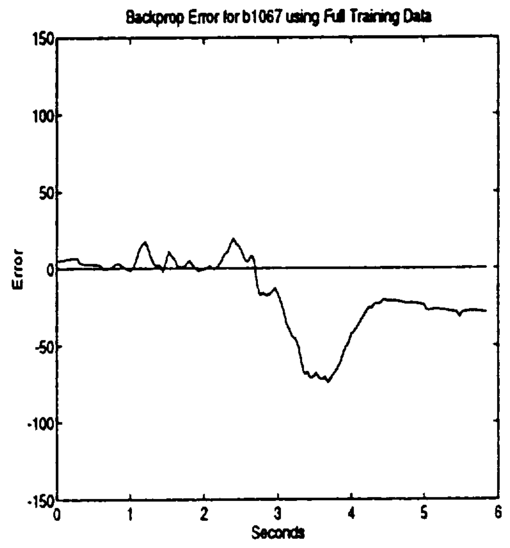

Output of Backprop for b1063 using Full Training Data
HPOT DS Temp
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6
Seconds

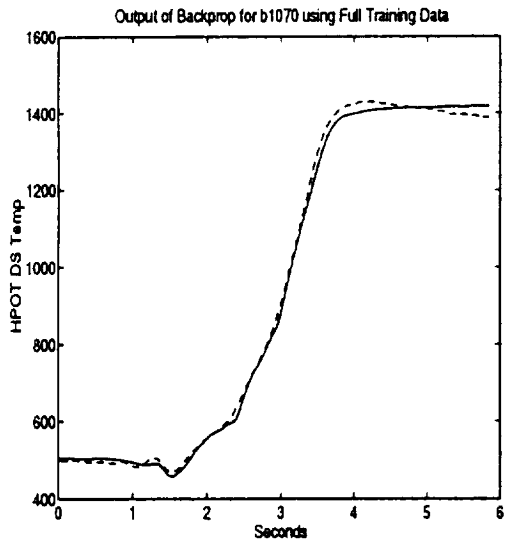

Backprop Error for b1063 using Full Training Data
Error
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6
Seconds

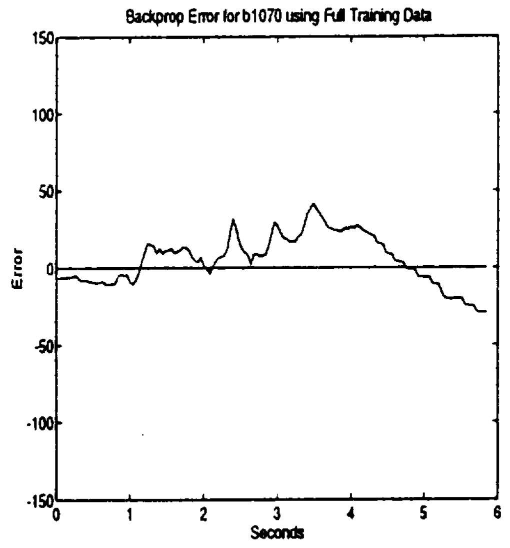

Output of Backprop for b1066 using Full Training Data
HPOT DS Temp
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6
Seconds

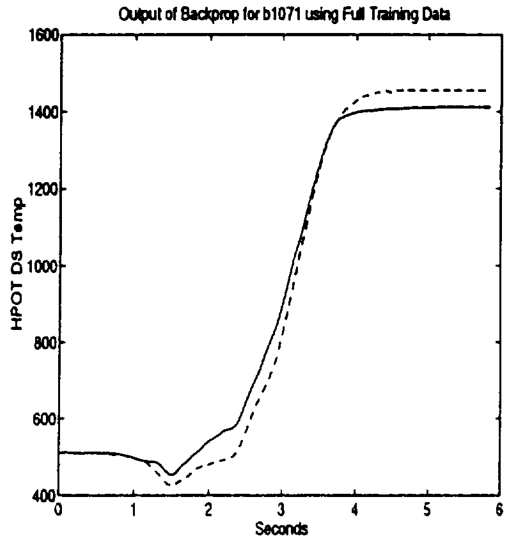

Backprop Error for b1066 using Full Training Data
Error
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6
Seconds

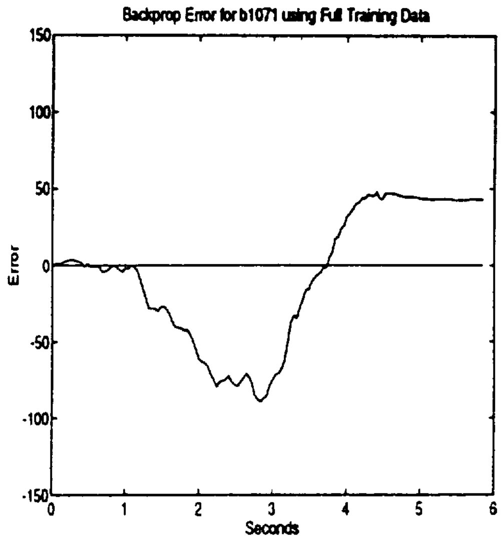

Output of Backprop for b1067 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

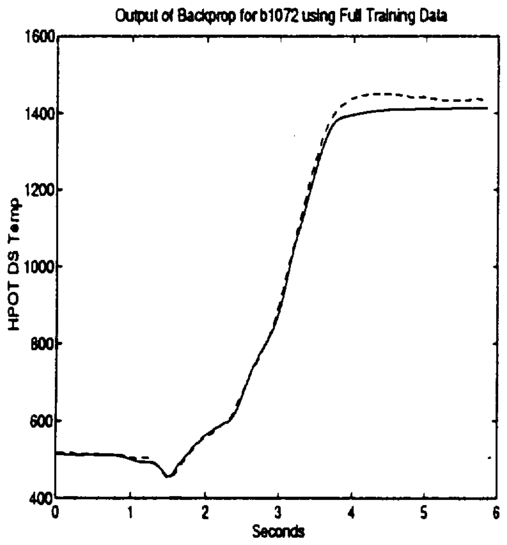

Backprop Error for b1067 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

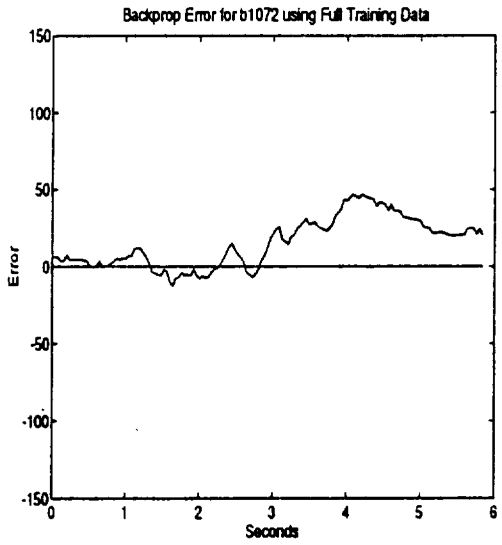

Output of Backprop for b1070 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

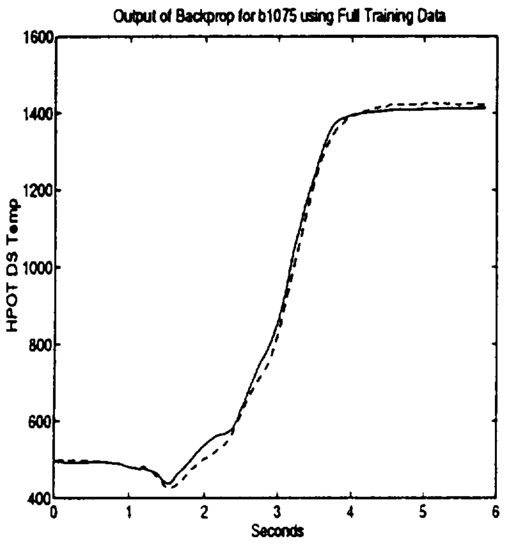

Backprop Error for b1070 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

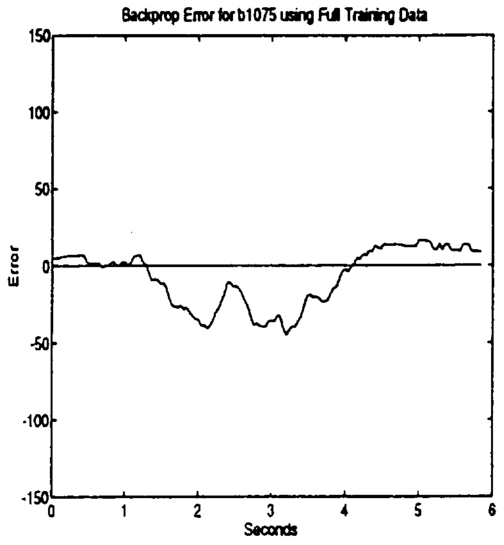

Output of Backprop for b1071 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

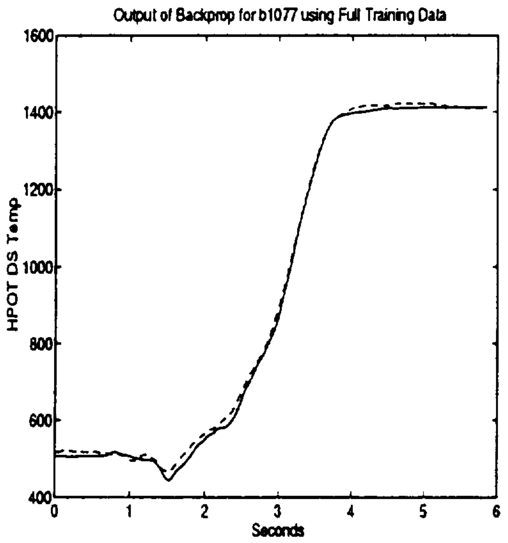

Backprop Error for b1071 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

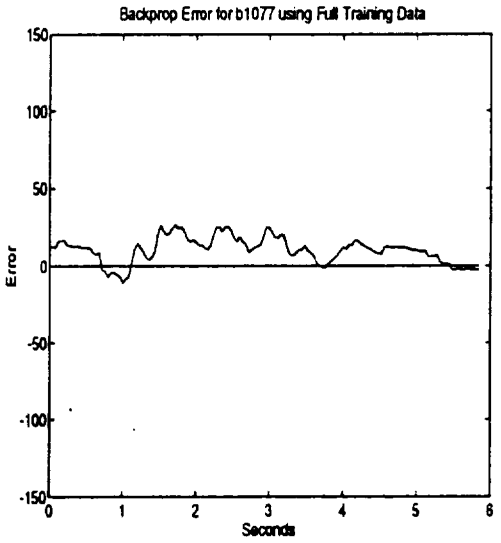

Output of Backprop for b1072 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

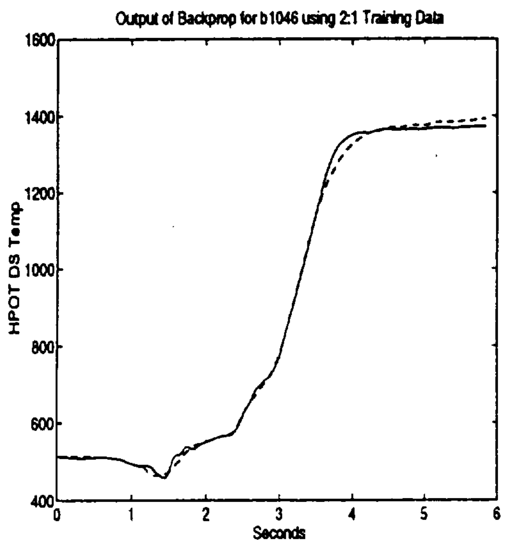

Backprop Error for b1072 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

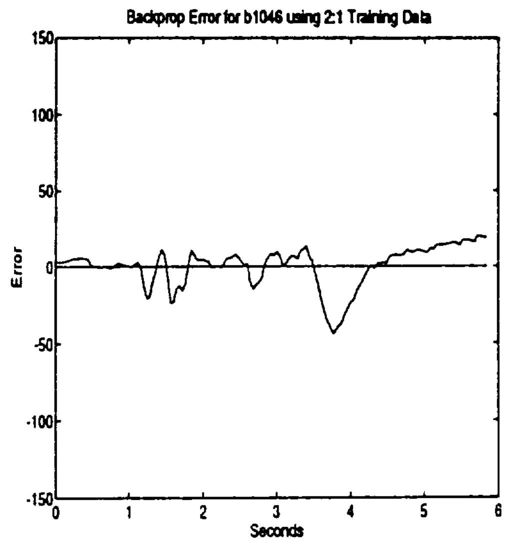

Output of Backprop for b1075 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

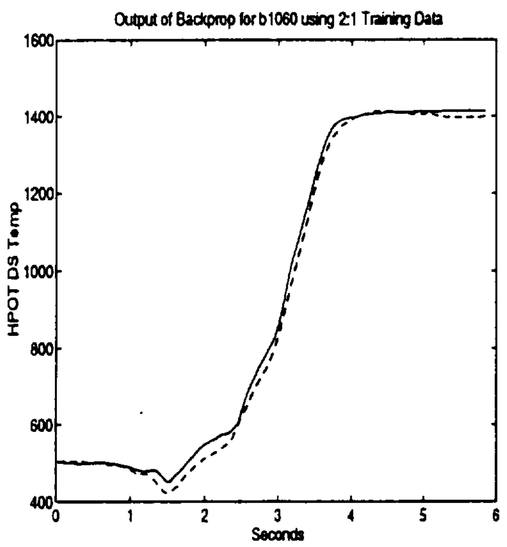

Backprop Error for b1075 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

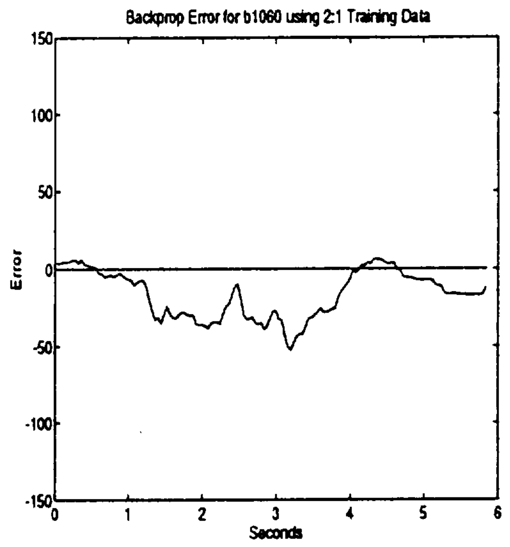

Output of Backprop for b1077 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

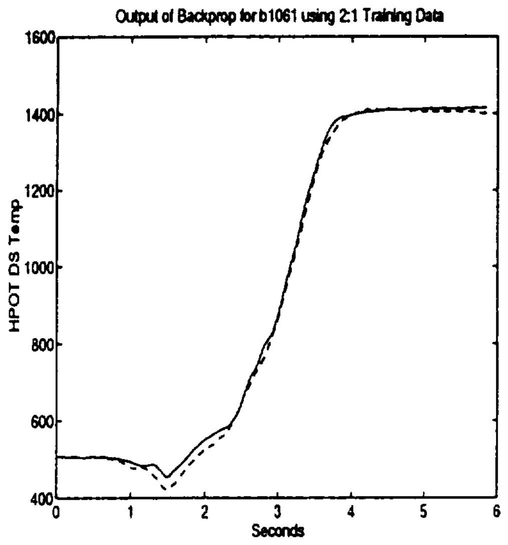

Backprop Error for b1077 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

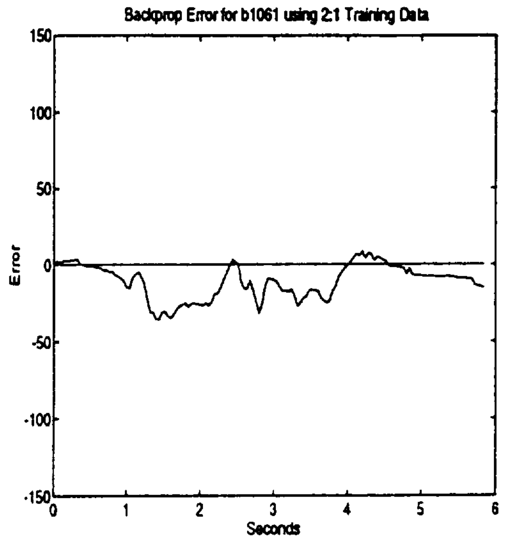

Output of Backprop for b1046 using 2:1 Training Data
HPOT DS Temp
Seconds

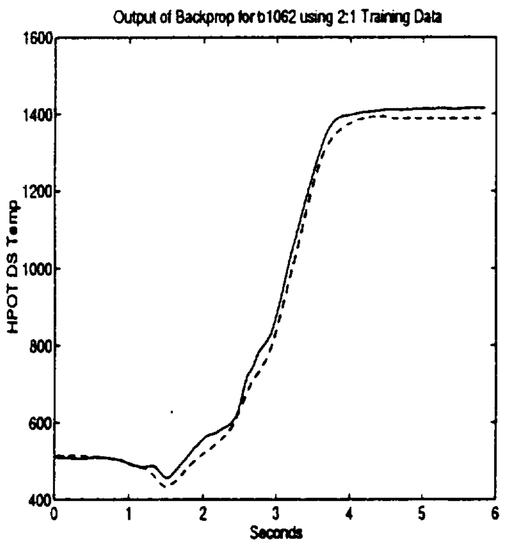

Backprop Error for b1046 using 2:1 Training Data
Error
Seconds

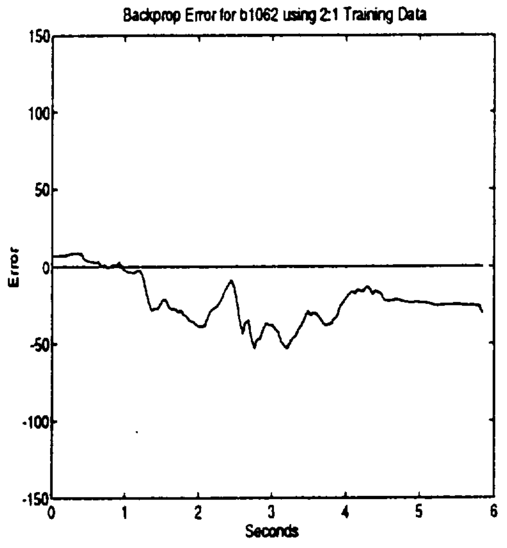

Output of Backprop for b1060 using 2:1 Training Data
HPOT DS Temp
Seconds

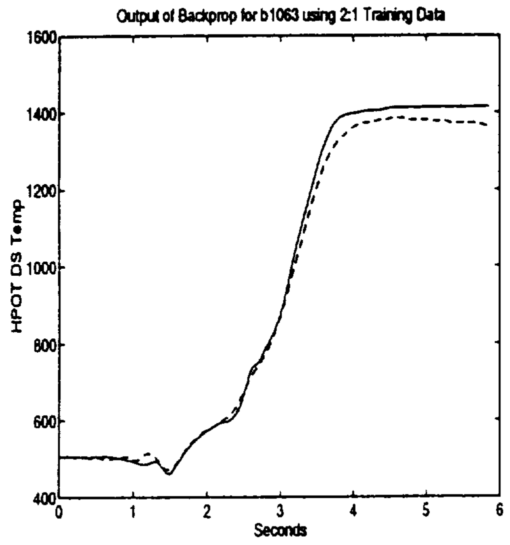

Backprop Error for b1060 using 2:1 Training Data
Error
Seconds

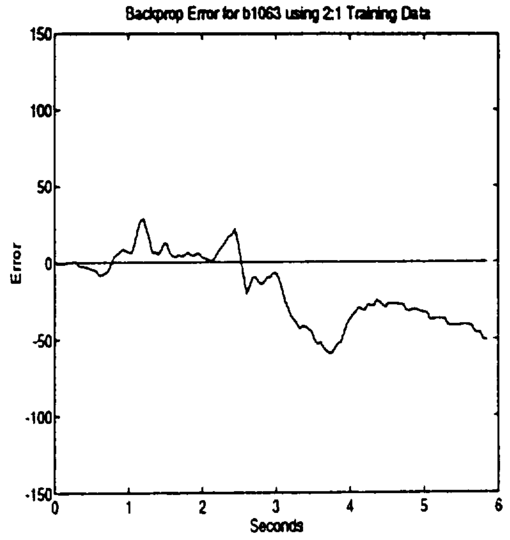

Output of Backprop for b1061 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

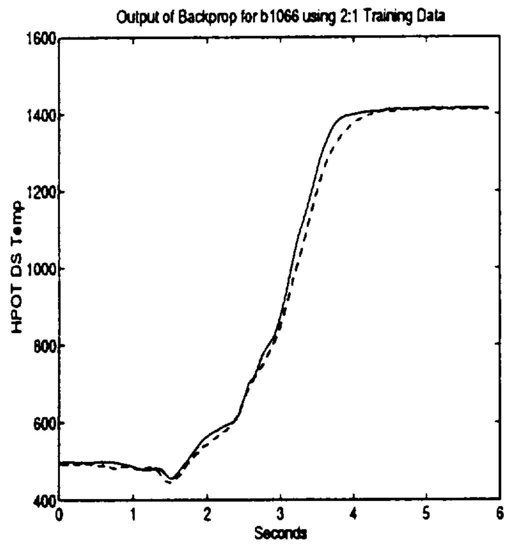

Backprop Error for b1061 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

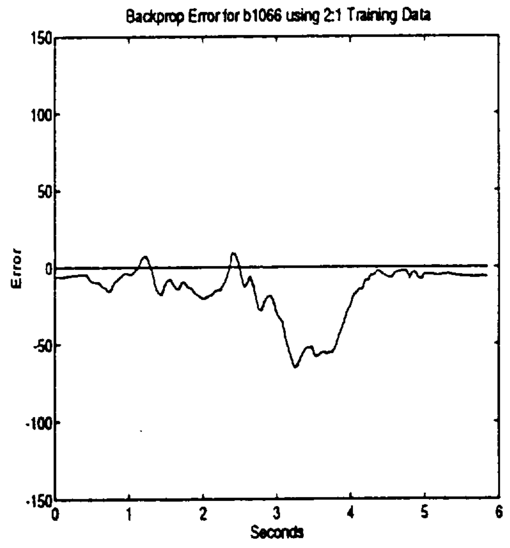

Output of Backprop for b1062 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

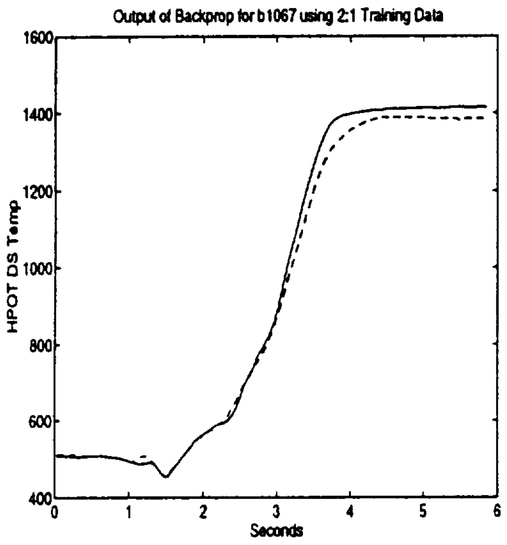

Backprop Error for b1062 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

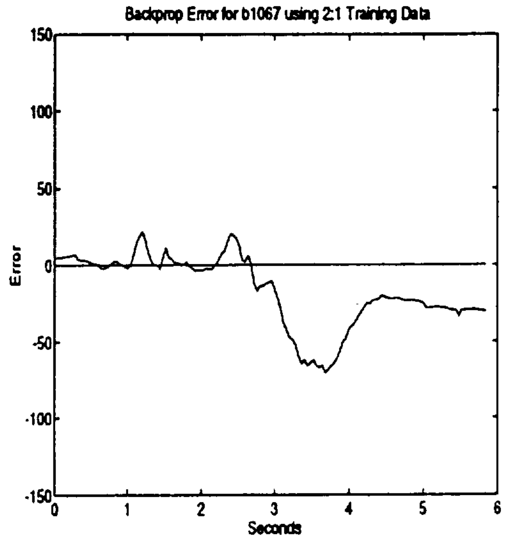

Output of Backprop for b1063 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

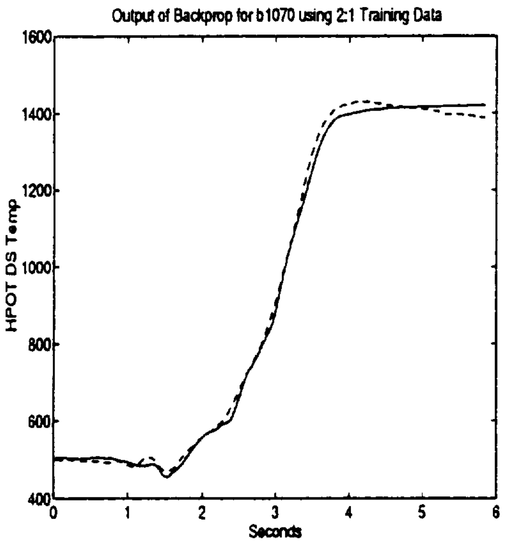

Backprop Error for b1063 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

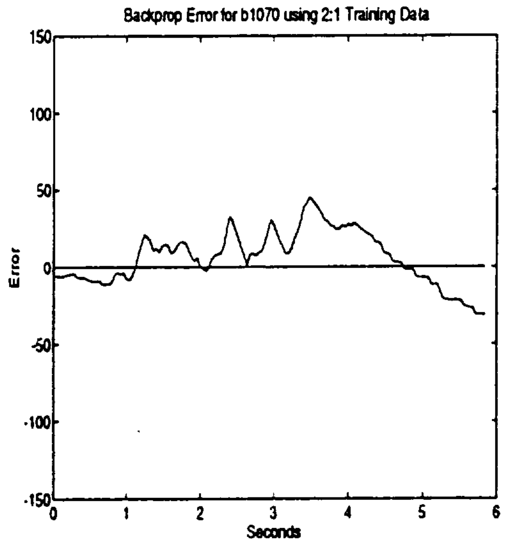

Output of Backprop for b1066 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

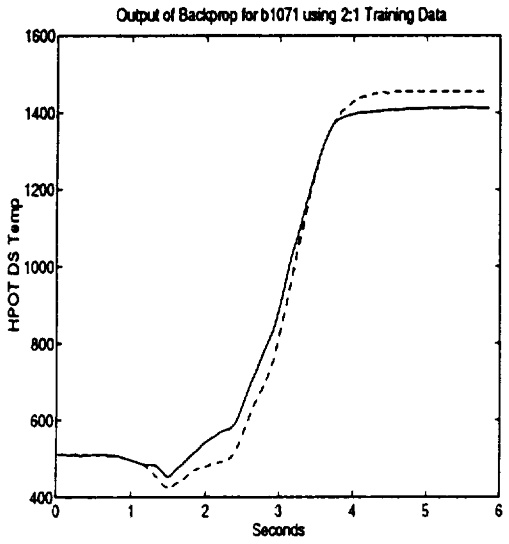

Backprop Error for b1066 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

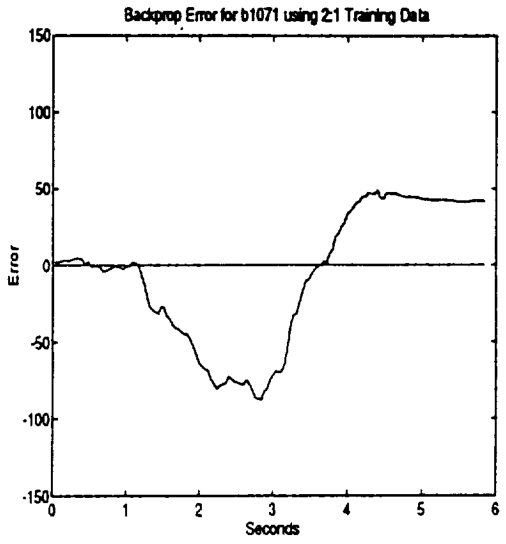

Output of Backprop for b1067 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

