Classification Techniques for Fault Detection and Diagnosis of an Air-Handling Unit

John M. House, Ph.D.
Associate Member ASHRAE

Won Yong Lee, Ph.D.

Dong Ryul Shin, Ph.D.

ABSTRACT

The objective of this study is to demonstrate the application of several classification techniques to the problem of detecting and diagnosing faults in data generated by a variable-air-volume air-handling unit simulation model and to describe the strengths and weaknesses of the techniques considered. Artificial neural network classifiers, nearest neighbor classifiers, nearest prototype classifiers, a rule-based classifier, and a Bayes classifier are considered for both fault detection and diagnostics.

Based on the performance of the classification techniques, the Bayes classifier appears to be a good choice for fault detection. It is a straightforward method that requires limited memory and computational effort, and it consistently yielded the lowest percentage of incorrect diagnoses. For fault diagnosis, the rule-based method is favored for classification problems such as the one considered here, where the various classes of faulty operation are well separated and can be distinguished by a single dominant symptom or feature. Results also indicate that the success or failure of classification techniques hinges to a large degree on an ability to separate different classes of operation in some feature (temperature, pressure, etc.) space. Hence, preprocessing of data to extract dominant features is as important as the selection of the classifier.

INTRODUCTION

It is generally accepted that the performance of heating, ventilating, and air-conditioning (HVAC) systems often falls short of expectations. Sensors and actuators degrade and fail, valves and dampers leak and stick, coils become fouled, and other problems arise. Some faults are easily isolated, while others may only be apparent at certain operating conditions. Thus, different approaches (some simple, some sophisticated) can be taken to detect and diagnose faults.

The acknowledgment of the presence of faults in HVAC systems and the fact that faults can lead to occupant discomfort, increased energy use, and shorter equipment life have resulted in considerable effort being devoted to the development of fault detection and diagnostic (FDD) methods for HVAC systems. Numerous studies report the use of expert systems to diagnose faults in HVAC components and systems. Papers by Haberl and Claridge (1987), Anderson et al. (1989), Bagby and Cormier (1989), and Kaler (1990) are a representative sample of this work. Pape et al. (1991) identified faults in a variable-air-volume (VAV) system by comparing the power consumption obtained from a simulation model with embedded faults to predictions based on a near-optimal model. Haves et al. (1996) used a condition-monitoring scheme based on physical models to detect valve leakage and water-side coil-fouling faults in a simulated cooling coil subsystem of an air-handling unit (AHU). Dexter and Benouarets (1996) described a fuzzy model-based fault detection scheme and presented simulation results demonstrating the capability to detect the faults considered by Haves et al. (1996). Artificial neural networks have been examined in several HVAC FDD studies (Li et al. 1996; Lee et al. 1996, 1997). Pernot et al. (1997) compared a rule-based and a neural network classifier for detecting and diagnosing faults in a heating system. Rossi and Braun (1997) described a statistical, rule-based FDD method developed for vapor compression air conditioners and used the method to detect and diagnose five simulated faults. The method also successfully detected and diagnosed these same faults in an experimental test rig. Dodier

John M. House is a mechanical engineer in the Mechanical Systems and Controls Group, Building Environment Division, Building and Fire Research Laboratory, National Institute of Standards and Technology, Gaithersburg, Md. **Won Yong Lee** is a senior researcher and **Dong Ryul Shin** is a principal researcher at the Korea Institute of Energy Research in Taejon, Korea.

et al. (1998) used Bayes' rule to predict the state of operation of a fan-powered VAV box. This literature review, though not comprehensive, does provide evidence of the variety of FDD approaches that are being examined for HVAC systems.

All FDD methods use classification techniques. In fault detection, a pattern of variables or parameters (also called features) representing current operation are classified as either normal or faulty. In some cases, a third class representing unknown operation may also be used. In fault diagnosis, the analysis involves classification of faulty conditions to a specific type of fault. This often involves comparing the current pattern of conditions to patterns that are deemed representative of each of the faults considered and labeling the current pattern according to the fault type that it resembles most. Thus, classification involves pattern recognition. Bezdek (1993) provides an excellent overview of the subject of pattern recognition and describes various statistical, fuzzy, and neural network models for pattern recognition that can be found in the literature.

Because all FDD methods employ classification techniques (i.e., assigning data to a class is classification, whether a heuristic rule or a sophisticated algorithm makes the assignment), one would like to use the "best" classification technique available. Identifying what technique is "best" is a difficult task involving subjective criteria. One criterion that does not appear to be subjective is the number of correct diagnoses out of a set of labeled test data. Subjectivity can creep into this criterion through biases in the data. For instance, the effectiveness of a technique may depend on whether the fault considered involves performance degradation or a complete failure. If the training and testing data do not contain faults of both types, results may be misleading. Other criteria might include the training effort required for a particular technique, computational resources required for a technique, and how well the technique is understood where techniques based on simple concepts and algorithms might be favored over more sophisticated techniques. Because of this subjectivity, the selection process sometimes becomes a process of elimination where problem constraints lead naturally to a particular technique.

The objective of this study is to demonstrate the application of several classification techniques to the problem of detecting and diagnosing faults in data generated by a VAV AHU simulation model. Determining the "best" classification technique is outside the scope of this study and may, in fact, be an exercise in futility. Instead, an attempt is made to describe the strengths and weaknesses of the techniques considered.

The first part of this paper briefly describes the considered system and simulation model. Next, residuals that characterize the operation of the system are defined and the faults considered are described. An overview of c-means clustering and several classification techniques is then presented. Fault detection and diagnosis results obtained by applying the clustering and classification techniques are then discussed. Finally, conclusions from this study are presented.

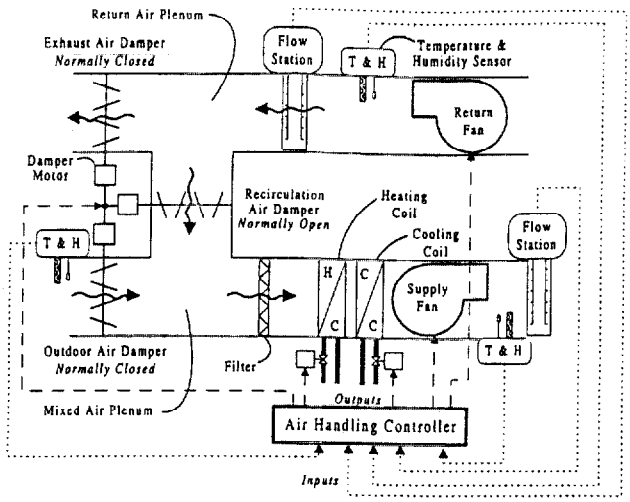

Figure 1 Schematic diagram of a variable-air-volume air-handling unit.

SYSTEM AND MODEL DESCRIPTIONS