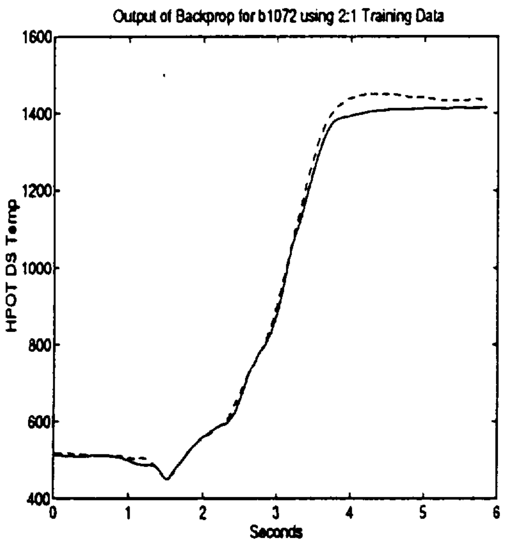

Backprop Error for b1067 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

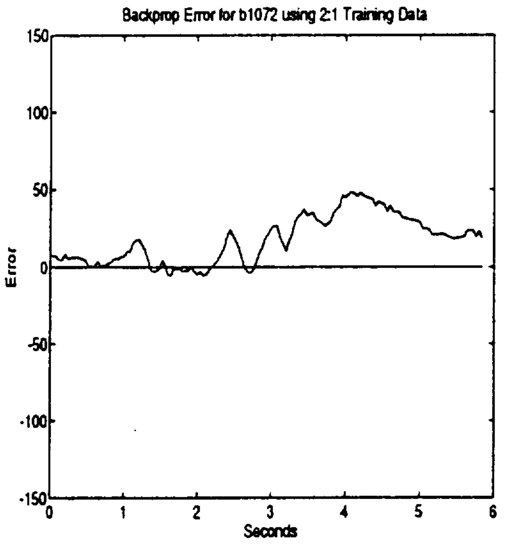

Output of Backprop for b1070 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

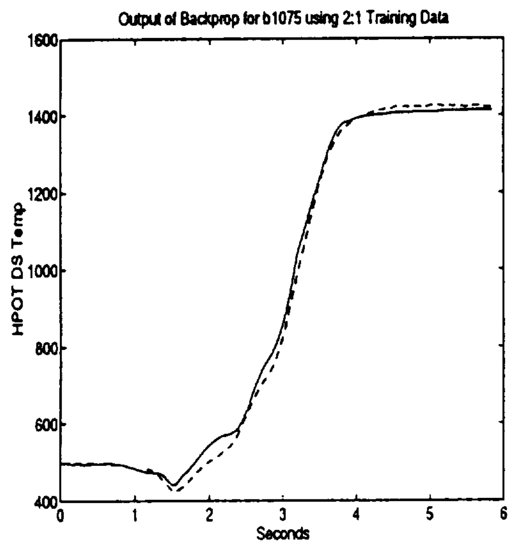

Backprop Error for b1070 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1071 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

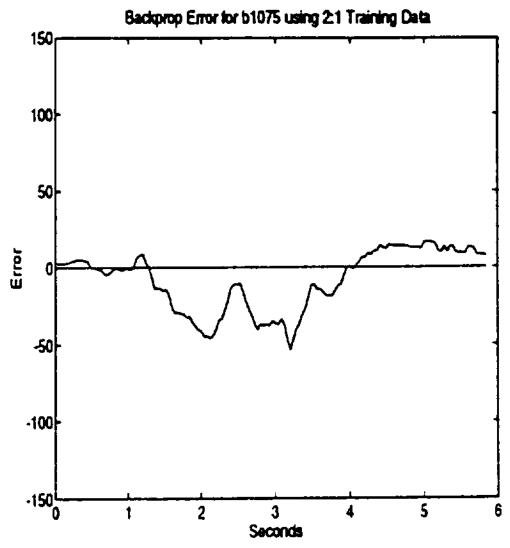

Backprop Error for b1071 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1072 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

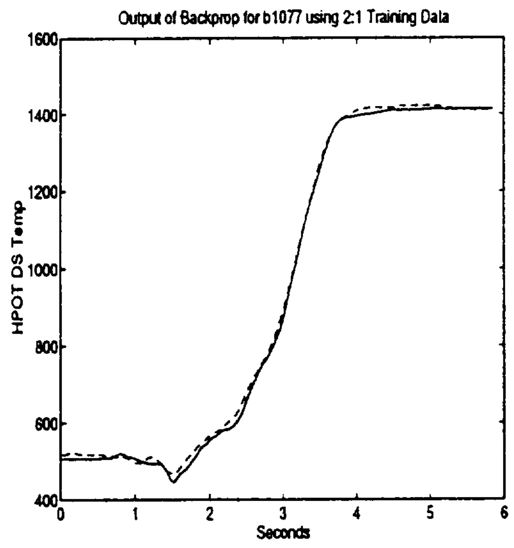

Backprop Error for b1072 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



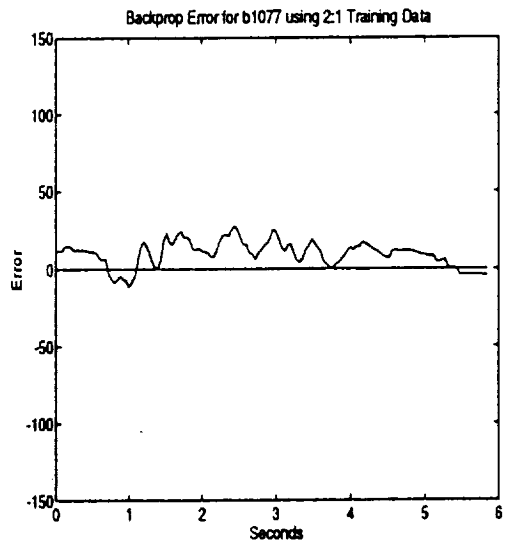
Output of Backprop for b1075 using 2:1 Training Data
HPOT DS Temp
Seconds



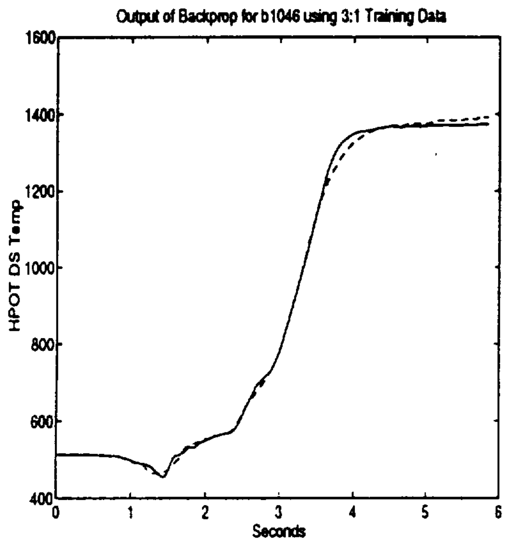
Backprop Error for b1075 using 2:1 Training Data
Error
Seconds

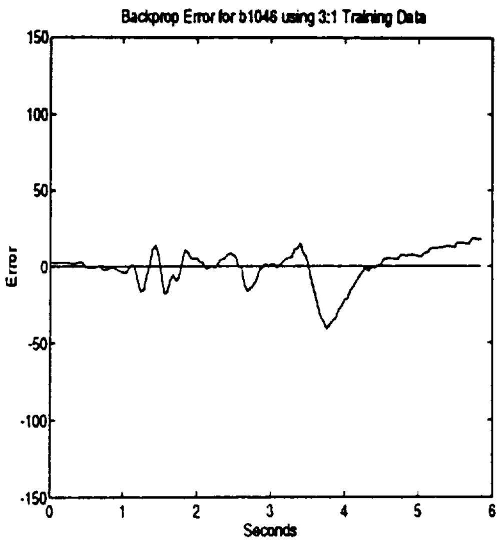


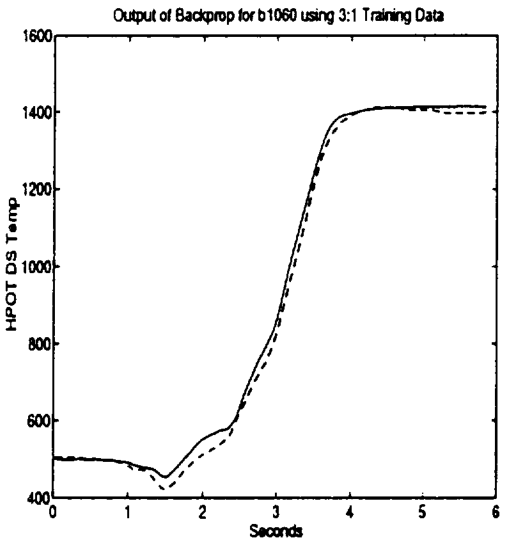
Output of Backprop for b1077 using 2:1 Training Data
HPOT DS Temp
Seconds

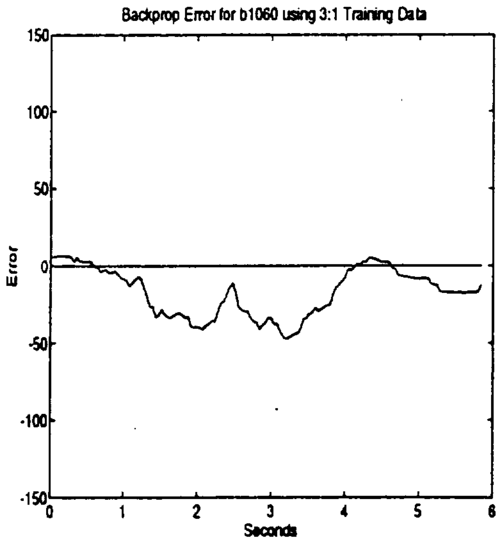


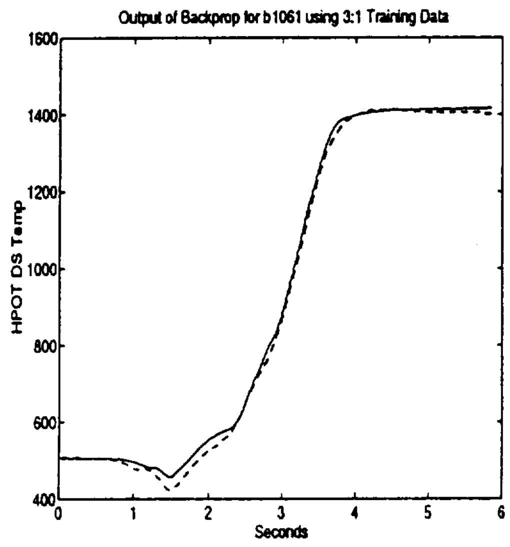
Backprop Error for b1077 using 2:1 Training Data
Error
Seconds


Output of Backprop for b1046 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

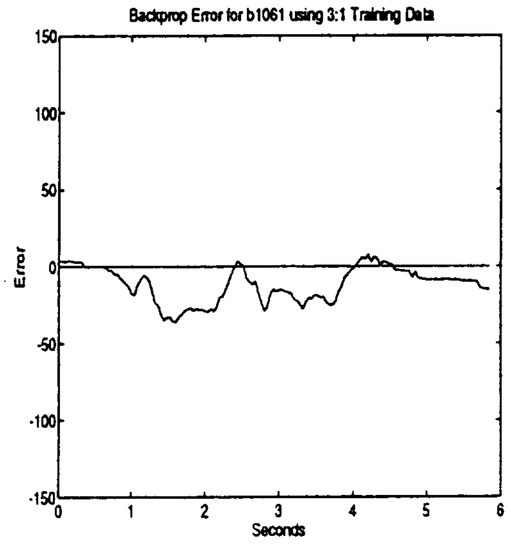

Backprop Error for b1046 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1060 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

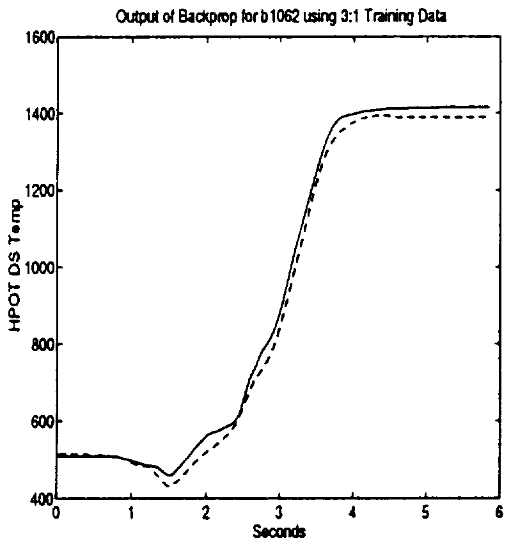

Backprop Error for b1060 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



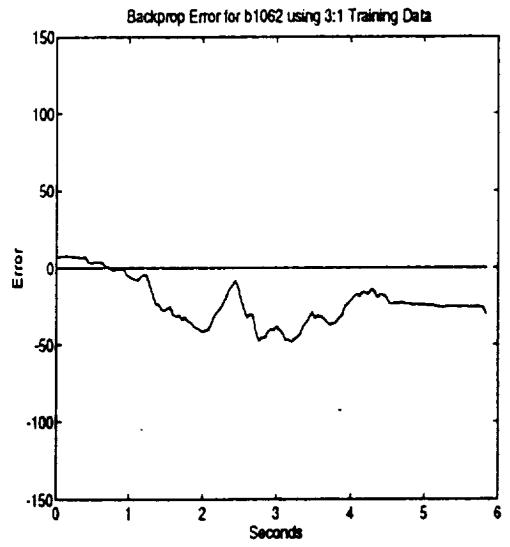
Output of Backprop for b1061 using 3:1 Training Data
HPOT DS Temp
Seconds



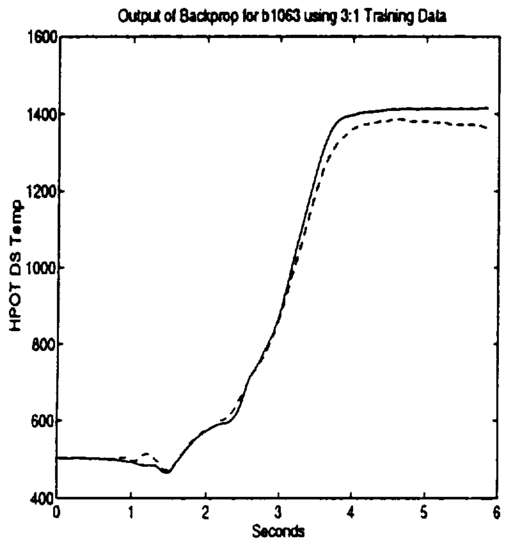
Backprop Error for b1061 using 3:1 Training Data
Error
Seconds

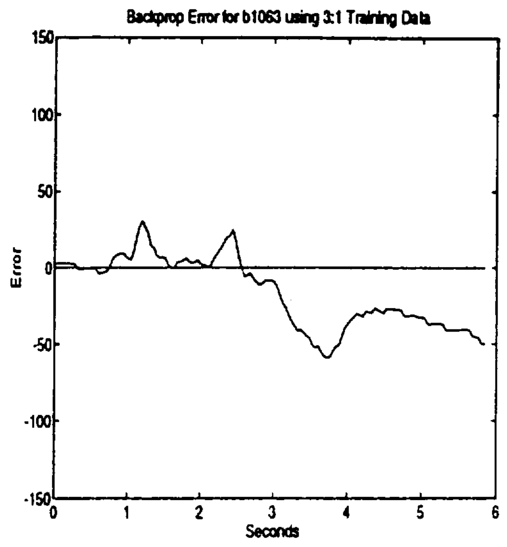


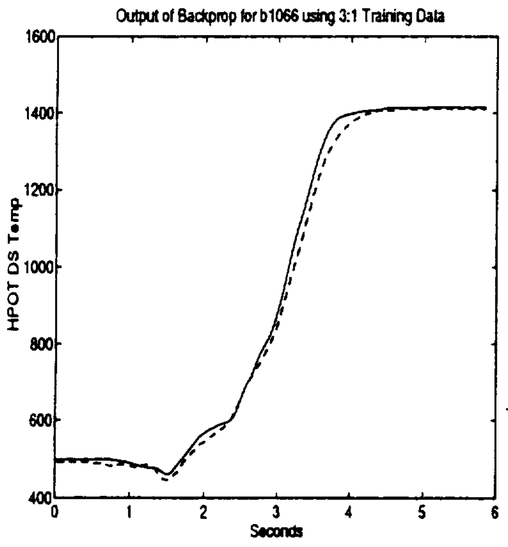
Output of Backprop for b1062 using 3:1 Training Data
HPOT DS Temp
Seconds

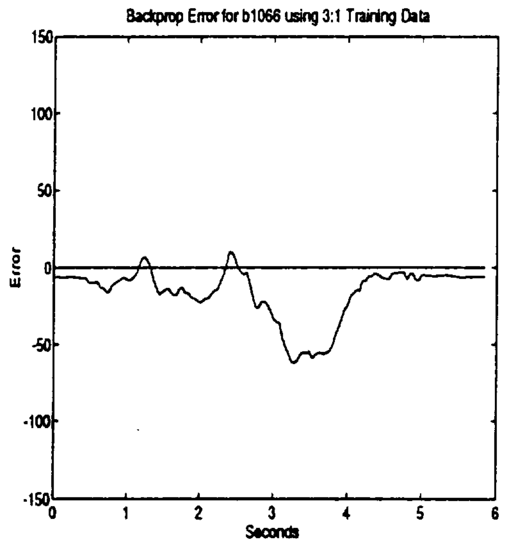


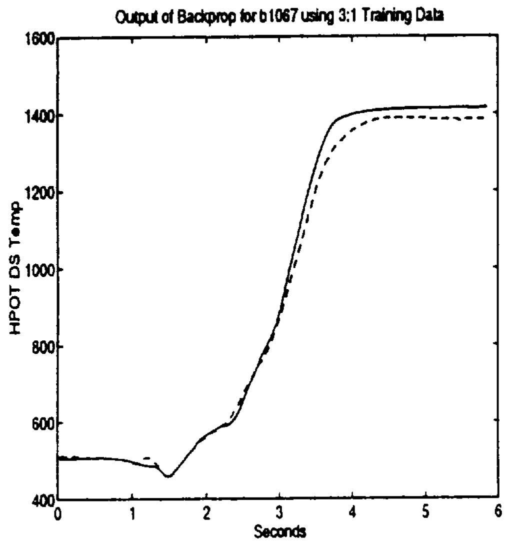
Backprop Error for b1062 using 3:1 Training Data
Error
Seconds

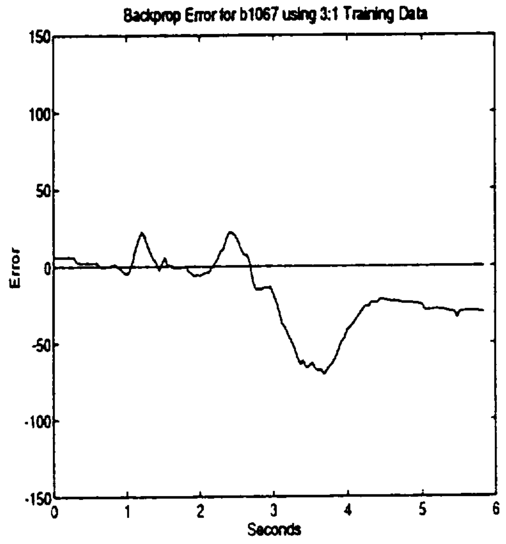

Output of Backprop for b1063 using 3:1 Training Data
HPOT DS Temp
Seconds

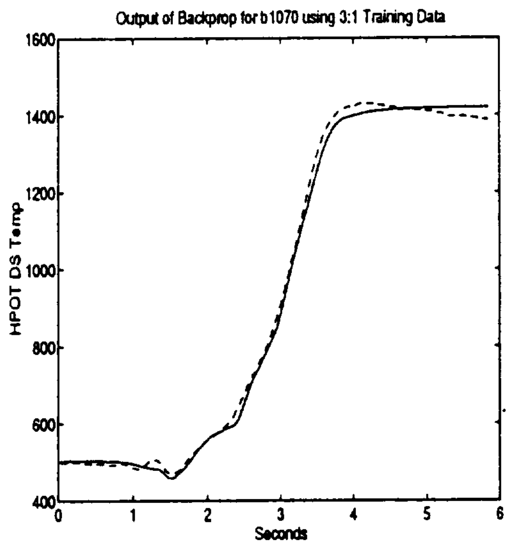

Backprop Error for b1063 using 3:1 Training Data
Error
Seconds

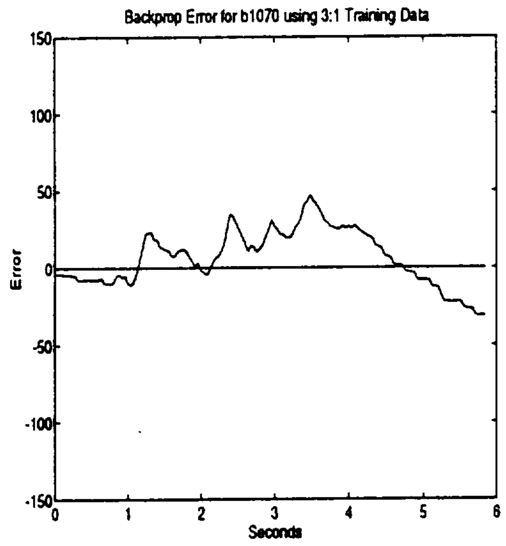

Output of Backprop for b1066 using 3:1 Training Data
HPOT DS Temp
Seconds

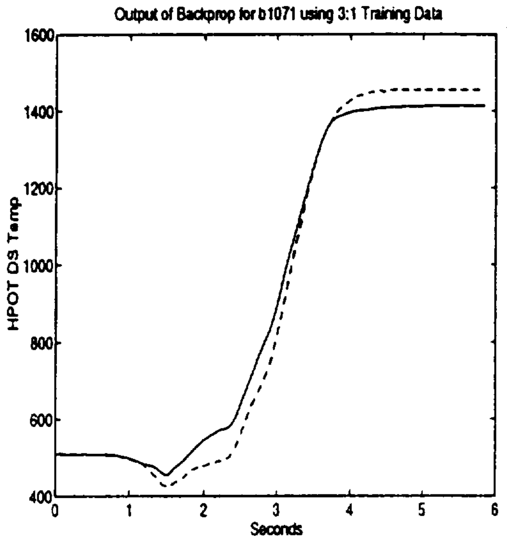

Backprop Error for b1066 using 3:1 Training Data
Error
Seconds

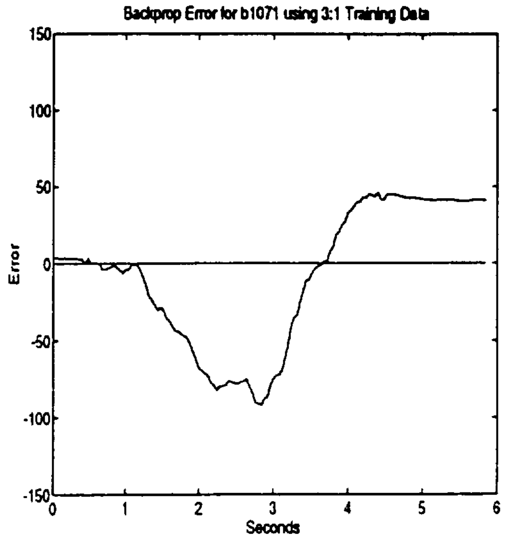

Output of Backprop for b1067 using 3:1 Training Data
HPOT DS Temp
Seconds

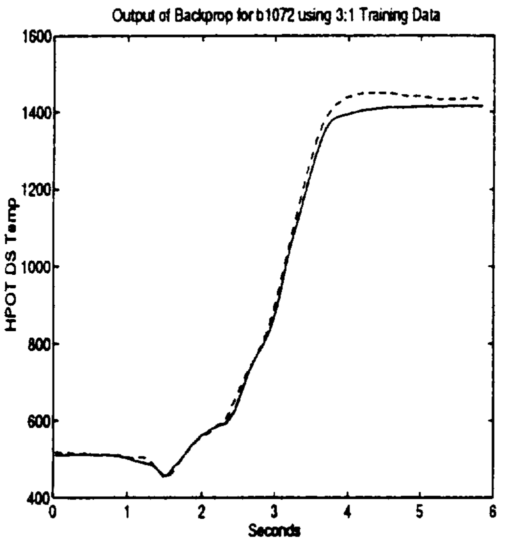

Backprop Error for b1067 using 3:1 Training Data
Error
Seconds

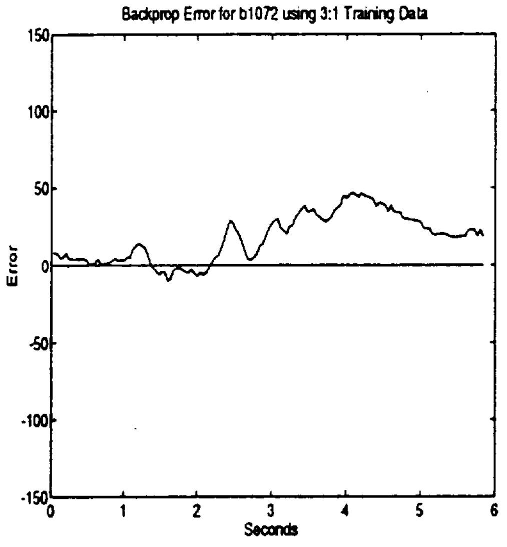

Output of Backprop for b1070 using 3:1 Training Data
HPOT DS Temp
Seconds

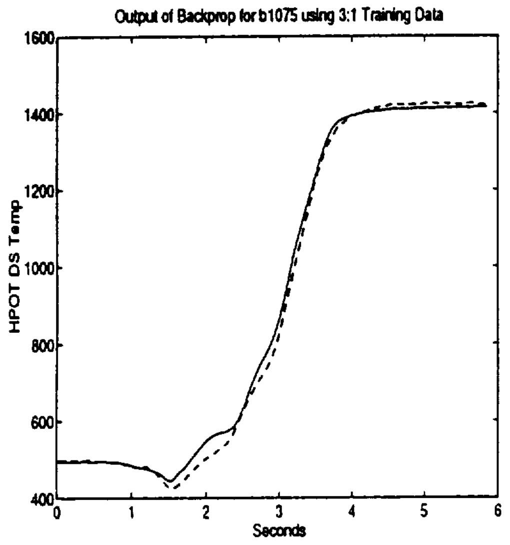

Backprop Error for b1070 using 3:1 Training Data
Error
Seconds

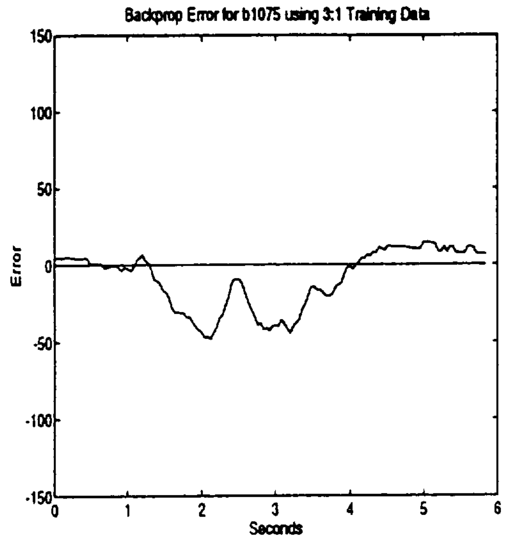

Output of Backprop for b1071 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

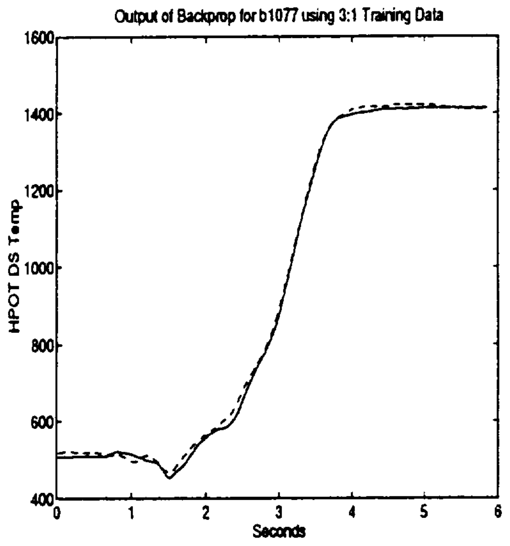

Backprop Error for b1071 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

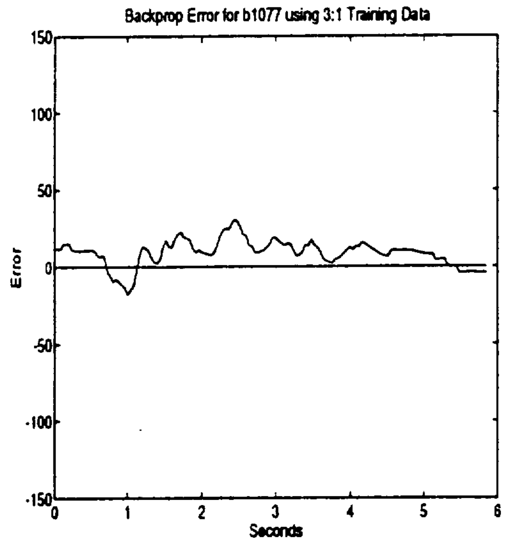

Output of Backprop for b1072 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

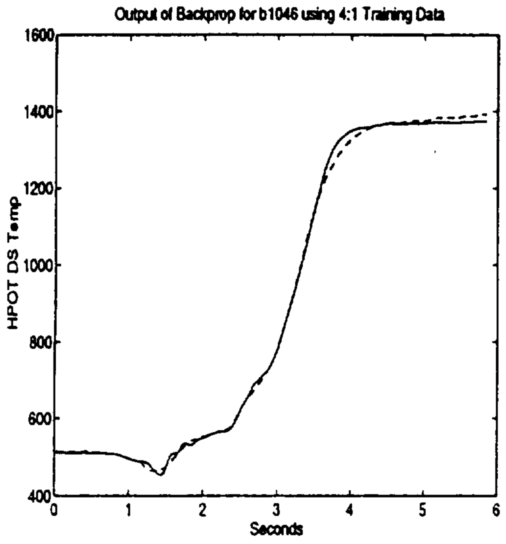

Backprop Error for b1072 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

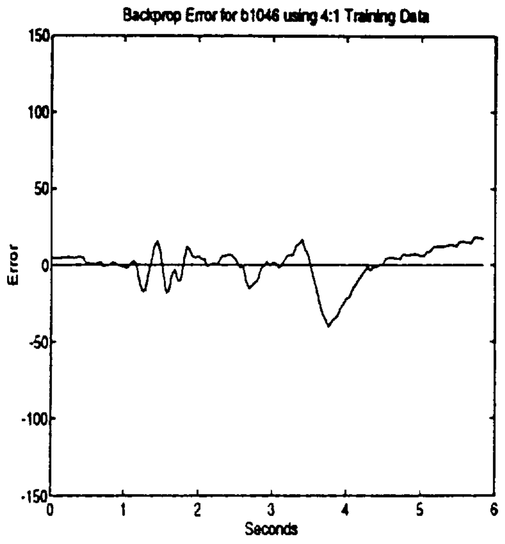

Output of Backprop for b1075 using 3:1 Training Data
HPOT DS Temp
Seconds

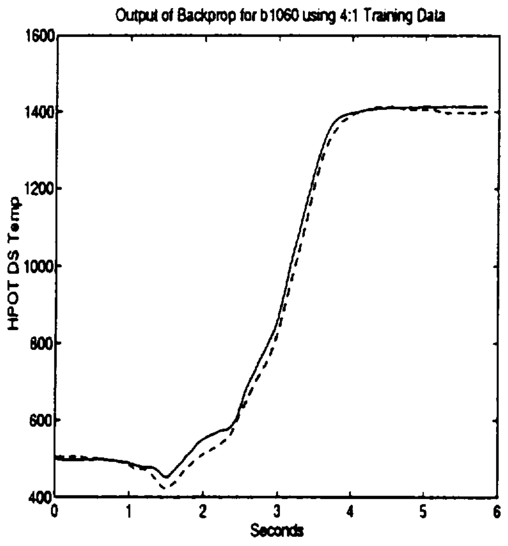

Backprop Error for b1075 using 3:1 Training Data
Error
Seconds

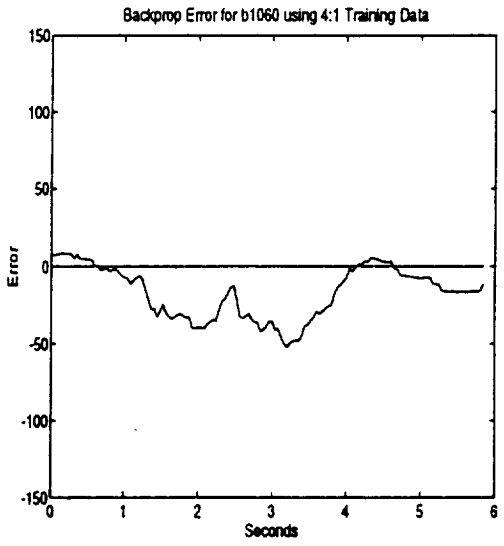

Output of Backprop for b1077 using 3:1 Training Data
HPOT DS Temp
Seconds

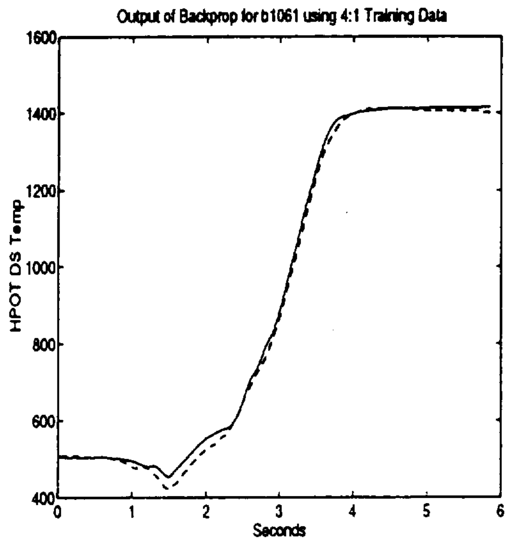

Backprop Error for b1077 using 3:1 Training Data
Error
Seconds


Output of Backprop for b1046 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

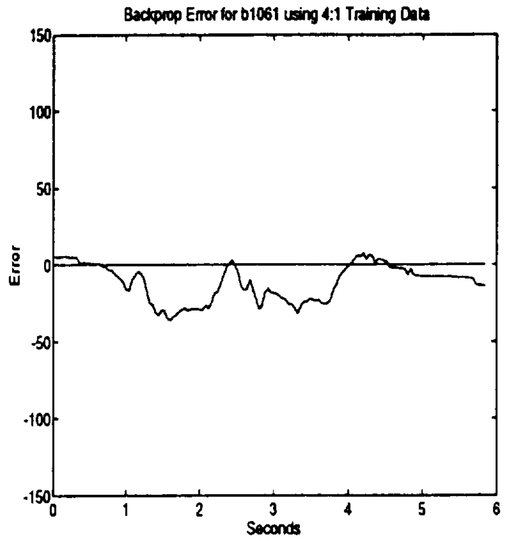

Backprop Error for b1046 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1060 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

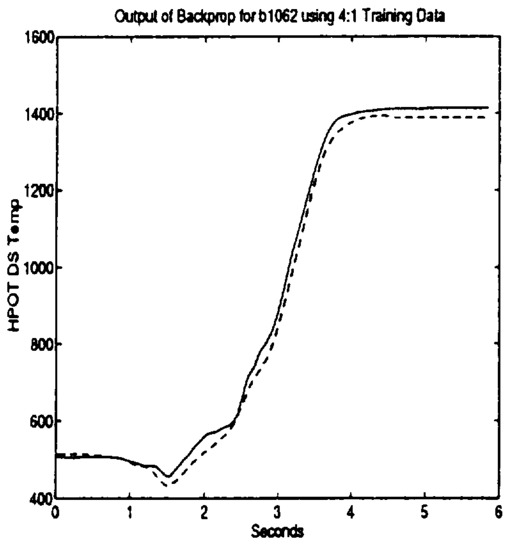

Backprop Error for b1060 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



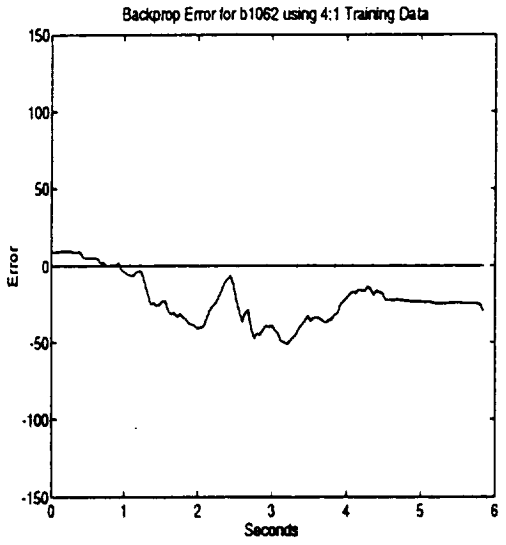
Output of Backprop for b1061 using 4:1 Training Data
HPOT DS Temp
Seconds



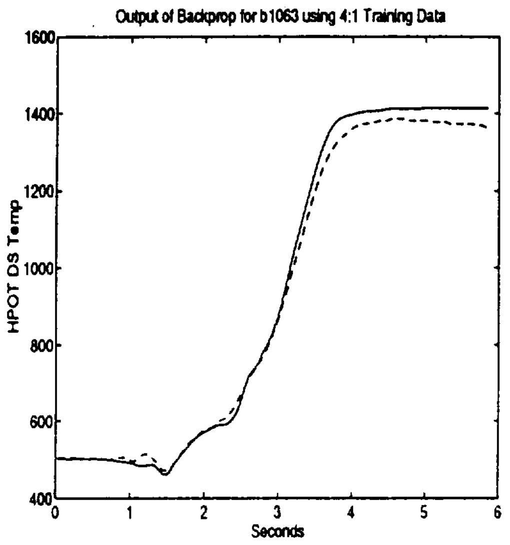
Backprop Error for b1061 using 4:1 Training Data
Error
Seconds

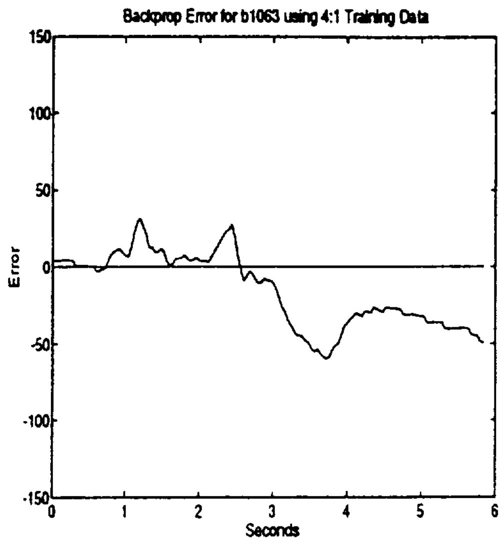


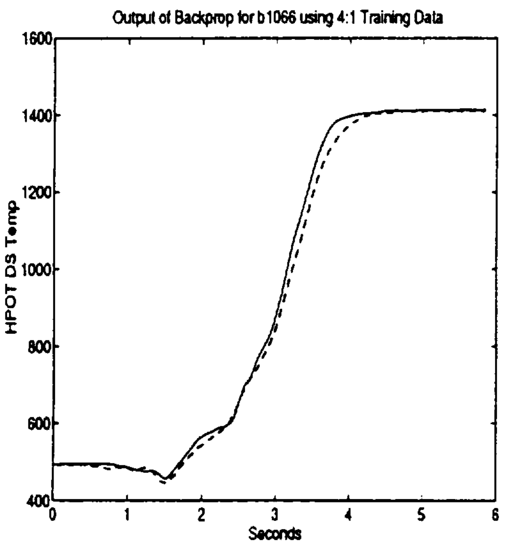
Output of Backprop for b1062 using 4:1 Training Data
HPOT DS Temp
Seconds

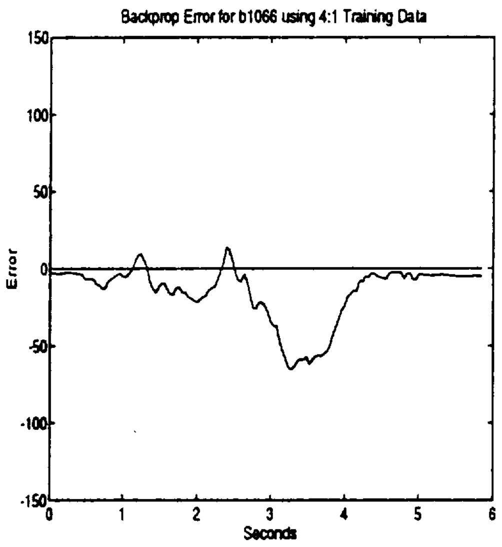


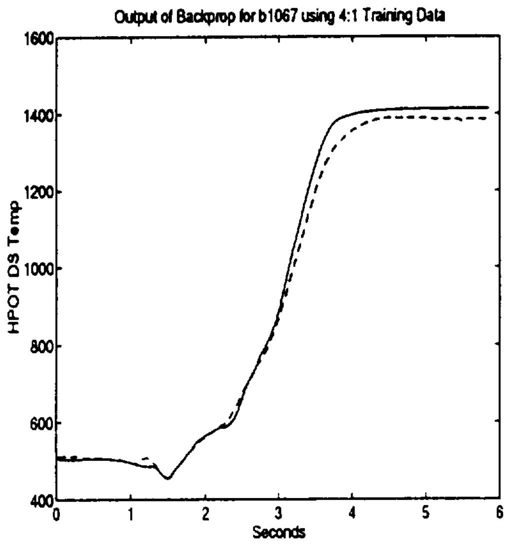
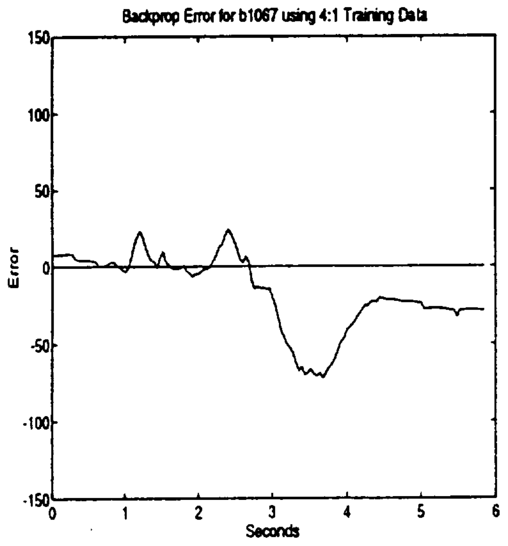
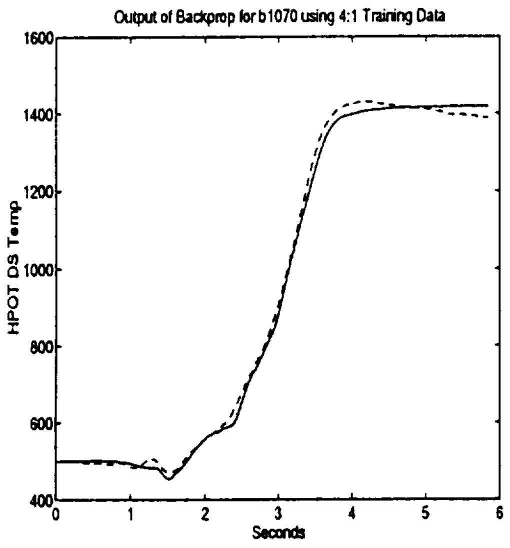
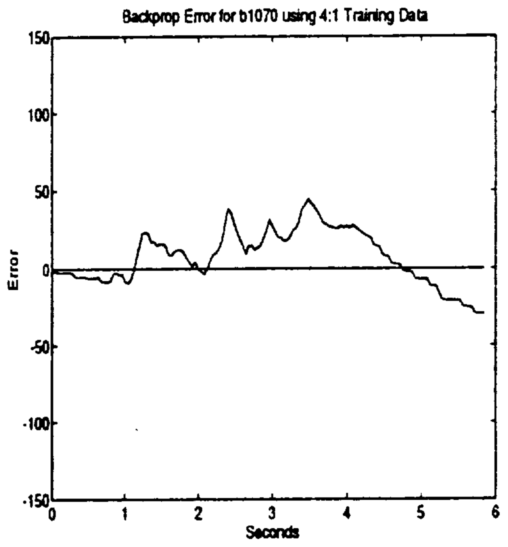
Backprop Error for b1062 using 4:1 Training Data
Error
Seconds

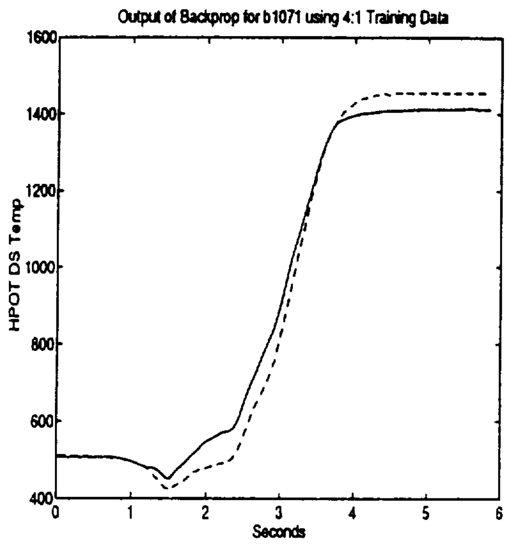

Output of Backprop for b1063 using 4:1 Training Data
HPOT DS Temp
Seconds

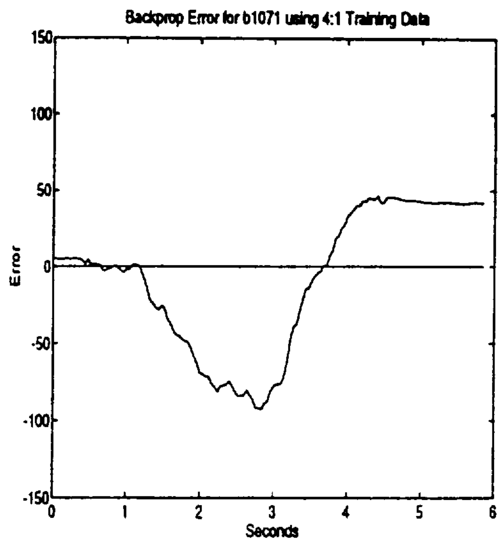

Backprop Error for b1063 using 4:1 Training Data
Error
Seconds

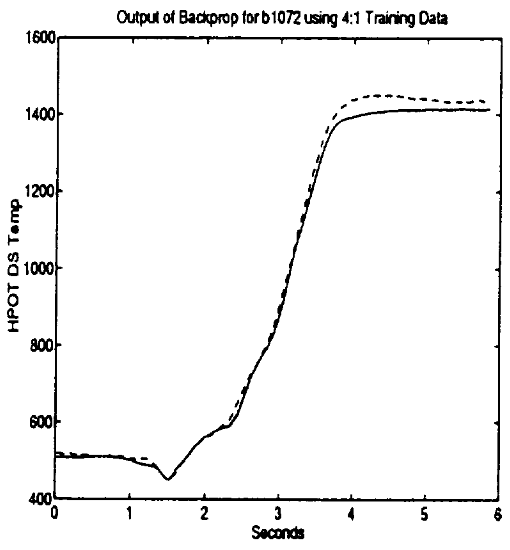

Output of Backprop for b1066 using 4:1 Training Data
HPOT DS Temp
Seconds

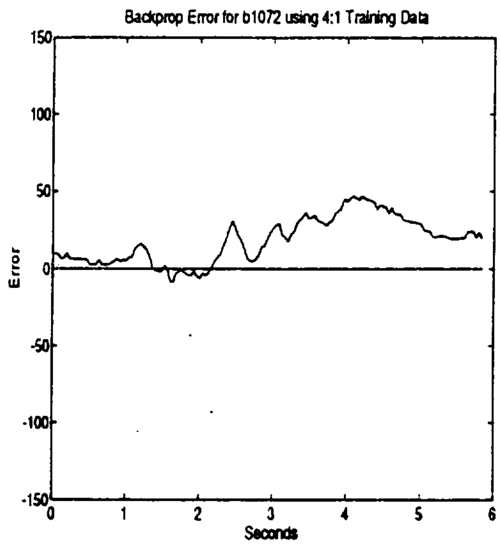

Backprop Error for b1066 using 4:1 Training Data
Error
Seconds

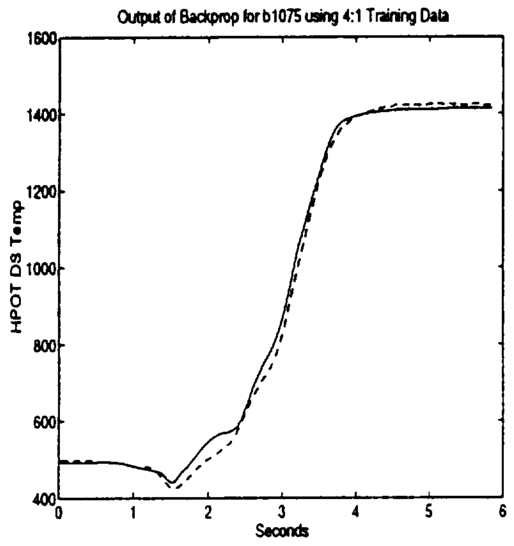

Output of Backprop for b1067 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1067 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

Output of Backprop for b1070 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

Backprop Error for b1070 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

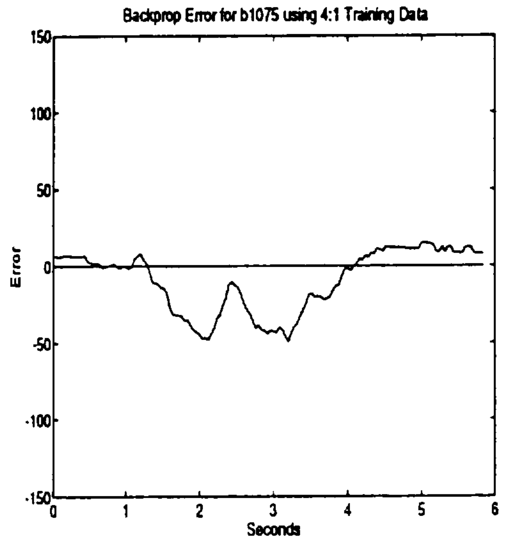

Output of Backprop for b1071 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

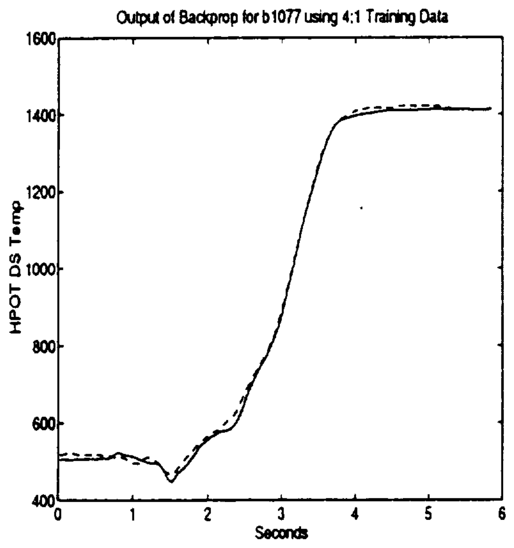

Backprop Error for b1071 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

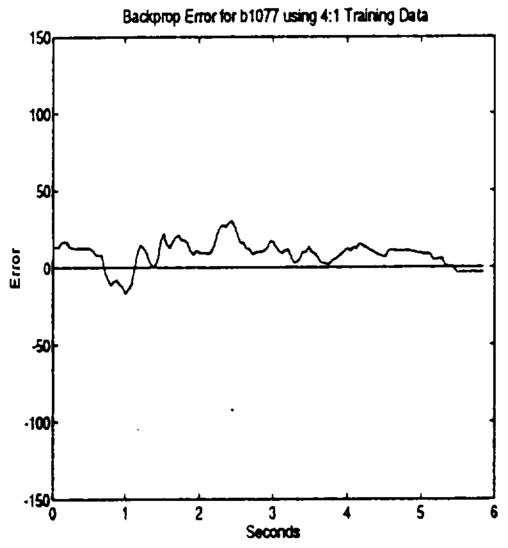

Output of Backprop for b1072 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

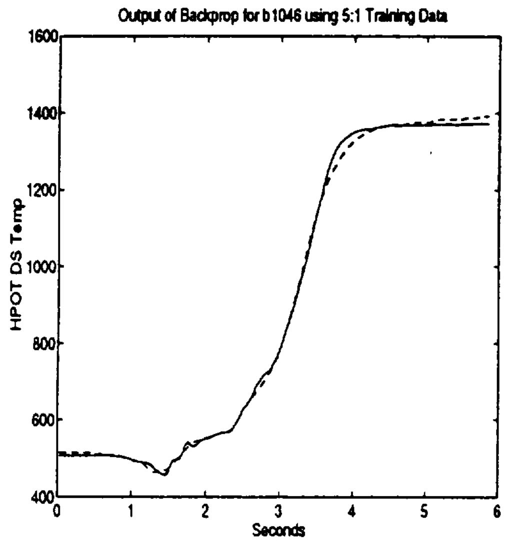

Backprop Error for b1072 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

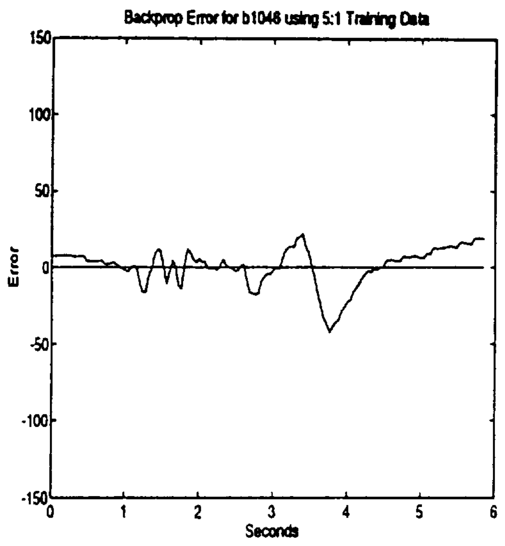

Output of Backprop for b1075 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

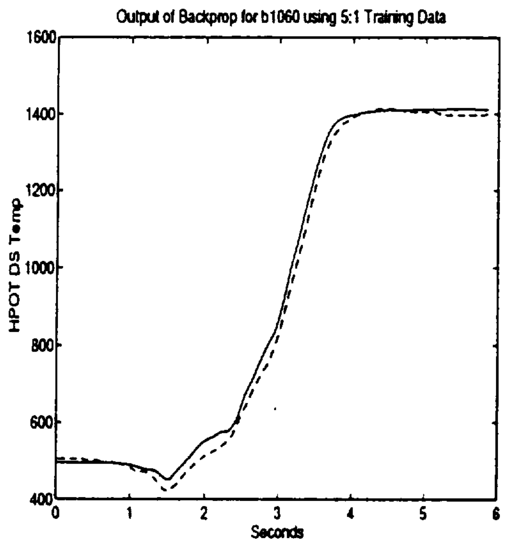

Backprop Error for b1075 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

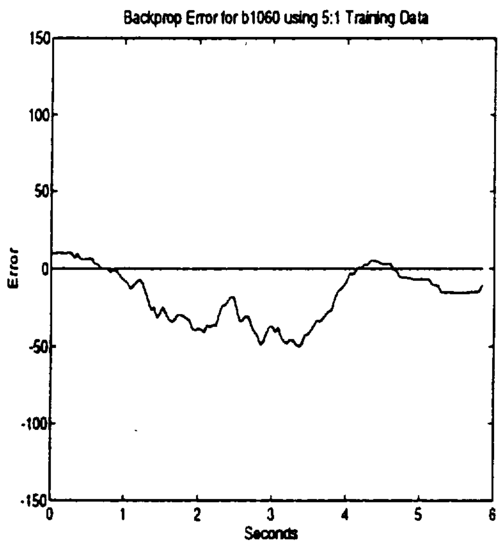

Output of Backprop for b1077 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

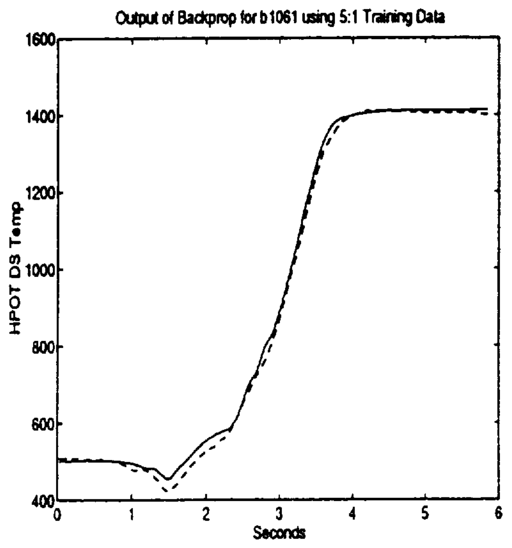

Backprop Error for b1077 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

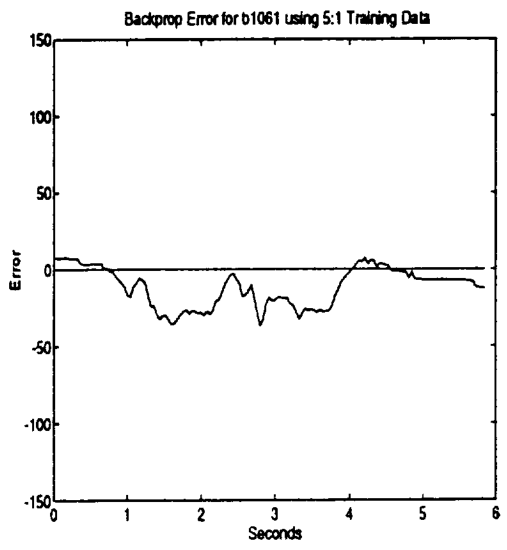

Output of Backprop for b1046 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

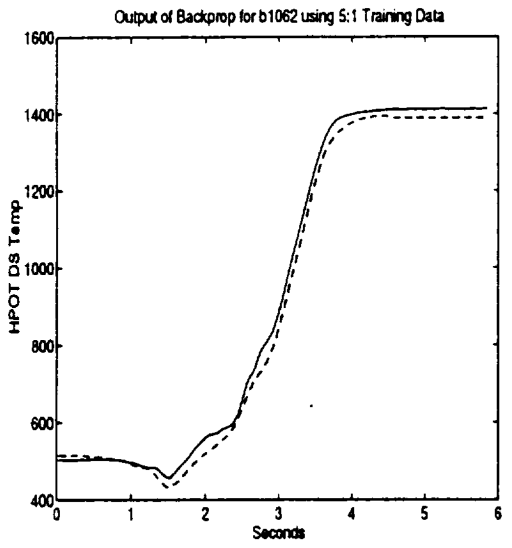

Backprop Error for b1046 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

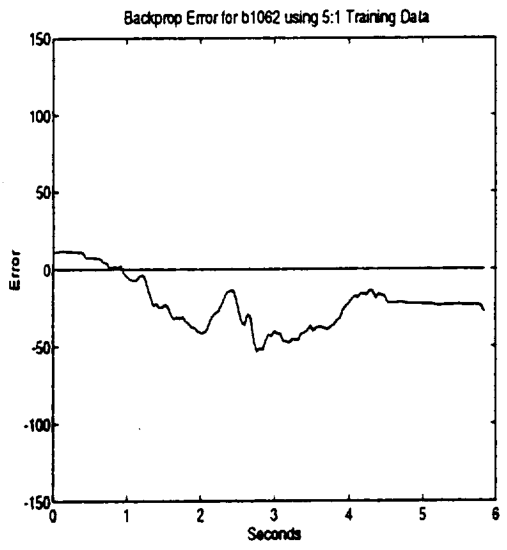

Output of Backprop for b1060 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

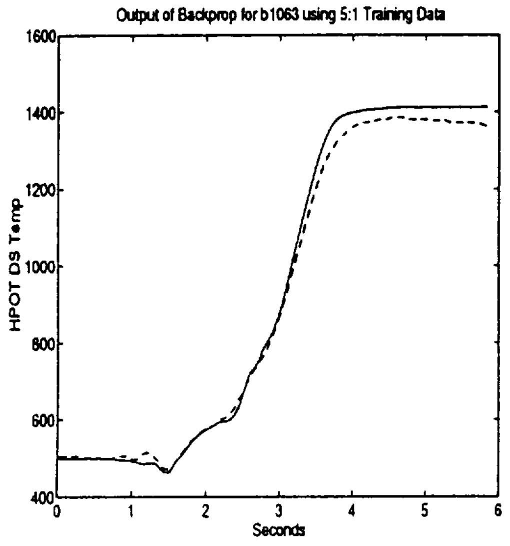

Backprop Error for b1060 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

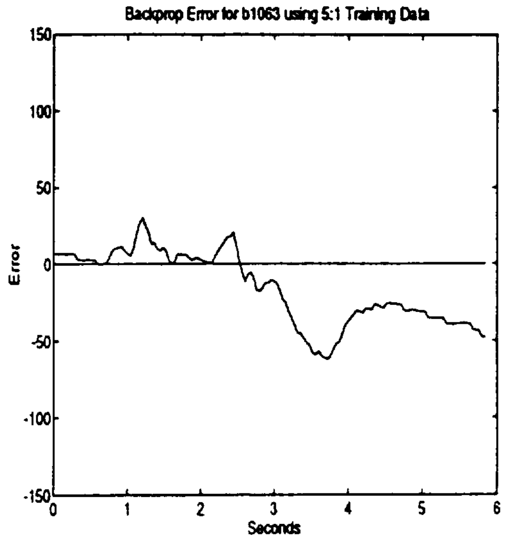

Output of Backprop for b1061 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

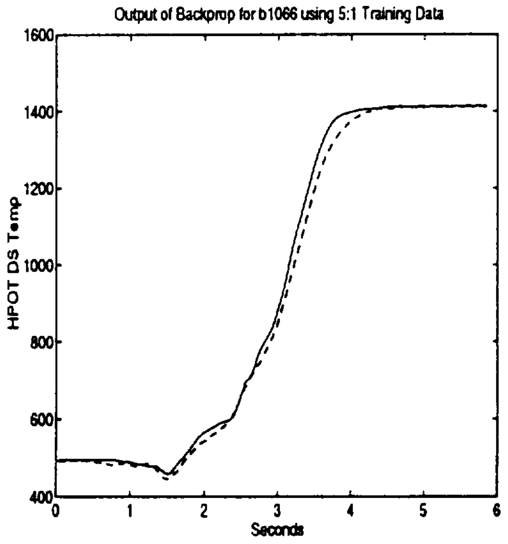

Backprop Error for b1061 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

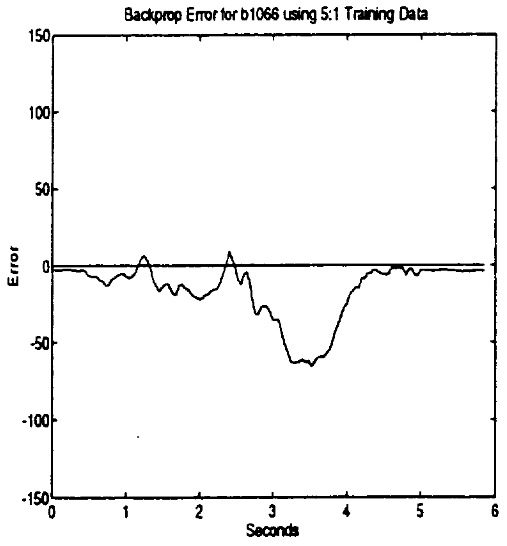

Output of Backprop for b1062 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

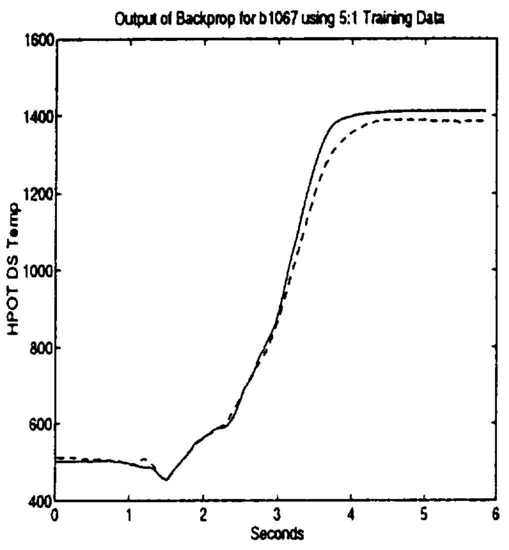

Backprop Error for b1062 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

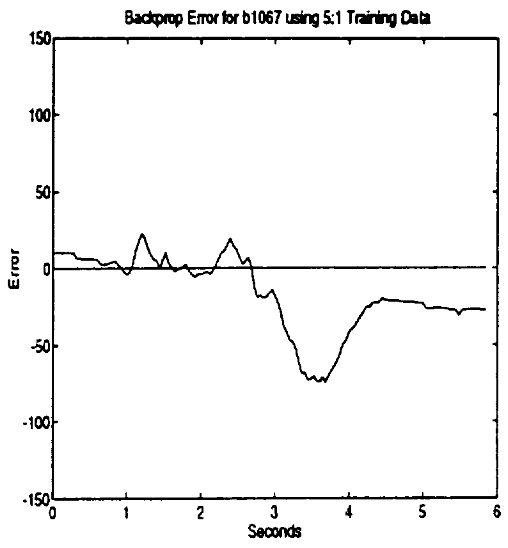

Output of Backprop for b1063 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

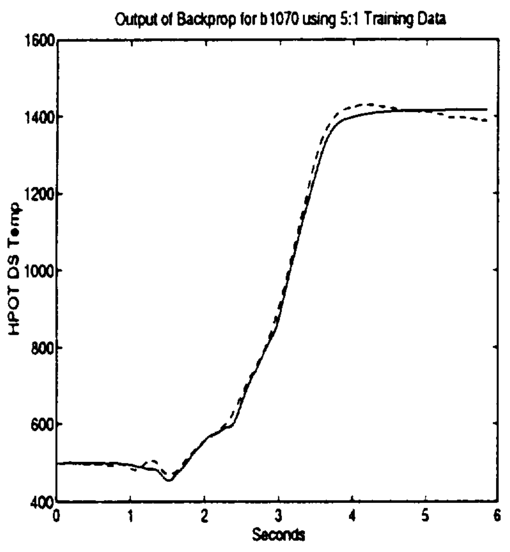

Backprop Error for b1063 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

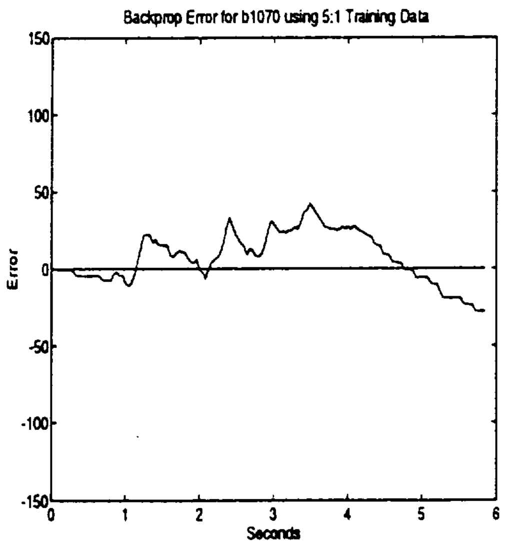

Output of Backprop for b1066 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

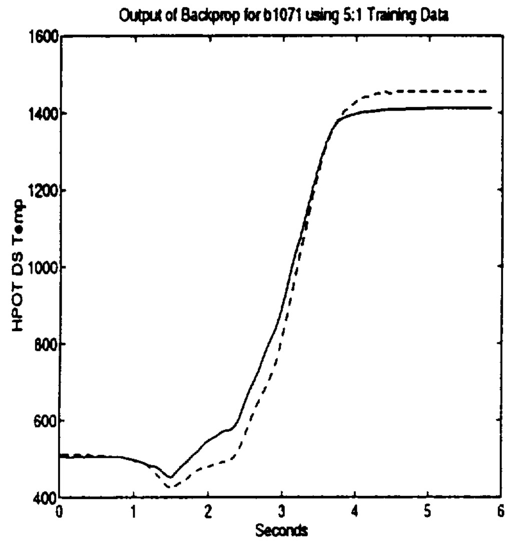

Backprop Error for b1066 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1067 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

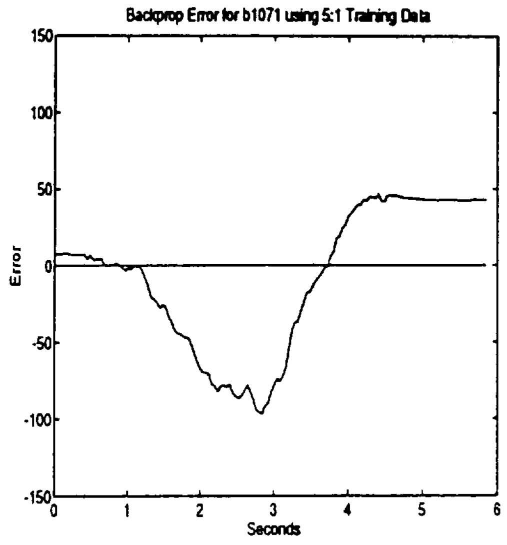

Backprop Error for b1067 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of Backprop for b1070 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

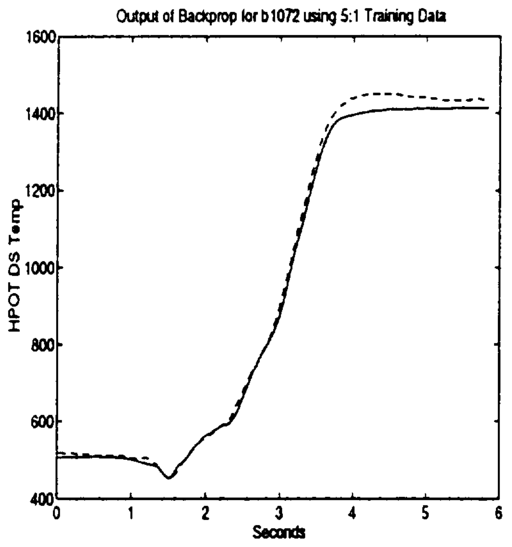

Backprop Error for b1070 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



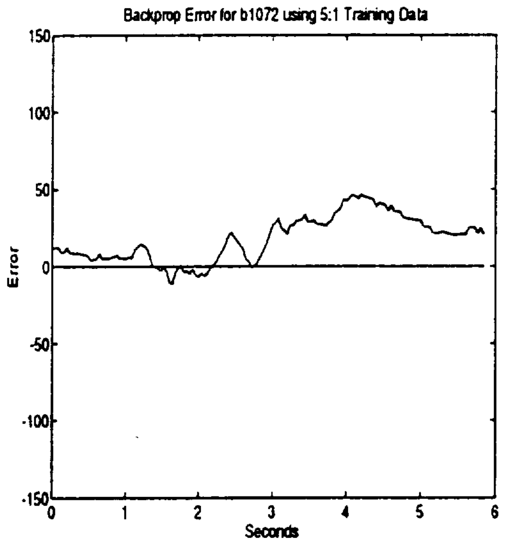
Output of Backprop for b1071 using 5:1 Training Data
HPOT DS Temp
Seconds



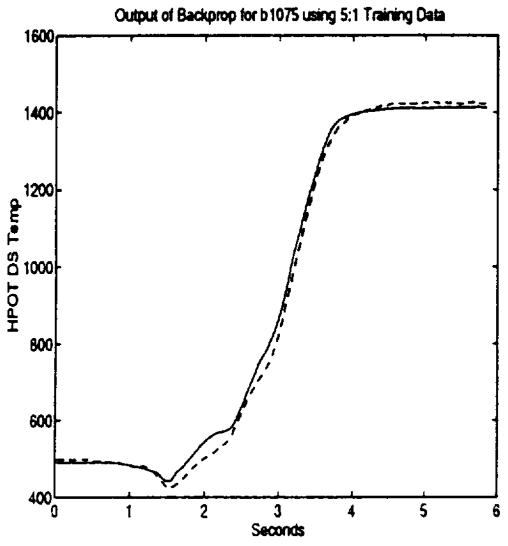
Backprop Error for b1071 using 5:1 Training Data
Error
Seconds

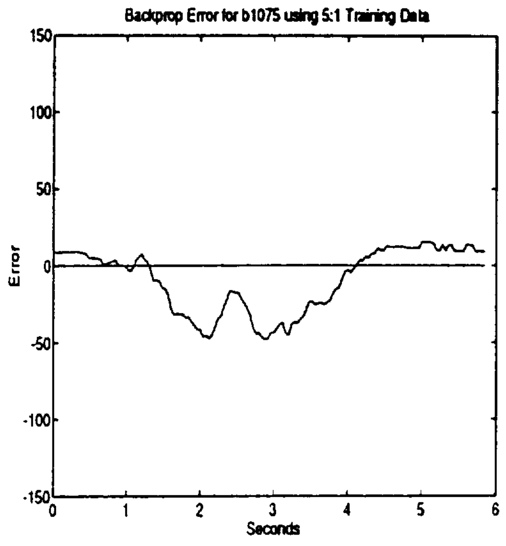


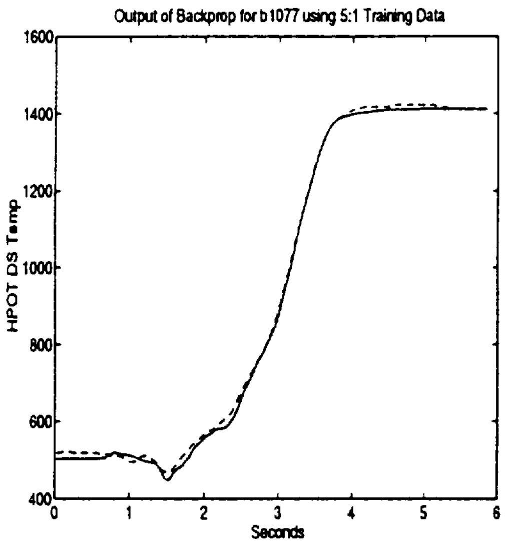
Output of Backprop for b1072 using 5:1 Training Data
HPOT DS Temp
Seconds

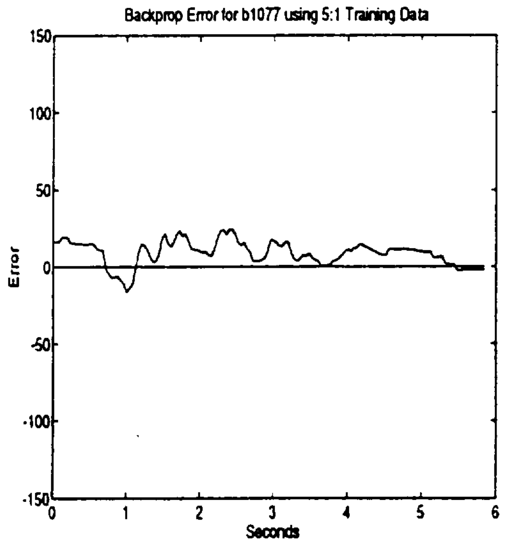


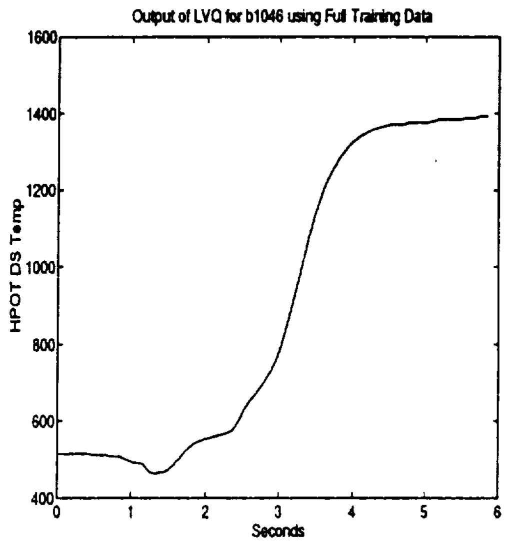
Backprop Error for b1072 using 5:1 Training Data
Error
Seconds

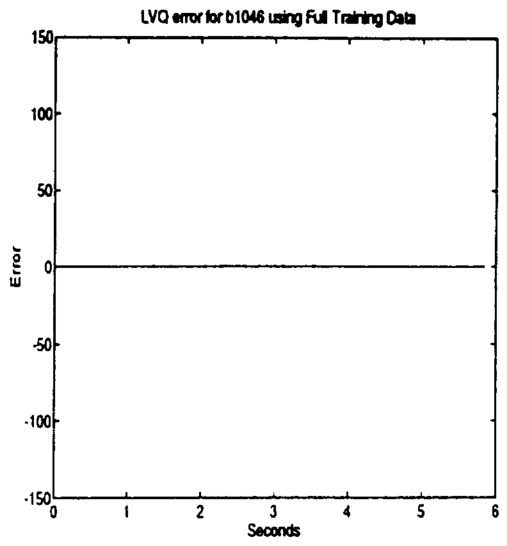

Output of Backprop for b1075 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

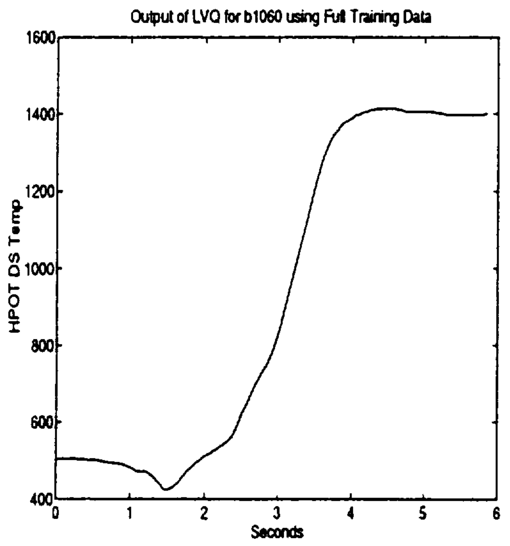

Backprop Error for b1075 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

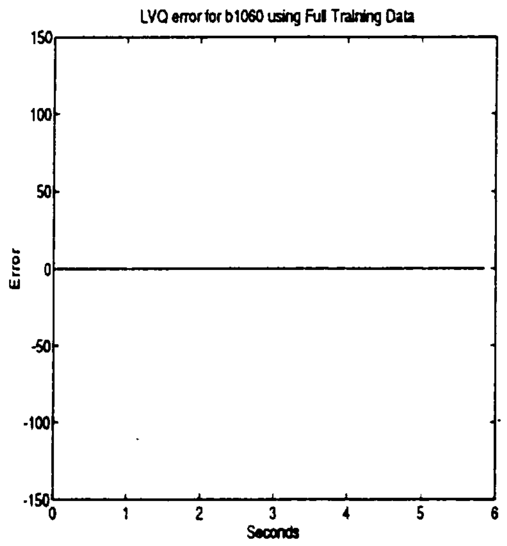

Output of Backprop for b1077 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

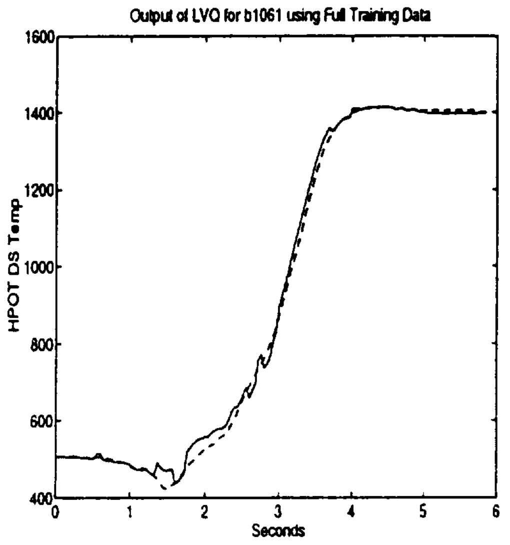

Backprop Error for b1077 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

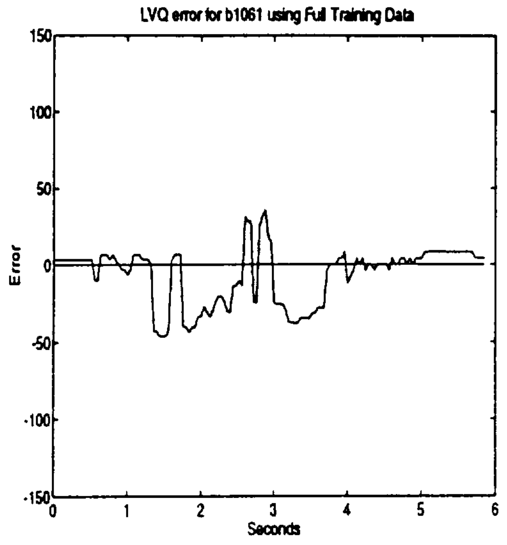

Output of LVQ for b1046 using Full Training Data
HPOT DS Temp
Seconds

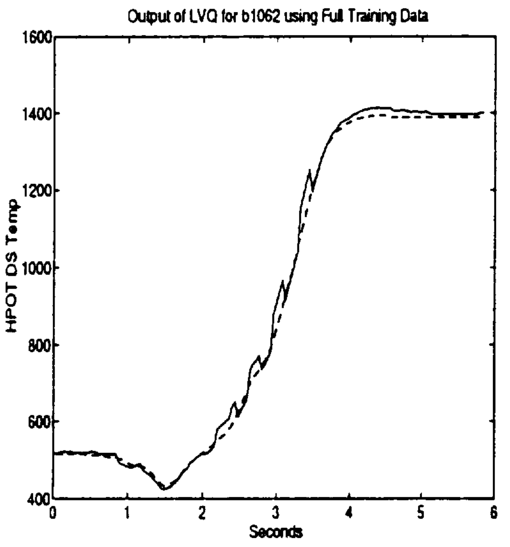

LVQ error for b1046 using Full Training Data
Error
Seconds

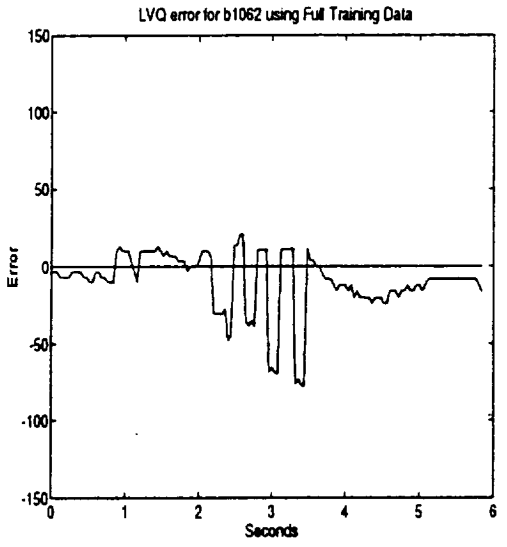

Output of LVQ for b1060 using Full Training Data
HPOT DS Temp
Seconds

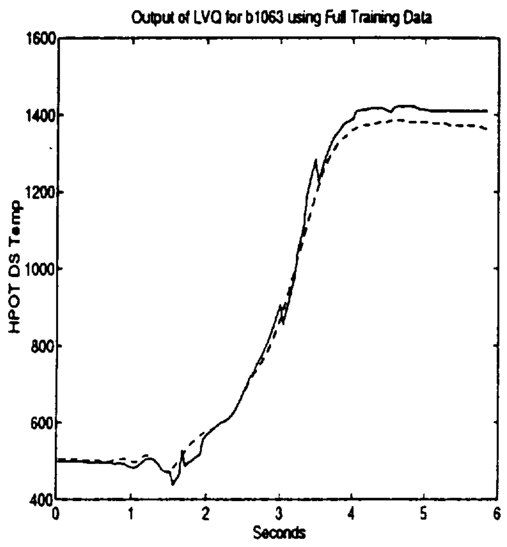

LVQ error for b1060 using Full Training Data
Error
Seconds

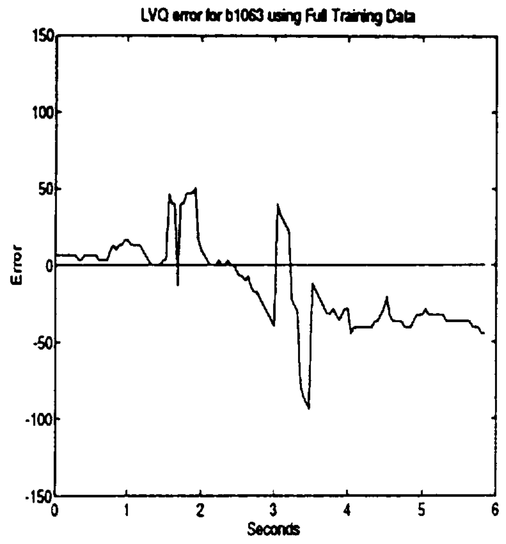

Output of LVQ for b1061 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

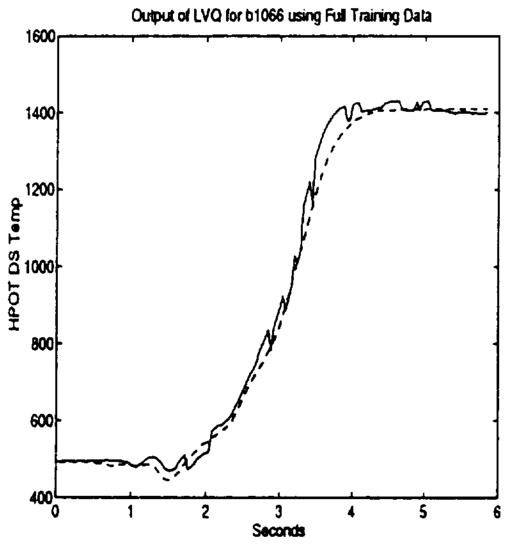

LVQ error for b1061 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

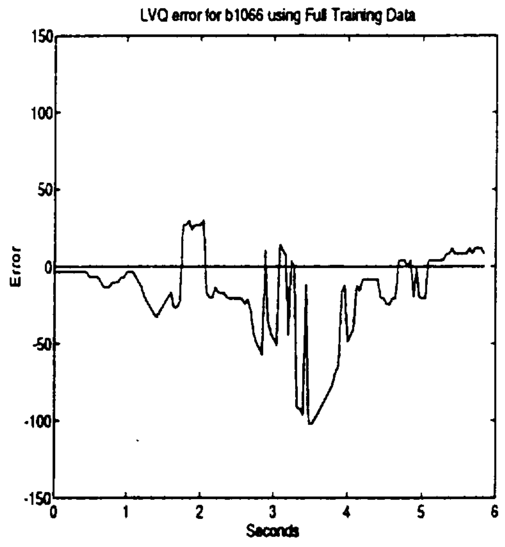

Output of LVQ for b1062 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

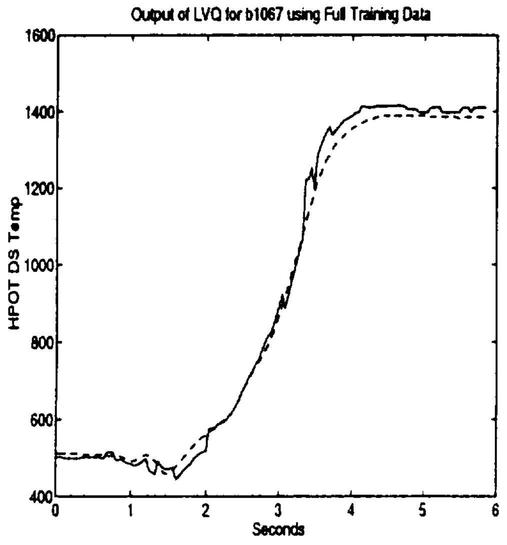

LVQ error for b1062 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

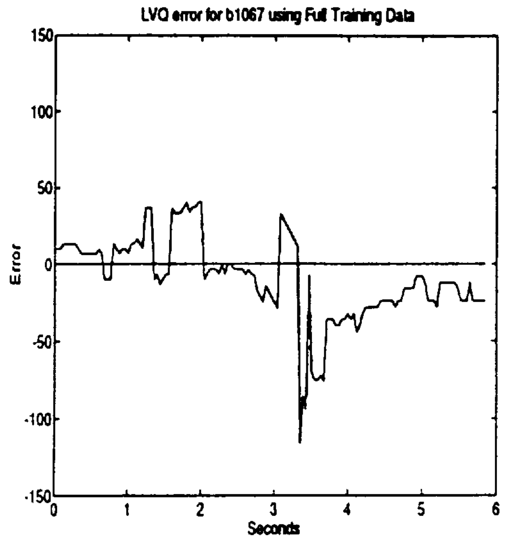

Output of LVQ for b1063 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

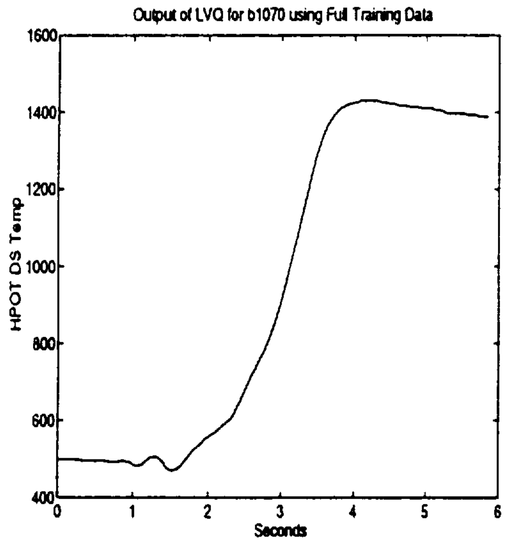

LVQ error for b1063 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

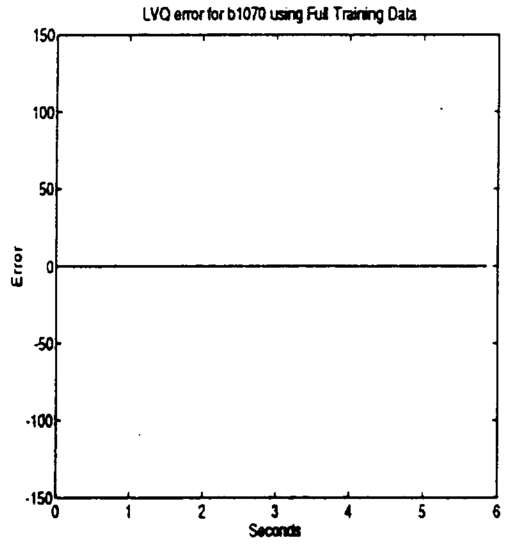

Output of LVQ for b1066 using Full Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

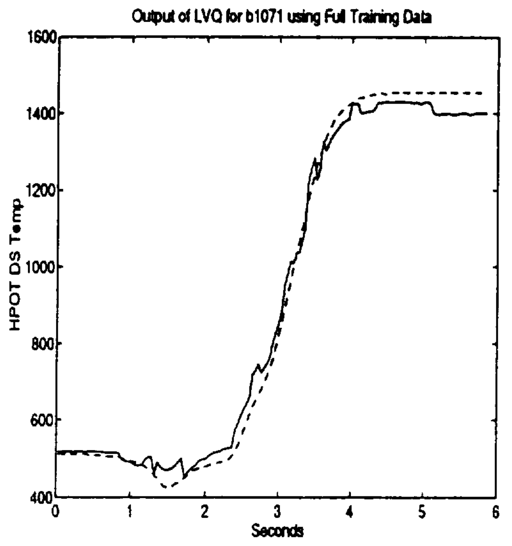

LVQ error for b1066 using Full Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

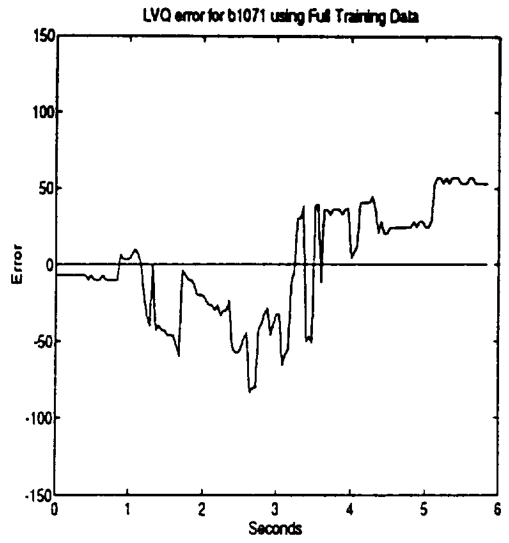

Output of LVQ for b1067 using Full Training Data
HPOT DS Temp
Seconds

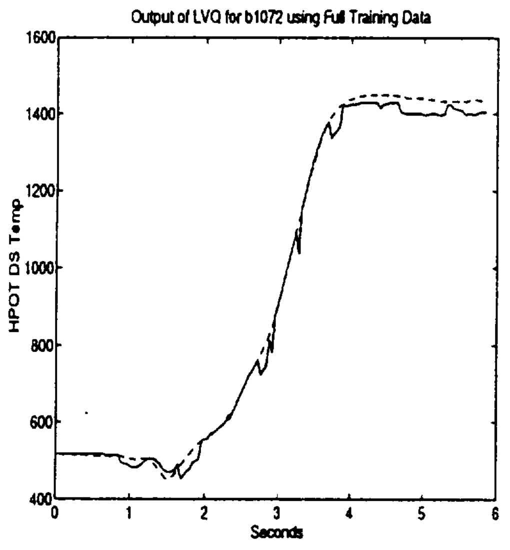

LVQ error for b1067 using Full Training Data
Error
Seconds

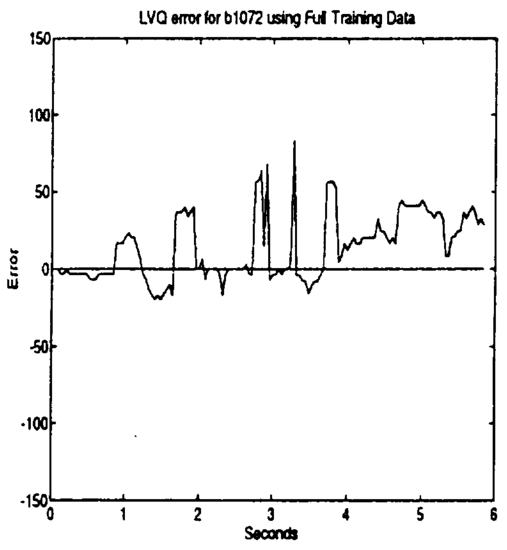

Output of LVQ for b1070 using Full Training Data
HPOT DS Temp
Seconds

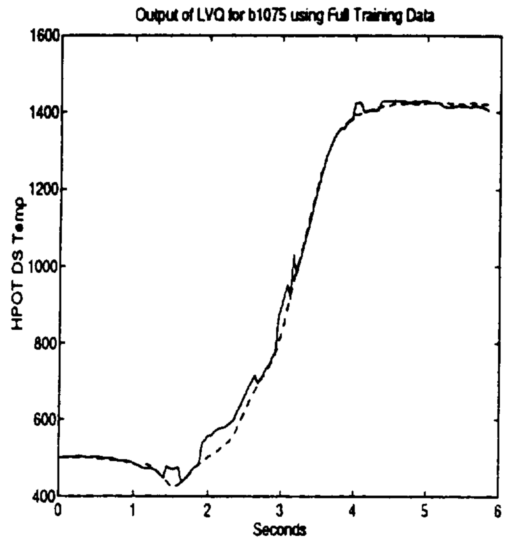

LVQ error for b1070 using Full Training Data
Error
Seconds


Output of LVQ for b1071 using Full Training Data
HPOT DS Temp
Seconds

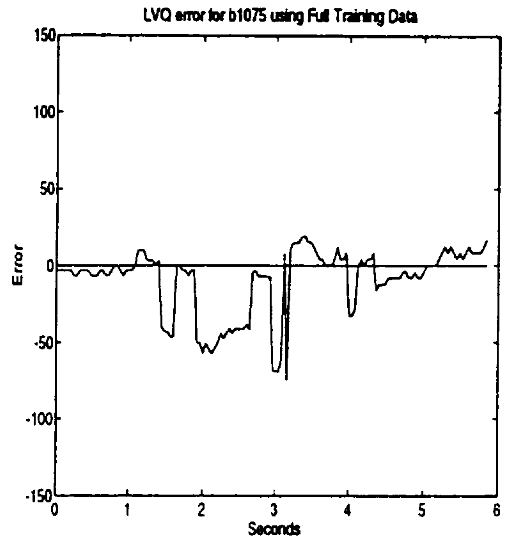

LVQ error for b1071 using Full Training Data
Error
Seconds


Output of LVQ for b1072 using Full Training Data
HPOT DS Temp
Seconds

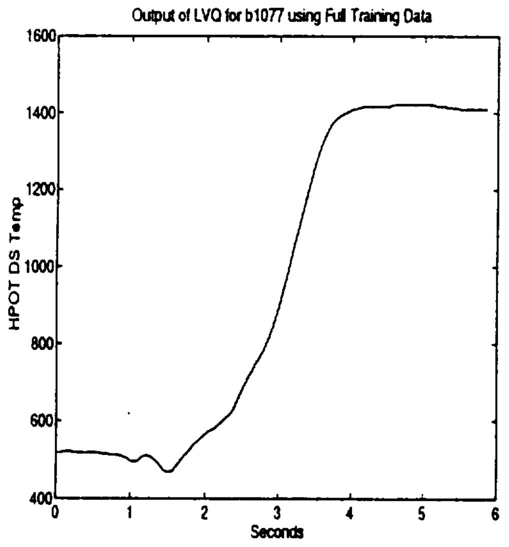

LVQ error for b1072 using Full Training Data
Error
Seconds



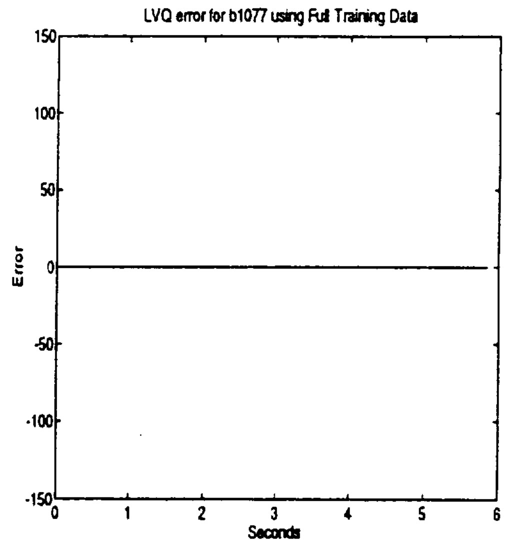
Output of LVQ for b1075 using Full Training Data
HPOT DS Temp
Seconds



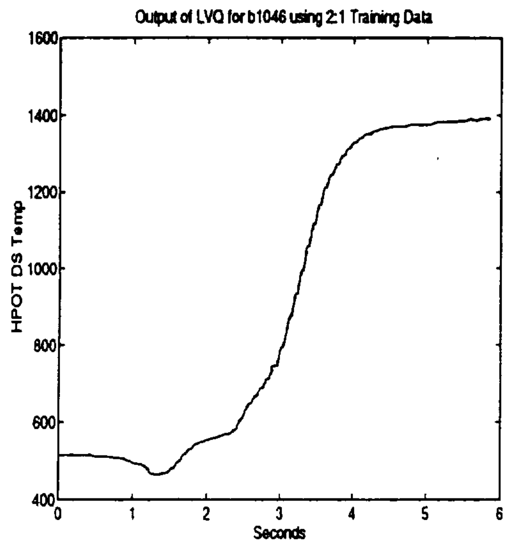
LVQ error for b1075 using Full Training Data
Error
Seconds

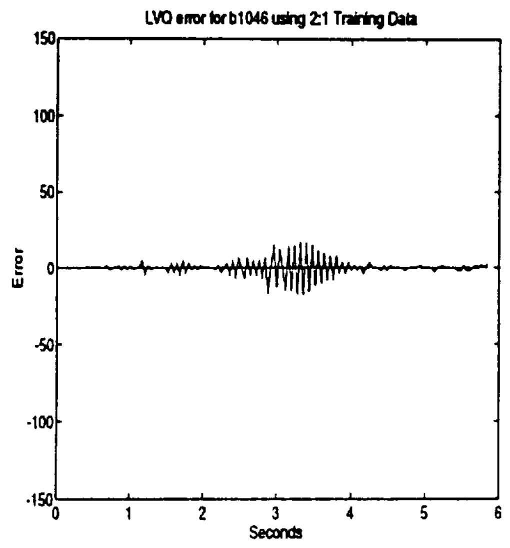


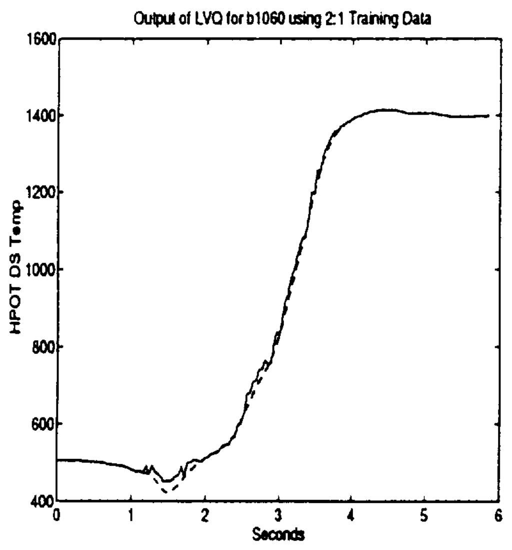
Output of LVQ for b1077 using Full Training Data
HPOT DS Temp
Seconds

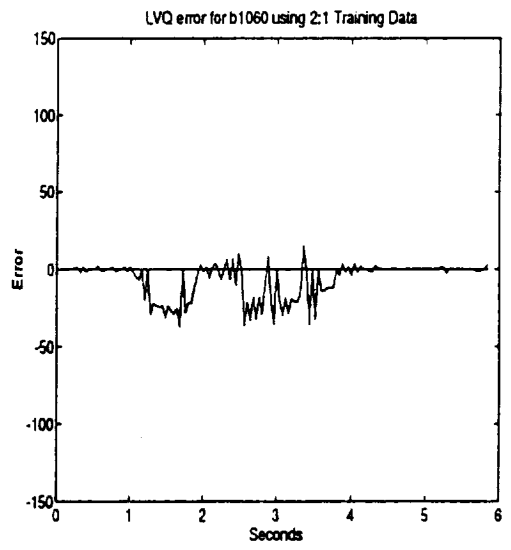


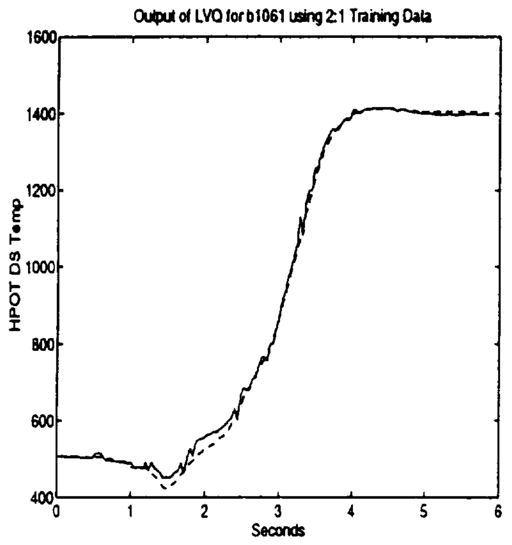
LVQ error for b1077 using Full Training Data
Error
Seconds


Output of LVQ for b1046 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

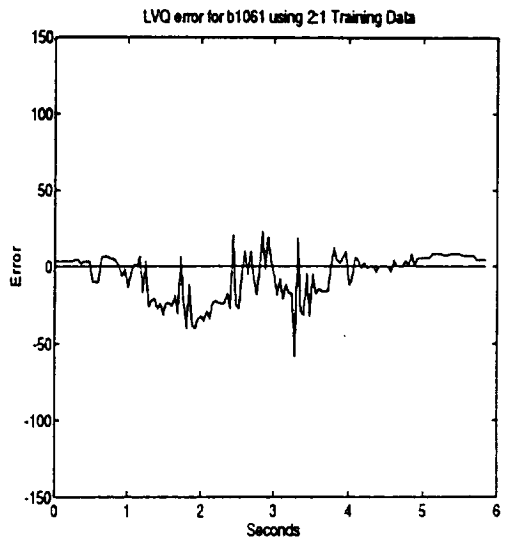

LVQ error for b1046 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of LVQ for b1060 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

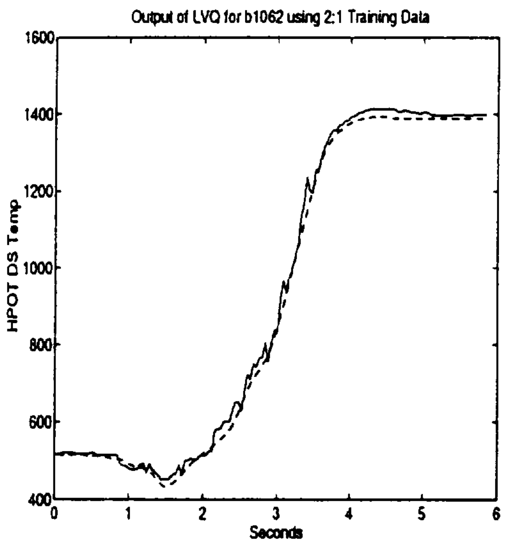

LVQ error for b1060 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



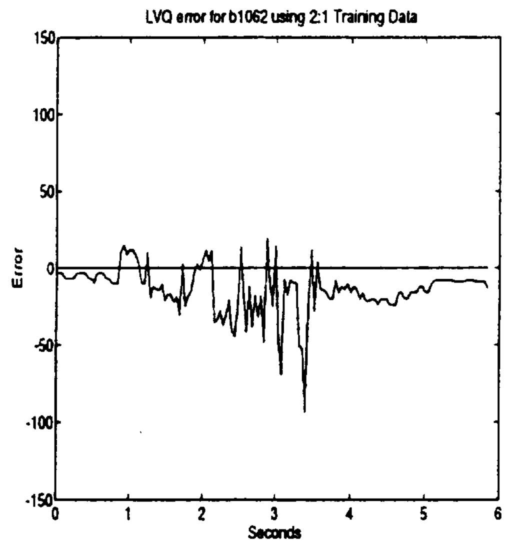
Output of LVQ for b1061 using 2:1 Training Data
HPOT DS Temp
Seconds



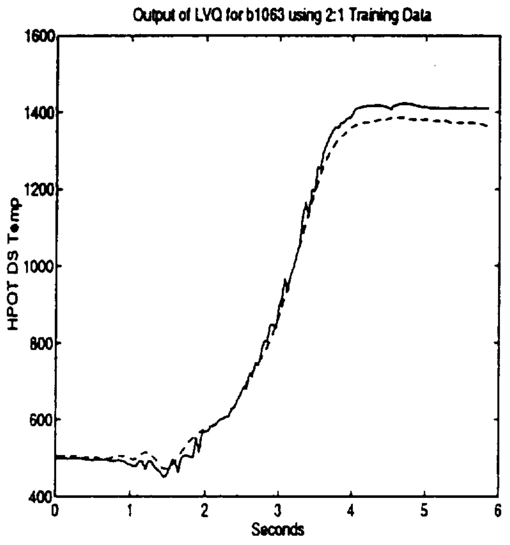
LVQ error for b1061 using 2:1 Training Data
Error
Seconds

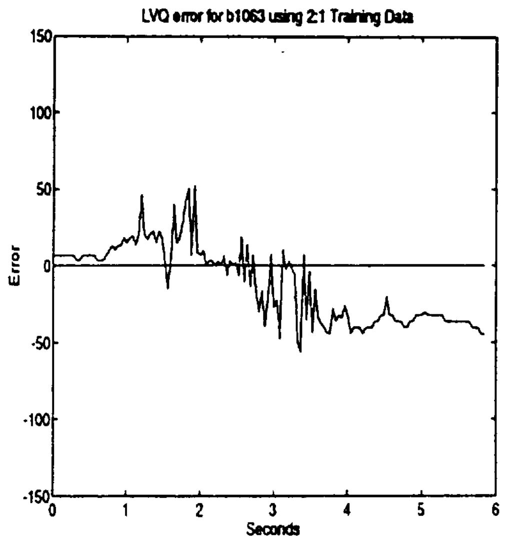


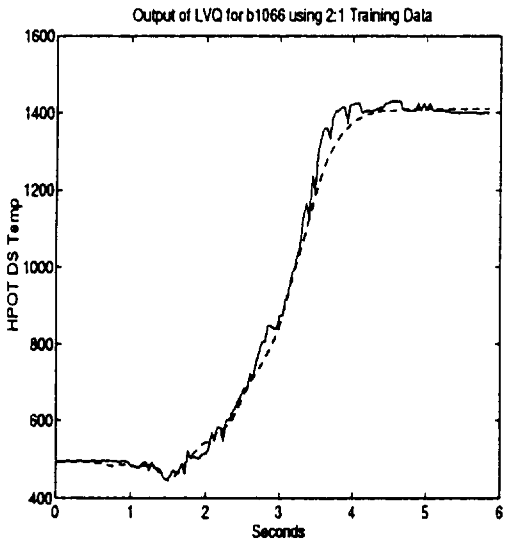
Output of LVQ for b1062 using 2:1 Training Data
HPOT DS Temp
Seconds

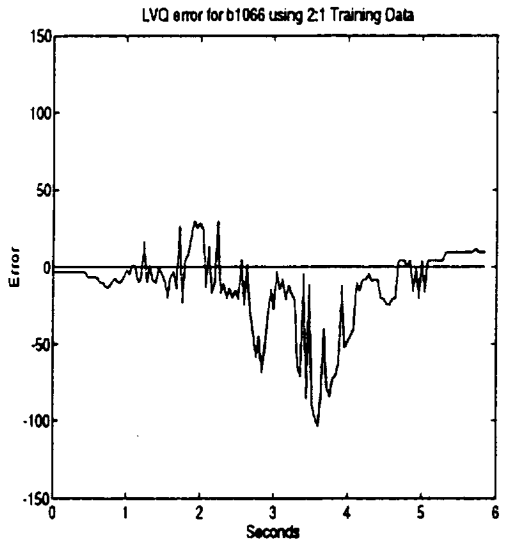


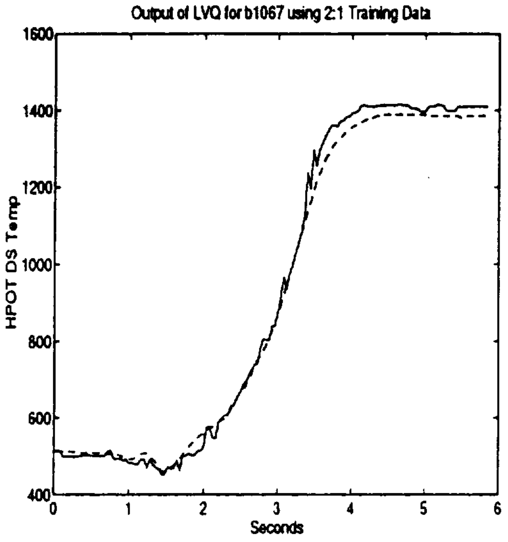
LVQ error for b1062 using 2:1 Training Data
Error
Seconds


Output of LVQ for b1063 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

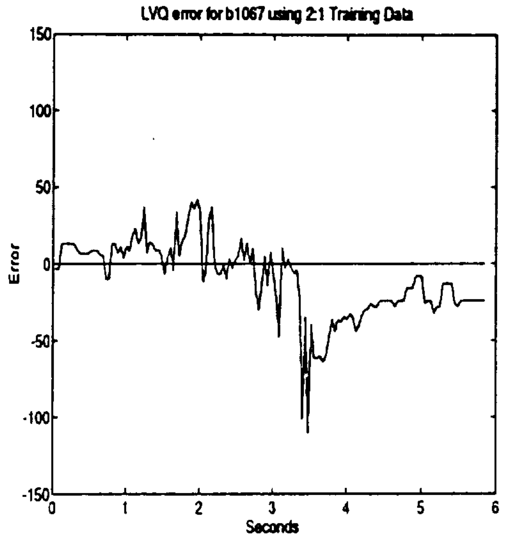

LVQ error for b1063 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of LVQ for b1066 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

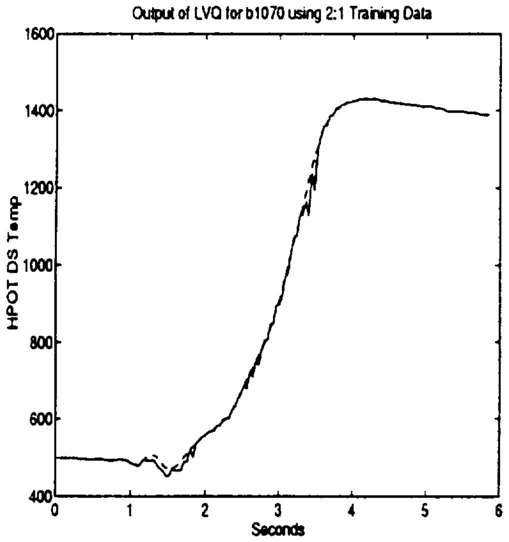

LVQ error for b1066 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6



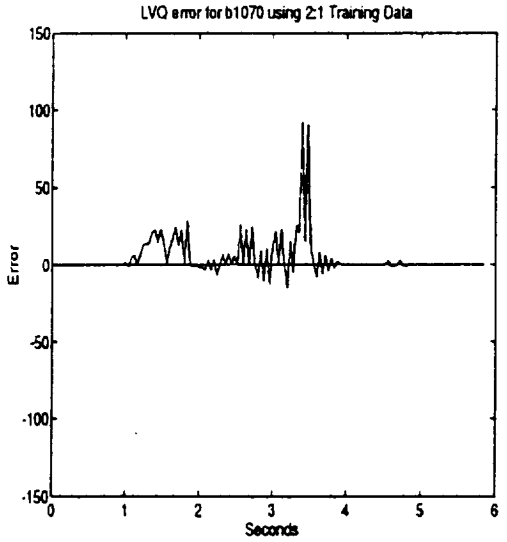
Output of LVQ for b1067 using 2:1 Training Data
HPOT DS Temp
Seconds



LVQ error for b1067 using 2:1 Training Data
Error
Seconds

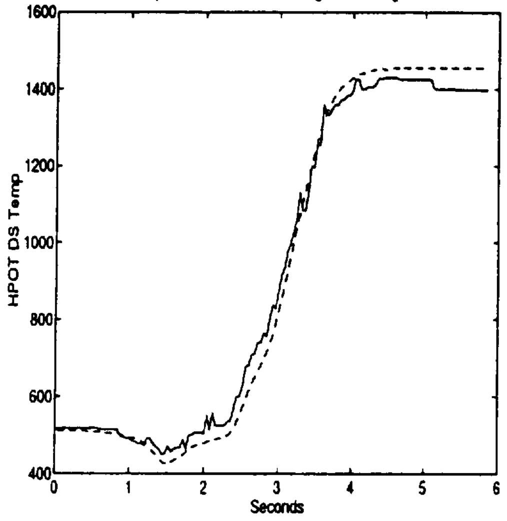


Output of LVQ for b1070 using 2:1 Training Data
HPOT DS Temp
Seconds

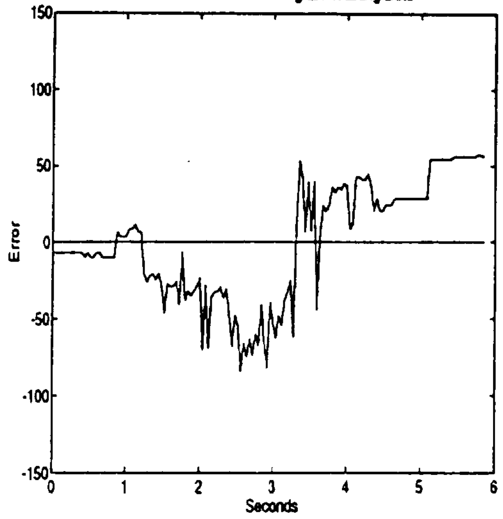


LVQ error for b1070 using 2:1 Training Data
Error
Seconds

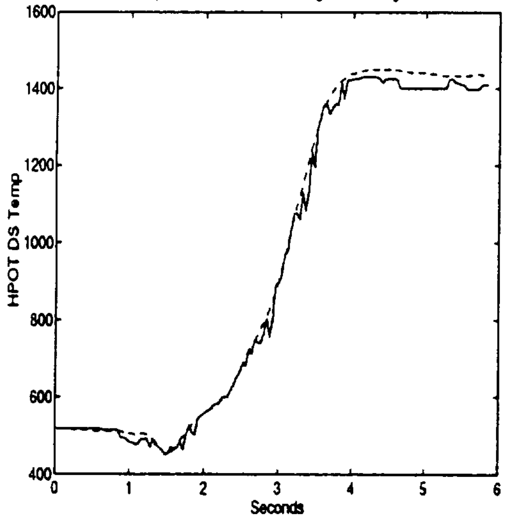
Output of LVQ for b1071 using 2:1 Training Data



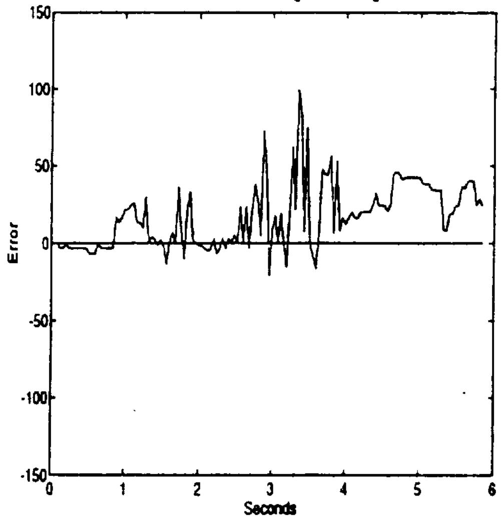
LVQ error for b1071 using 2:1 Training Data

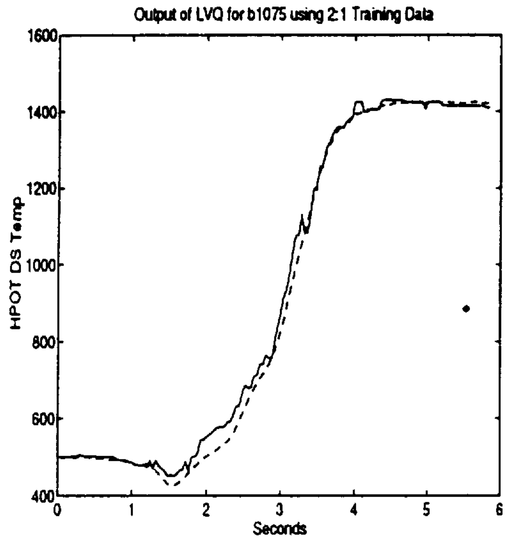


Output of LVQ for b1072 using 2:1 Training Data

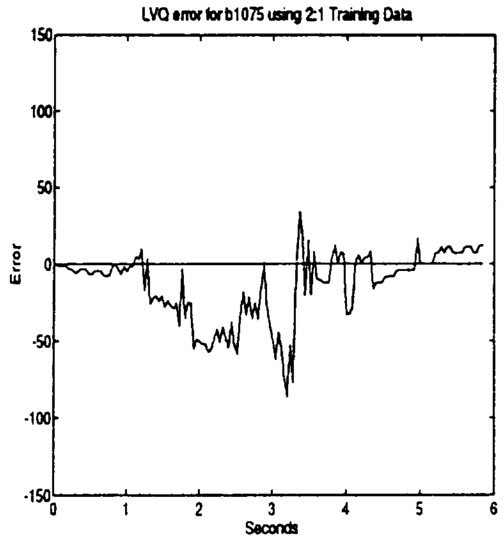


LVQ error for b1072 using 2:1 Training Data

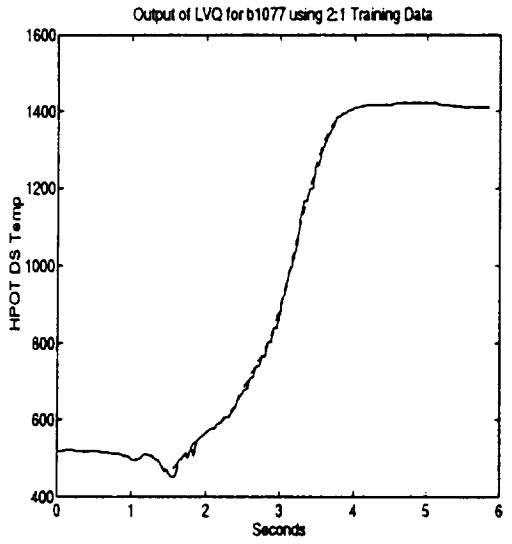




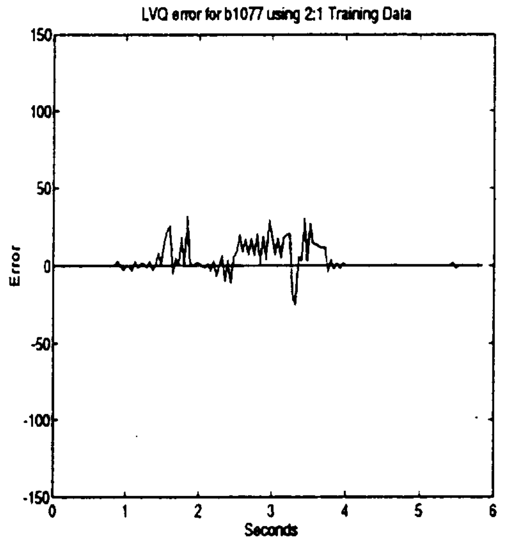
Output of LVQ for b1075 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6



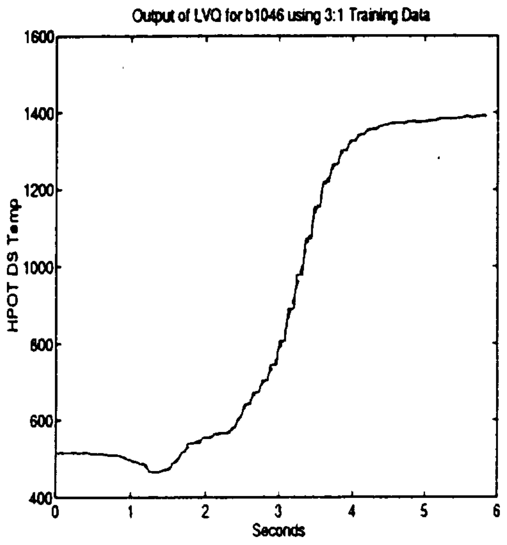
LVQ error for b1075 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

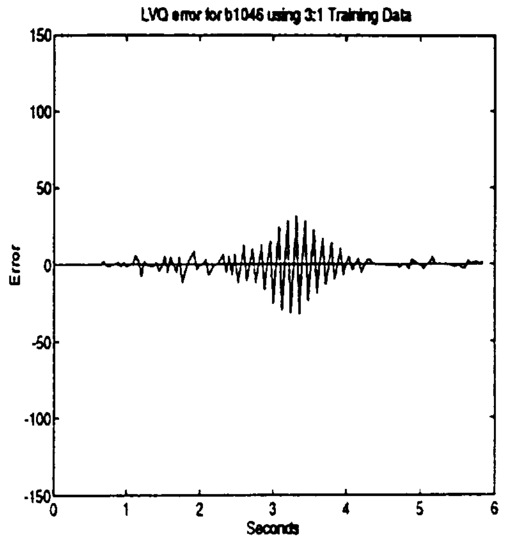


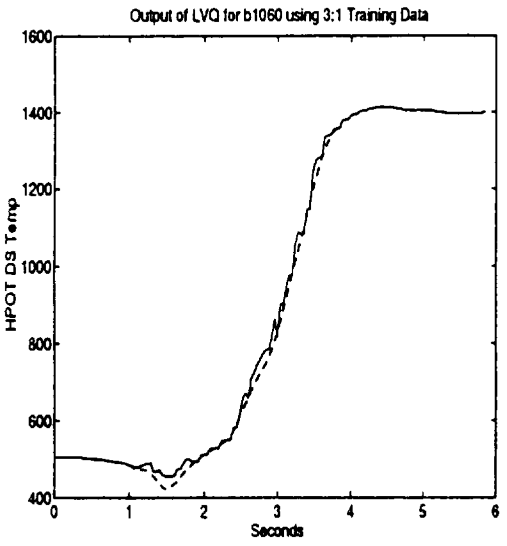
Output of LVQ for b1077 using 2:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

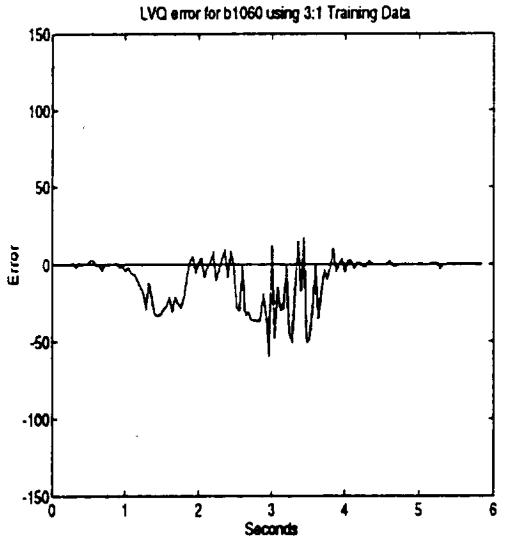


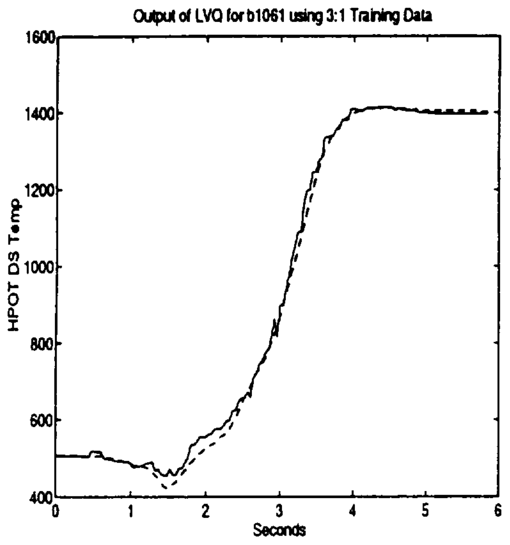
LVQ error for b1077 using 2:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

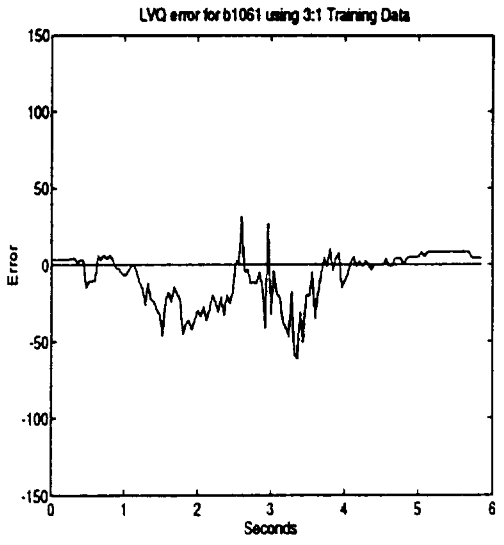

Output of LVQ for b1046 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

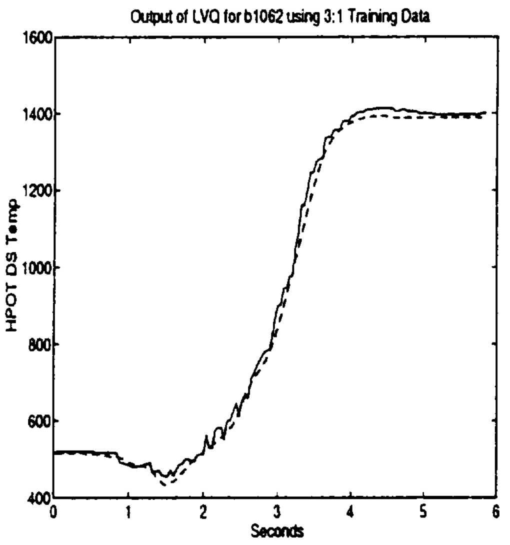

LVQ error for b1046 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

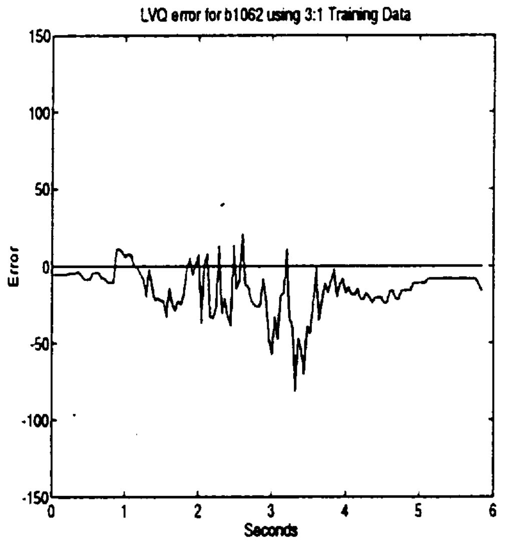

Output of LVQ for b1060 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

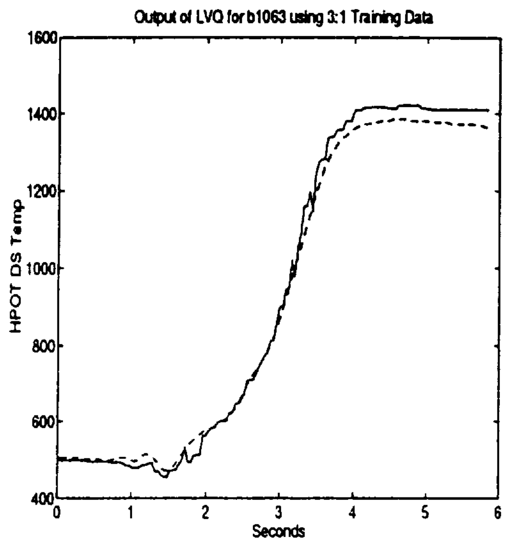

LVQ error for b1060 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

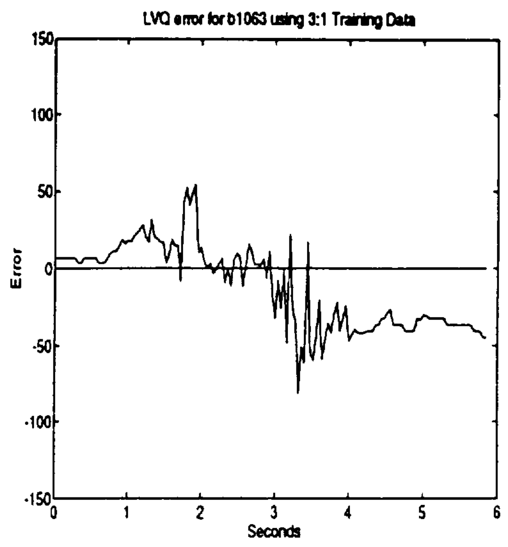

Output of LVQ for b1061 using 3:1 Training Data
HPOT DS Temp
Seconds

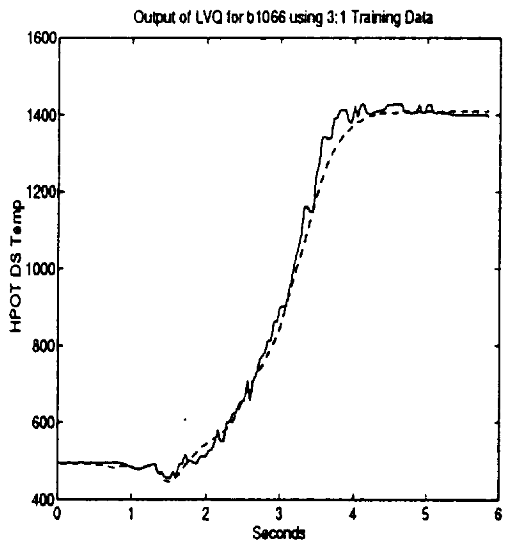

LVQ error for b1061 using 3:1 Training Data
Error
Seconds

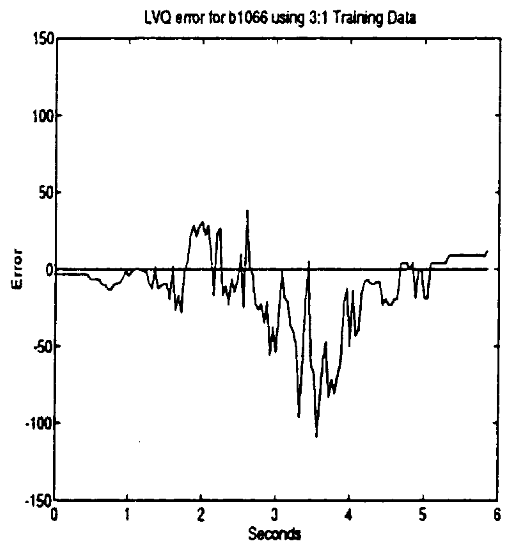

Output of LVQ for b1062 using 3:1 Training Data
HPOT DS Temp
Seconds

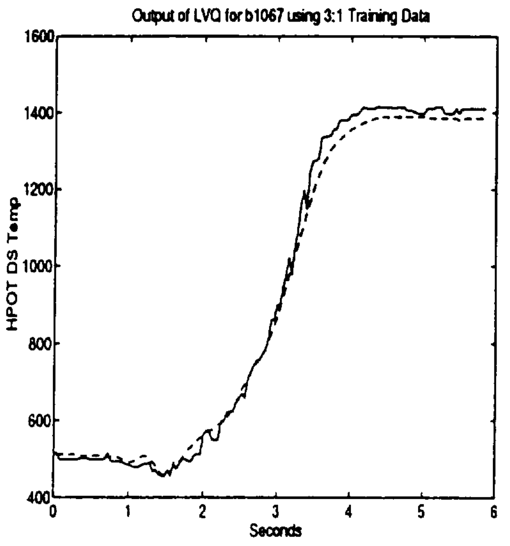

LVQ error for b1062 using 3:1 Training Data
Error
Seconds

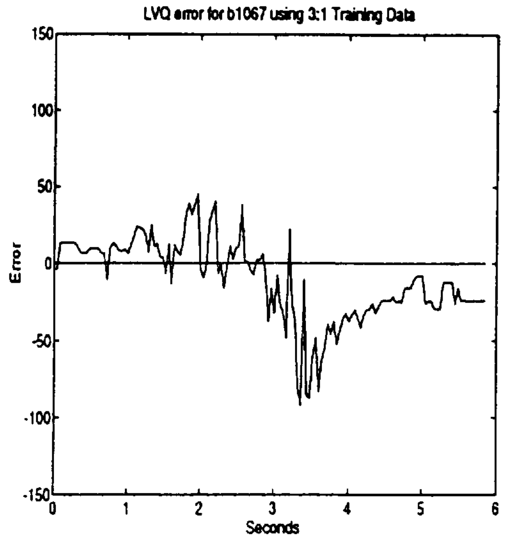

Output of LVQ for b1063 using 3:1 Training Data
HPOT DS Temp
Seconds

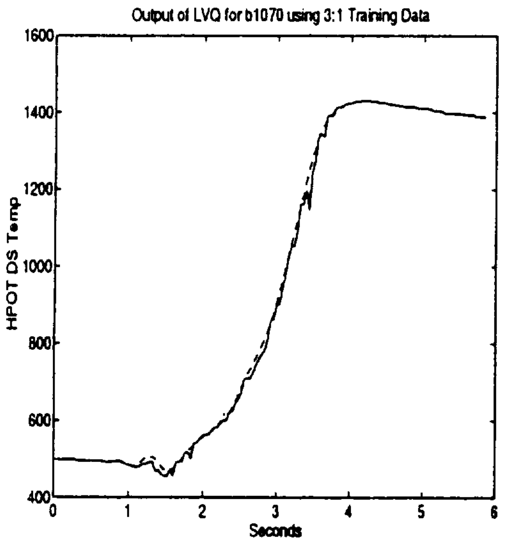

LVQ error for b1063 using 3:1 Training Data
Error
Seconds

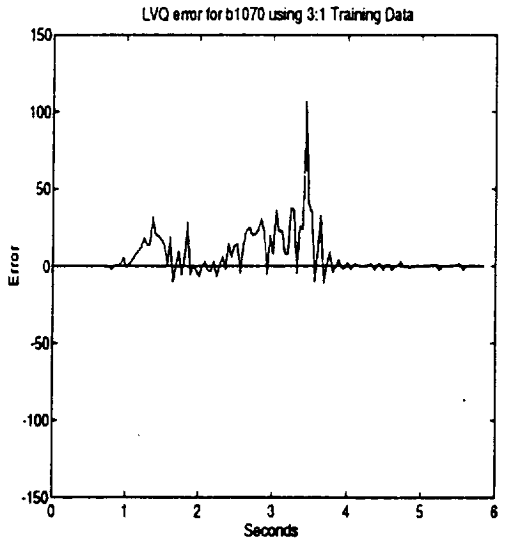

Output of LVQ for b1066 using 3:1 Training Data
HPOT DS Temp
Seconds

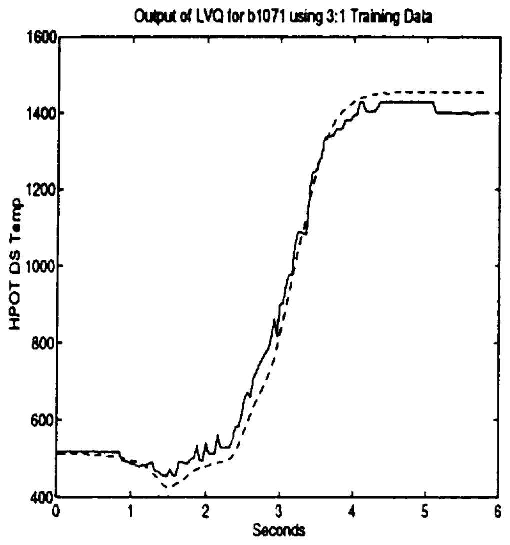

LVQ error for b1066 using 3:1 Training Data
Error
Seconds

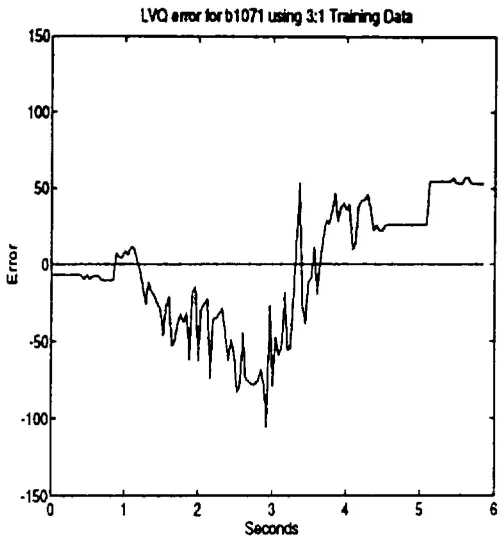

Output of LVQ for b1067 using 3:1 Training Data
HPOT DS Temp
Seconds

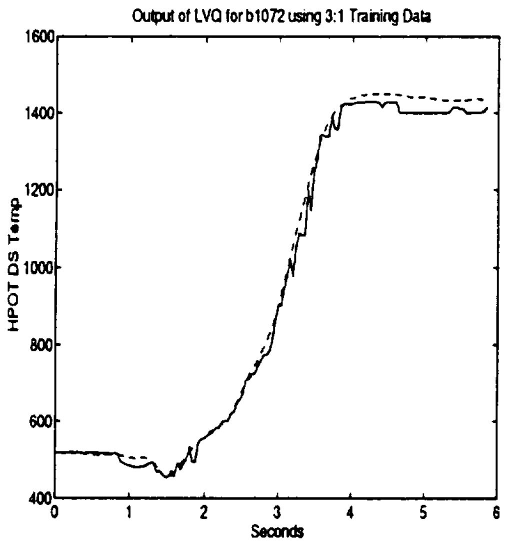

LVQ error for b1067 using 3:1 Training Data
Error
Seconds

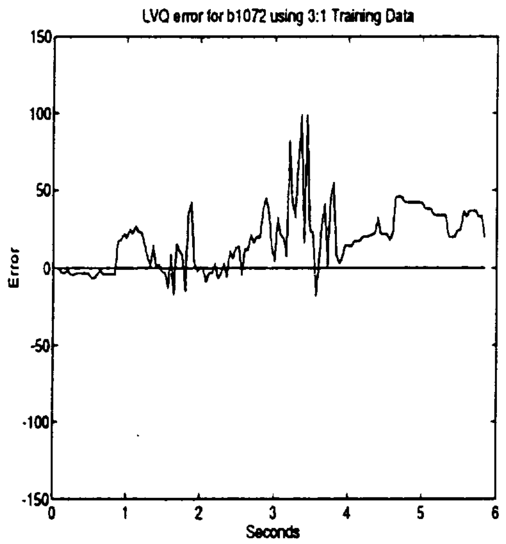

Output of LVQ for b1070 using 3:1 Training Data
HPOT DS Temp
Seconds

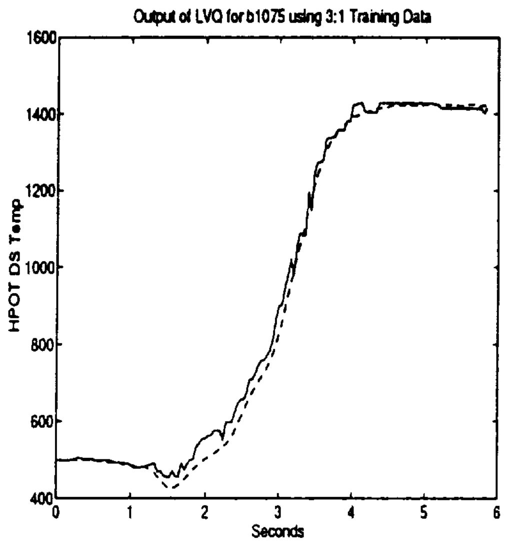

LVQ error for b1070 using 3:1 Training Data
Error
Seconds

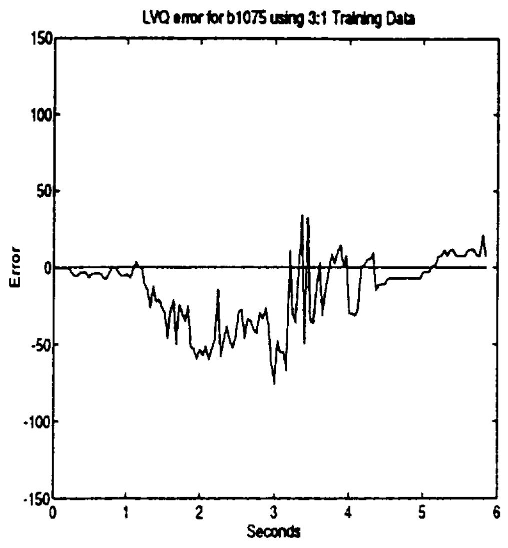

Output of LVQ for b1071 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

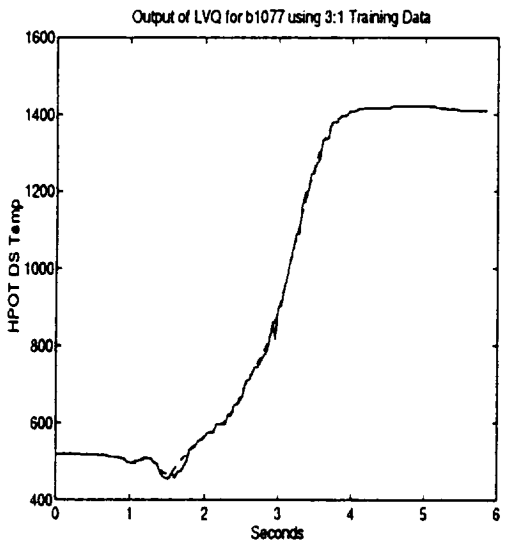

LVQ error for b1071 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

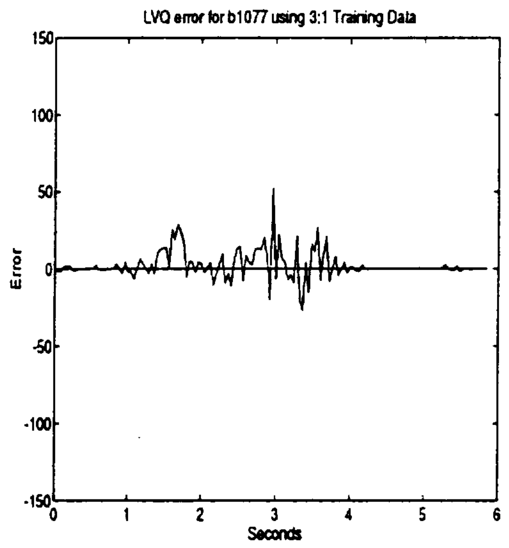

Output of LVQ for b1072 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

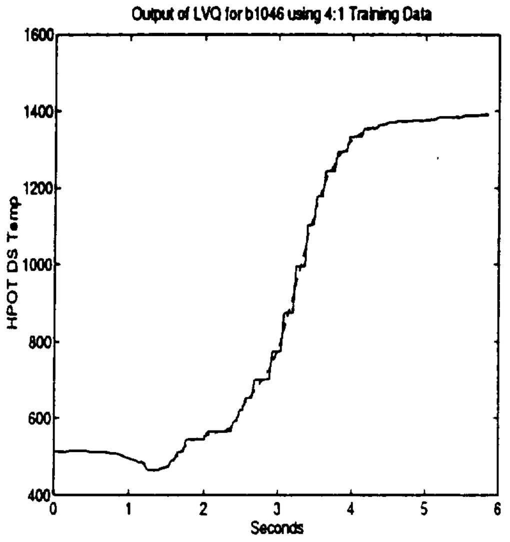

LVQ error for b1072 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

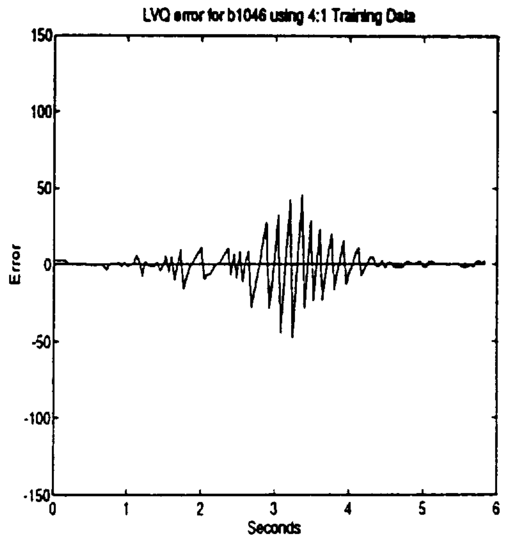

Output of LVQ for b1075 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

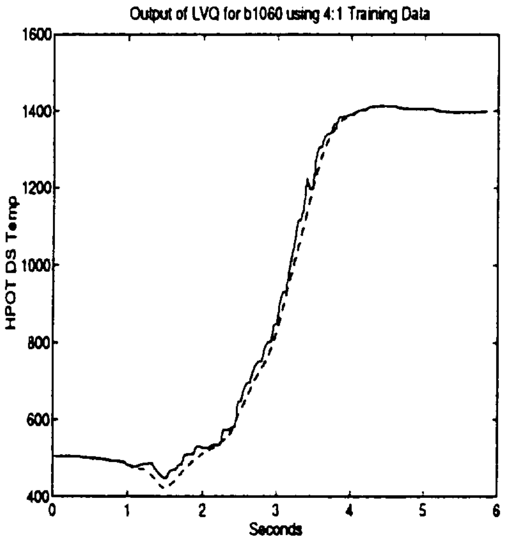

LVQ error for b1075 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

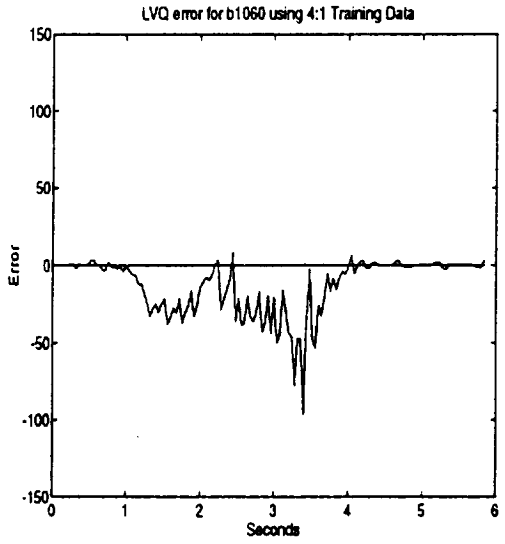

Output of LVQ for b1077 using 3:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

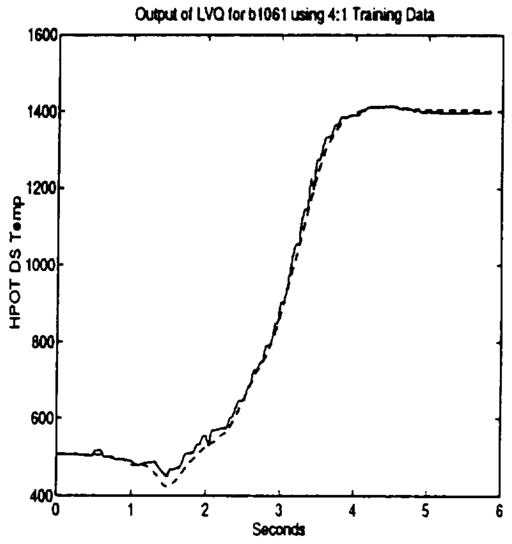

LVQ error for b1077 using 3:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

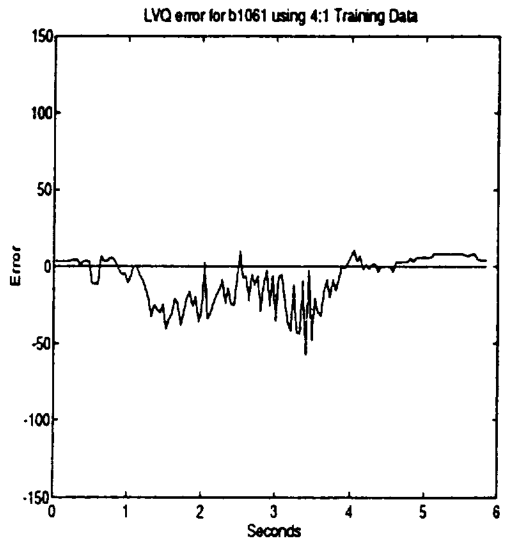

Output of LVQ for b1046 using 4:1 Training Data
HPOT DS Temp
Seconds

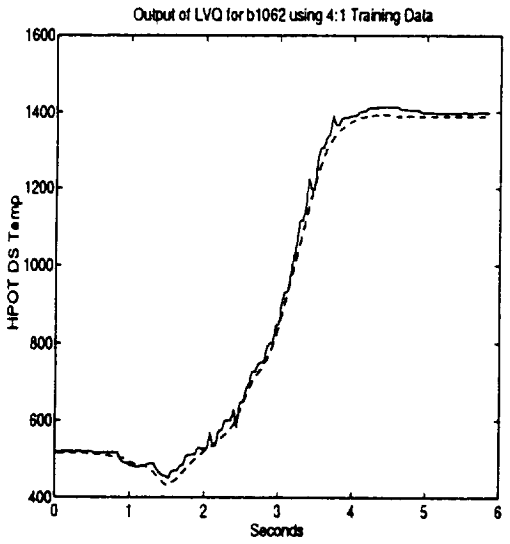

LVQ error for b1046 using 4:1 Training Data
Error
Seconds

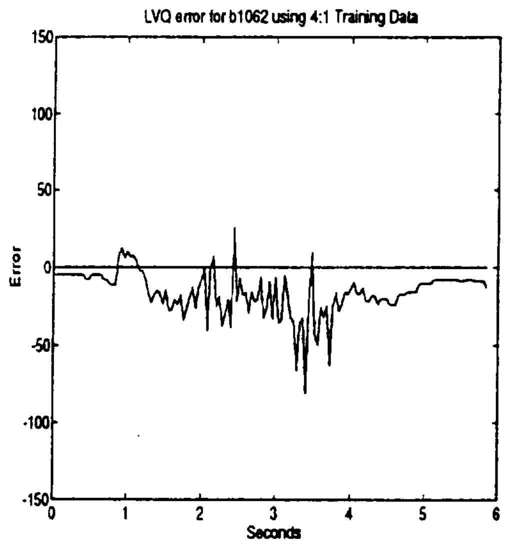

Output of LVQ for b1060 using 4:1 Training Data
HPOT DS Temp
Seconds

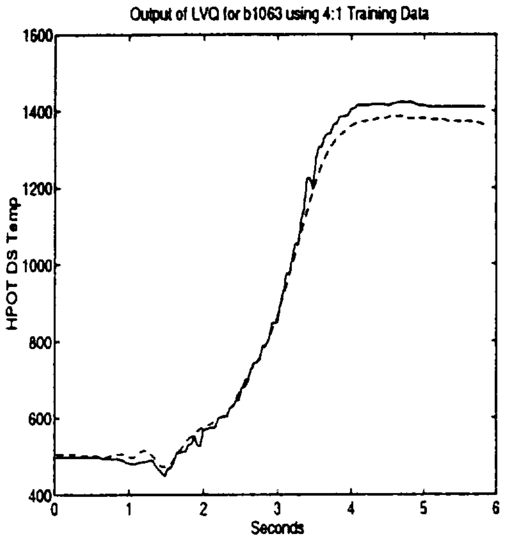

LVQ error for b1060 using 4:1 Training Data
Error
Seconds

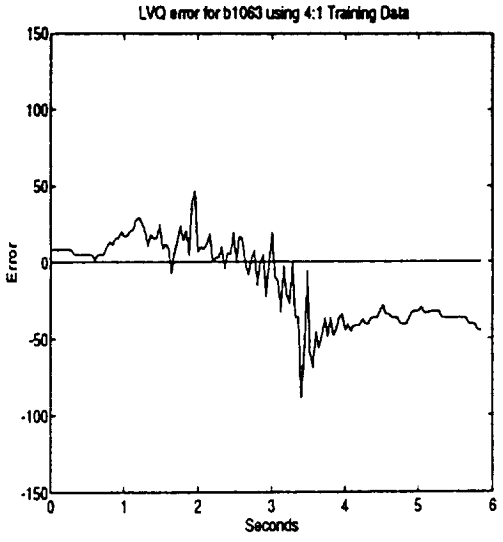

Output of LVQ for b1061 using 4:1 Training Data
HPOT DS Temp
Seconds

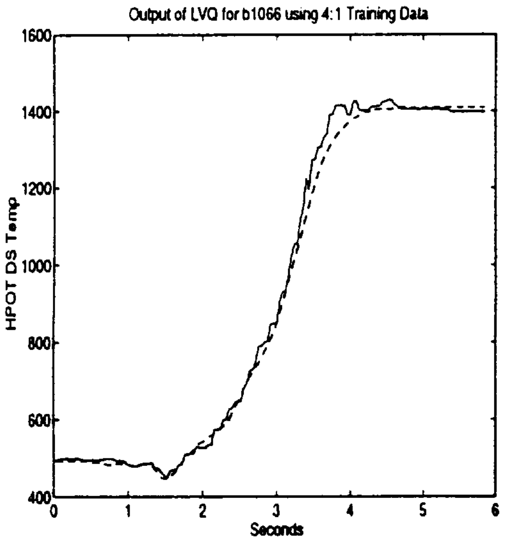

LVQ error for b1061 using 4:1 Training Data
Error
Seconds

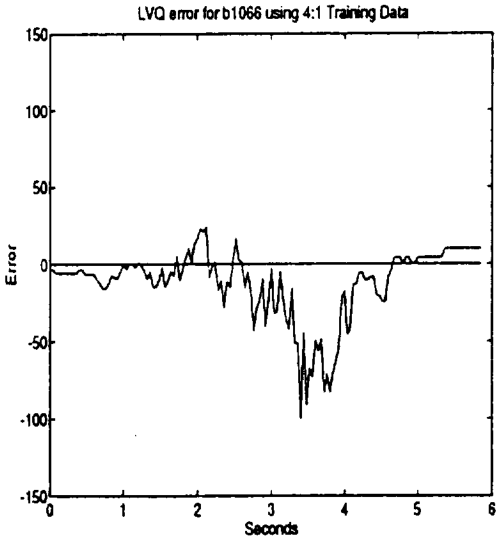

Output of LVQ for b1062 using 4:1 Training Data
HPOT DS Temp
Seconds

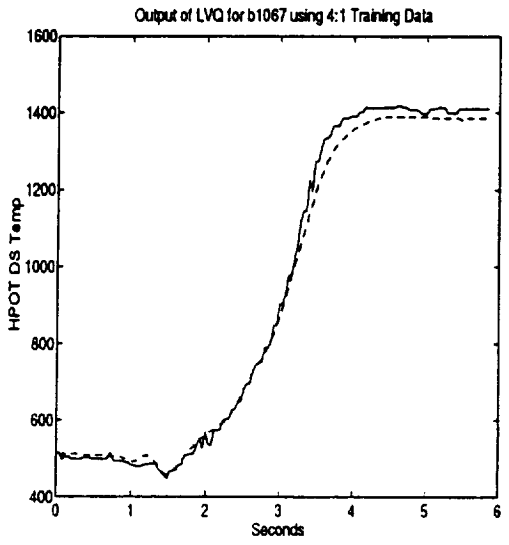

LVQ error for b1062 using 4:1 Training Data
Error
Seconds

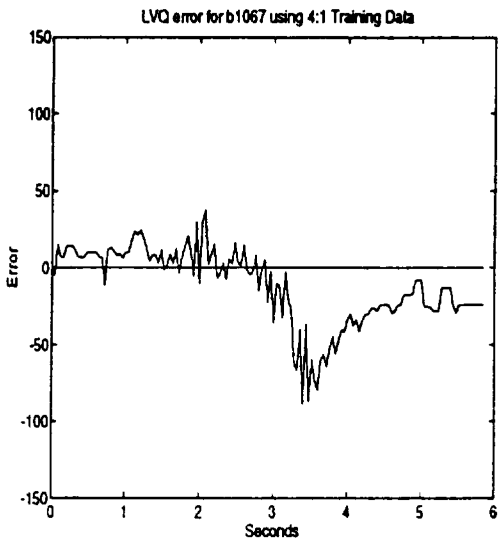

Output of LVQ for b1063 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

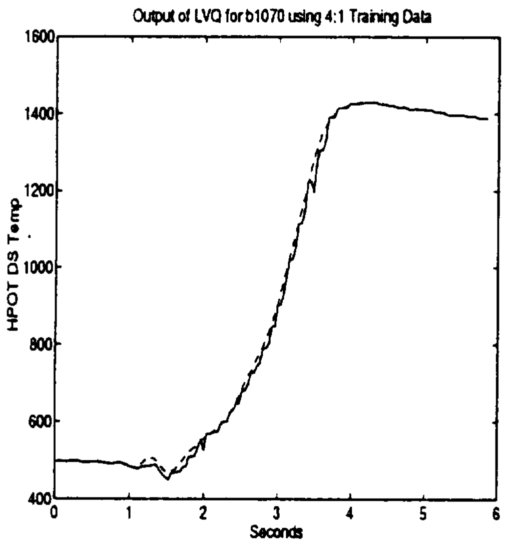

LVQ error for b1063 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

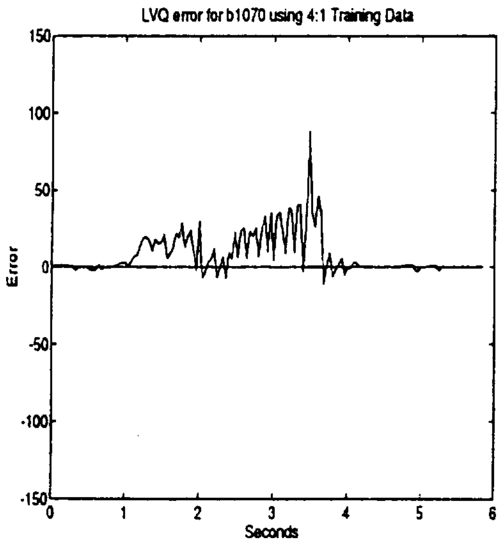

Output of LVQ for b1066 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

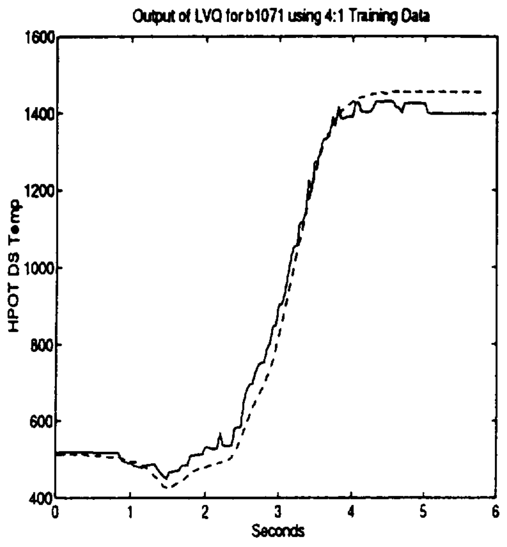

LVQ error for b1066 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

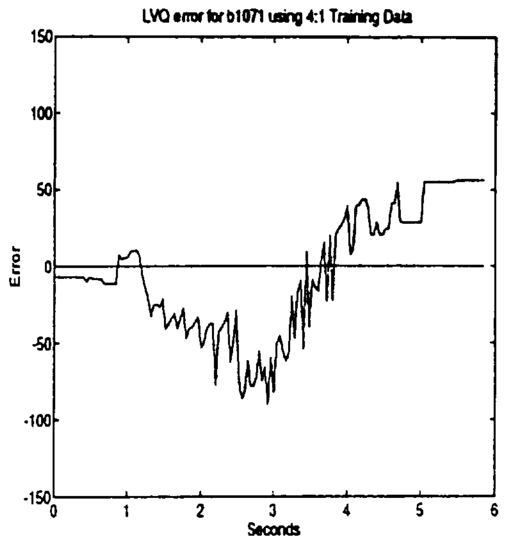

Output of LVQ for b1067 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

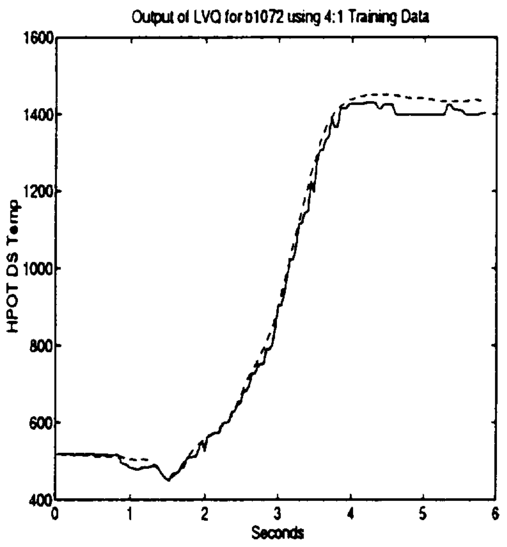

LVQ error for b1067 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

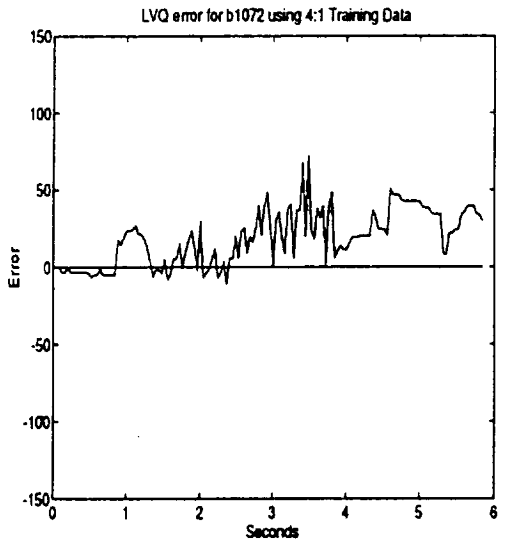

Output of LVQ for b1070 using 4:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

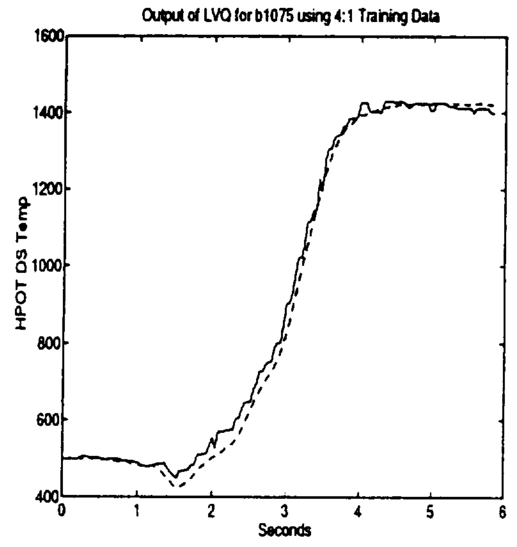

LVQ error for b1070 using 4:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of LVQ for b1071 using 4:1 Training Data
HPOT DS Temp
Seconds

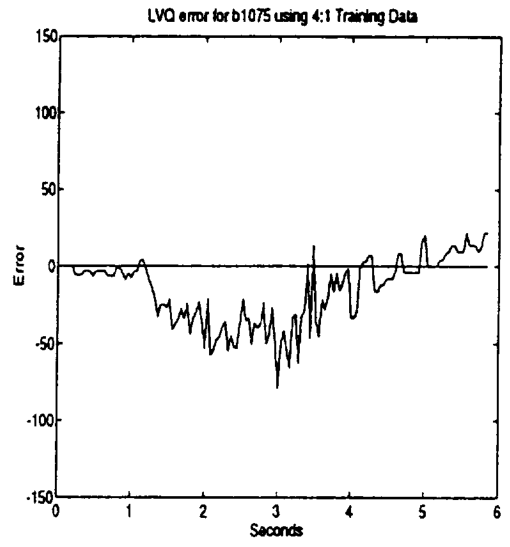

LVQ error for b1071 using 4:1 Training Data
Error
Seconds


Output of LVQ for b1072 using 4:1 Training Data
HPOT DS Temp
Seconds

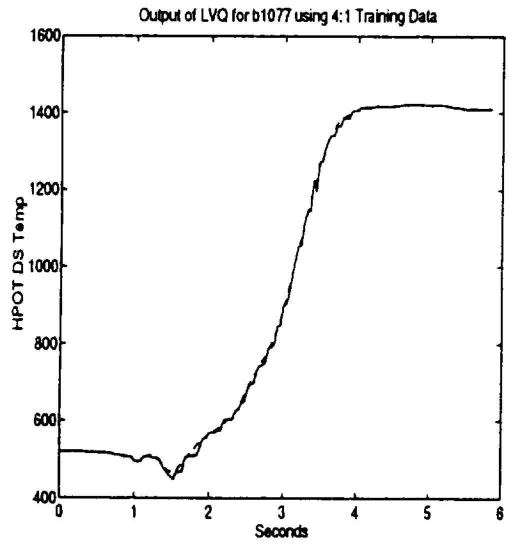

LVQ error for b1072 using 4:1 Training Data
Error
Seconds



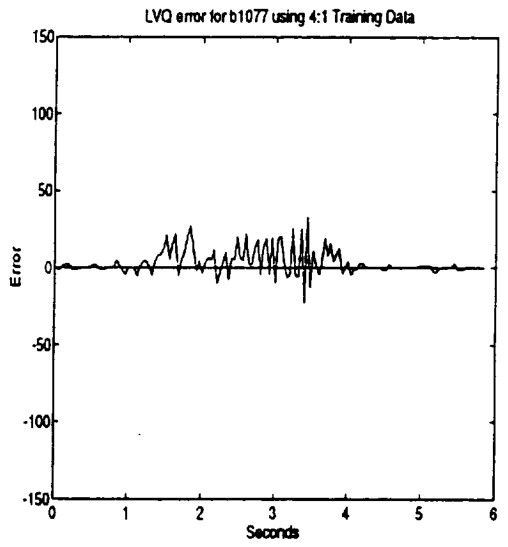
Output of LVQ for b1075 using 4:1 Training Data
HPOT DS Temp
Seconds



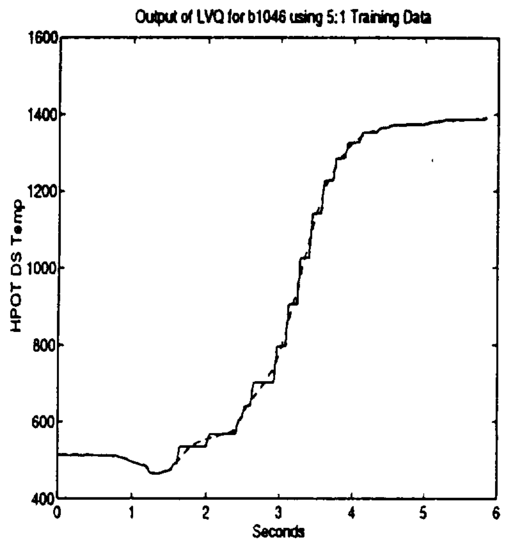
LVQ error for b1075 using 4:1 Training Data
Error
Seconds

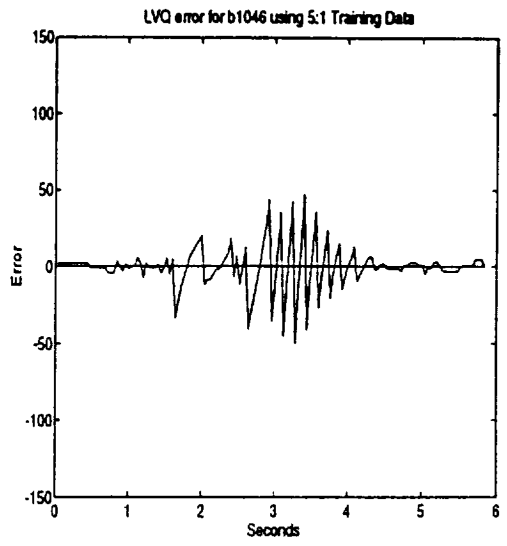


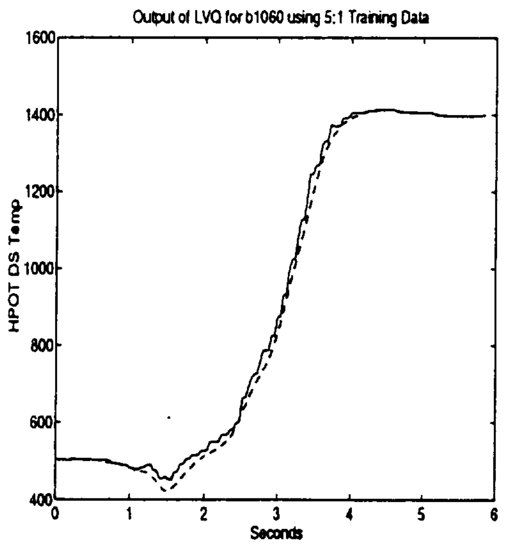
Output of LVQ for b1077 using 4:1 Training Data
HPOT DS Temp
Seconds

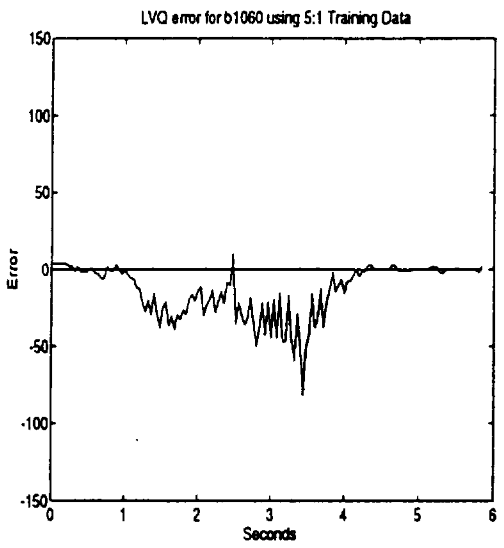


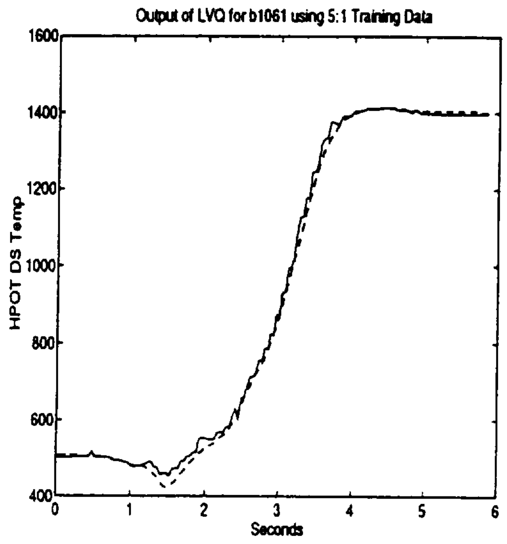
LVQ error for b1077 using 4:1 Training Data
Error
Seconds

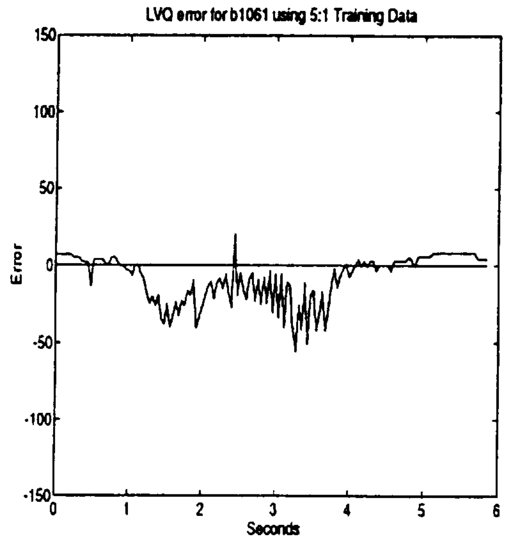

Output of LVQ for b1046 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

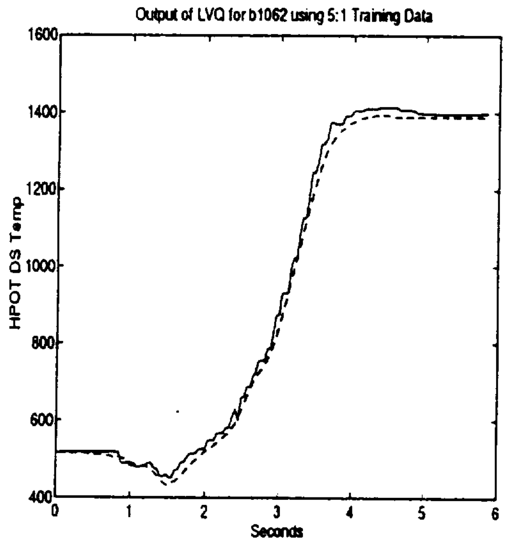

LVQ error for b1046 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

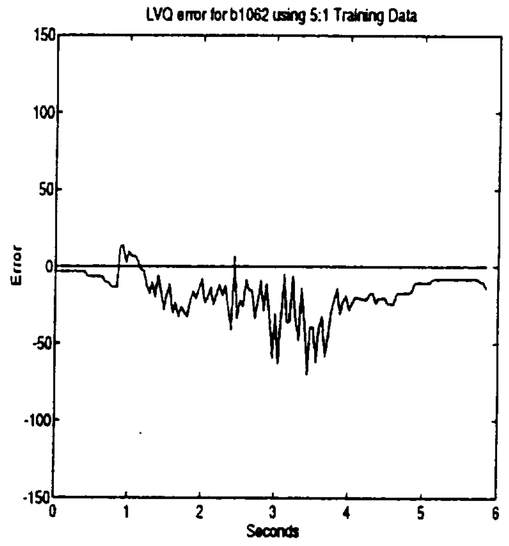

Output of LVQ for b1060 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6

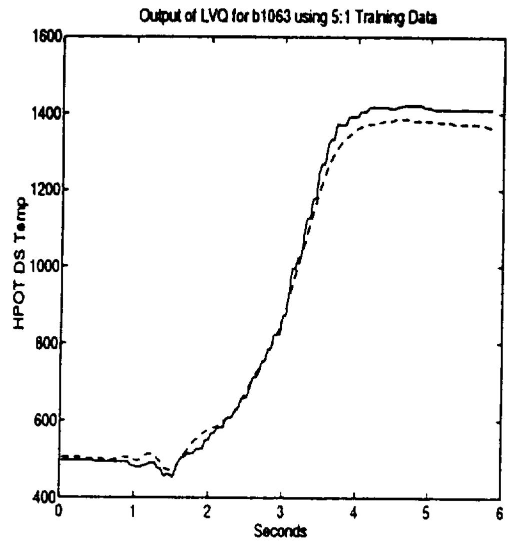

LVQ error for b1060 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

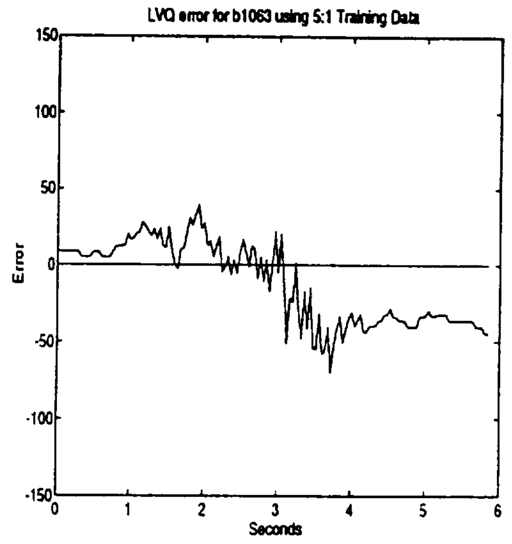

Output of LVQ for b1061 using 5:1 Training Data
HPOT DS Temp
Seconds

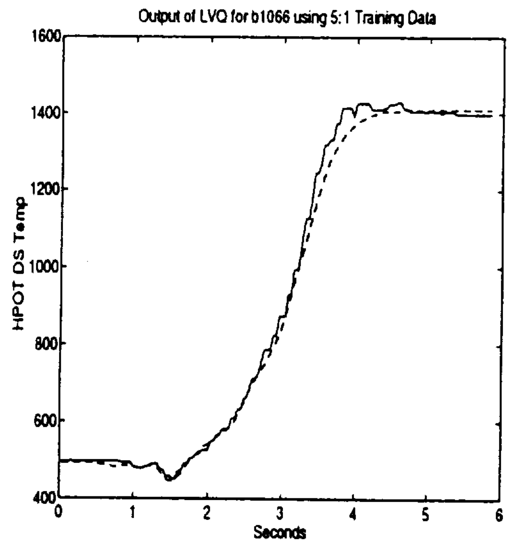

LVQ error for b1061 using 5:1 Training Data
Error
Seconds

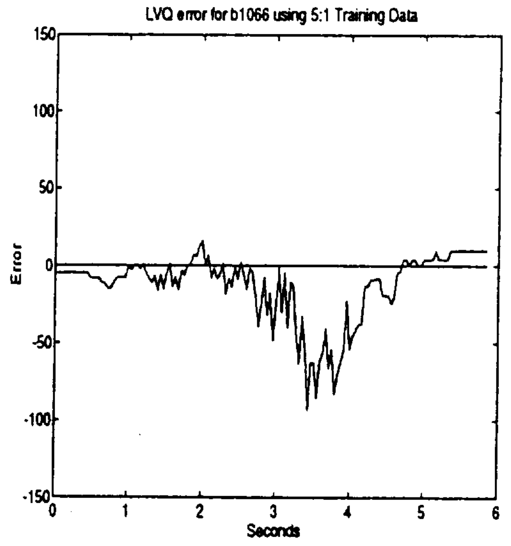

Output of LVQ for b1062 using 5:1 Training Data
HPOT DS Temp
Seconds

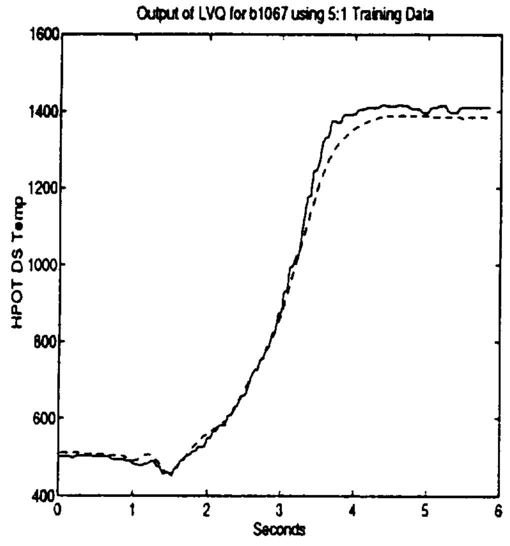

LVQ error for b1062 using 5:1 Training Data
Error
Seconds


Output of LVQ for b1063 using 5:1 Training Data
HPOT DS Temp
Seconds

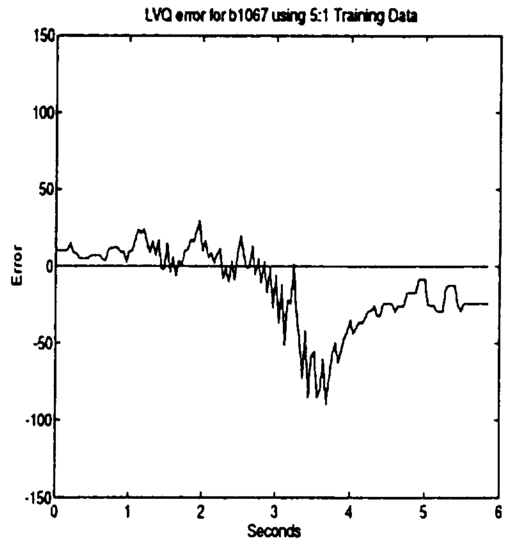

LVQ error for b1063 using 5:1 Training Data
Error
Seconds


Output of LVQ for b1066 using 5:1 Training Data
HPOT DS Temp
Seconds

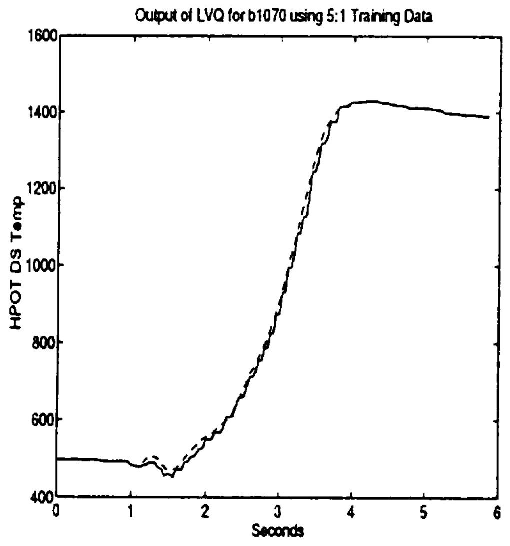

LVQ error for b1066 using 5:1 Training Data
Error
Seconds



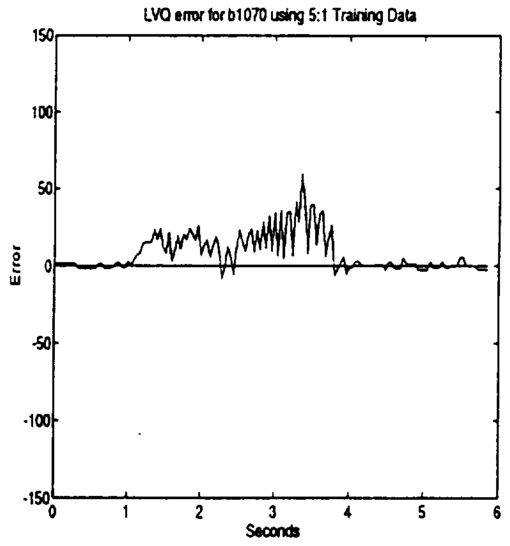
Output of LVQ for b1067 using 5:1 Training Data
HPOT DS Temp
Seconds



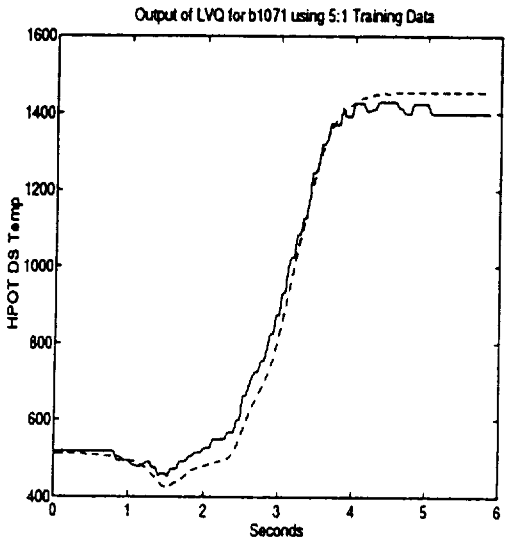
LVQ error for b1067 using 5:1 Training Data
Error
Seconds

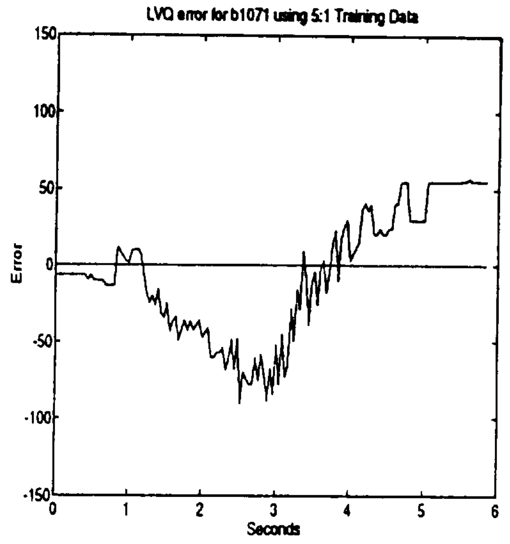


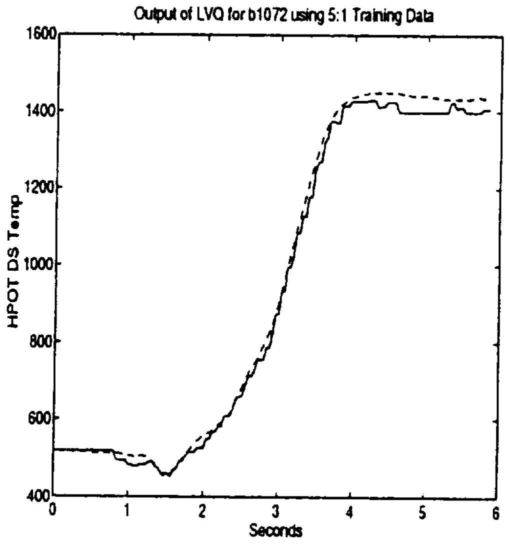
Output of LVQ for b1070 using 5:1 Training Data
HPOT DS Temp
Seconds

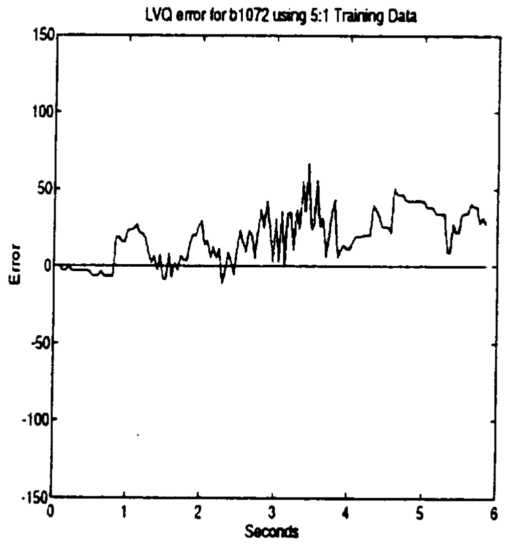


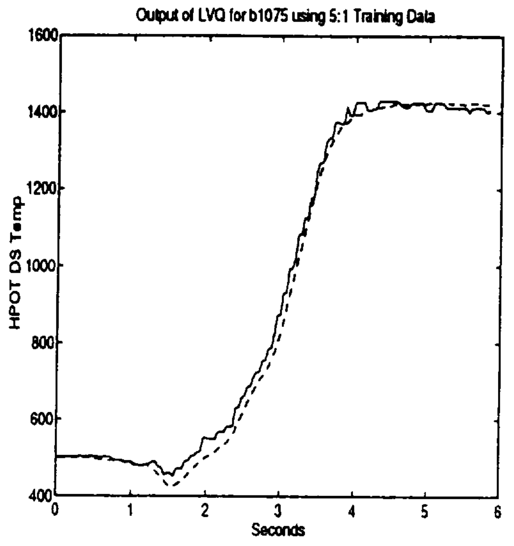
LVQ error for b1070 using 5:1 Training Data
Error
Seconds

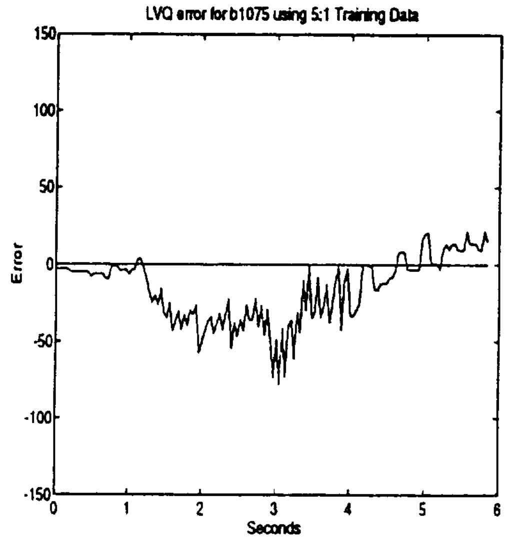

Output of LVQ for b1071 using 5:1 Training Data
HPOT DS Temp
Seconds

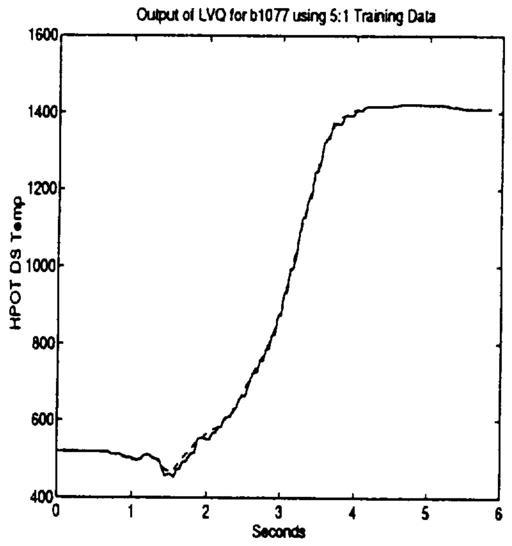

LVQ error for b1071 using 5:1 Training Data
Error
Seconds

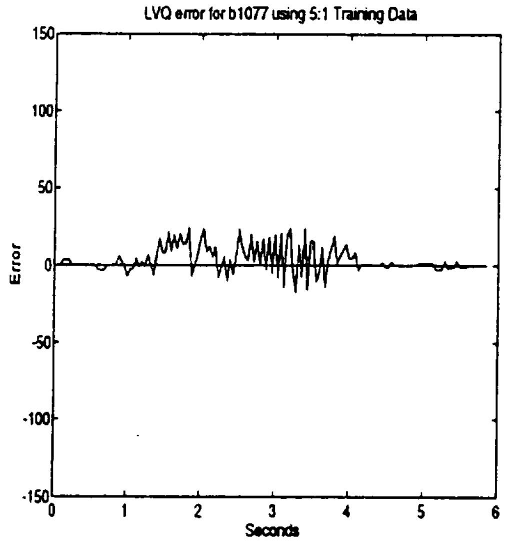

Output of LVQ for b1072 using 5:1 Training Data
HPOT DS Temp
Seconds


LVQ error for b1072 using 5:1 Training Data
Error
Seconds


Output of LVQ for b1075 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6


LVQ error for b1075 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6


Output of LVQ for b1077 using 5:1 Training Data
HPOT DS Temp
Seconds
1600
1400
1200
1000
800
600
400
0
1
2
3
4
5
6


LVQ error for b1077 using 5:1 Training Data
Error
Seconds
150
100
50
0
-50
-100
-150
0
1
2
3
4
5
6

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 1993	3. REPORT TYPE AND DATES COVERED Final Contractor Report

4. TITLE AND SUBTITLE LVQ and Backpropagation Neural Networks Applied to NASA SSME Data	5. FUNDING NUMBERS WU-584-03-11 NCC3-308
6. AUTHOR(S) Timothy F. Doniere and Atam P. Dhawan	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) University of Cincinnati Department of Electrical and Computer Engineering Cincinnati, Ohio 45221 ML 30	8. PERFORMING ORGANIZATION REPORT NUMBER E-9348
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191	10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA CR-195418

11. SUPPLEMENTARY NOTES

Project manager, June F. Zakrajsek, Space Propulsion Technology Division, NASA Lewis Research Center, organization code 5310, (216) 433-7470.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 15 This publication is available from the NASA Center for Aerospace Information, (301) 621-0390.	12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Feedforward neural networks with Backpropagation learning have been used as function approximators for modeling the Space Shuttle Main Engine (SSME) sensor signals. The modeling of these sensor signals is aimed at the development of a sensor fault detection system that can be used during test firings. The generalization capability of a neural network based function approximator depends on the training vectors which in this application may be derived from a number of SSME ground test-firings. This yields a large number of training vectors. Large training sets can cause the time required to train the network to be very large. Also, the network may not be able to generalize for large training sets. To reduce the size of the training sets, the SSME test-firing data is reduced using the Learning Vector Quantization (LVQ) based technique. Different compression ratios were used to obtain compressed data in training the neural network model. The performance of the neural model trained using reduced sets of training patterns is presented and compared with the performance of the model trained using complete data. The LVQ can also be used as a function approximator. The performance of the LVQ as a function approximator using reduced training sets is presented and compared with the performance of the backpropagation network.

14. SUBJECT TERMS Neural networks; Space shuttle main engine			15. NUMBER OF PAGES 79 16. PRICE CODE A05
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT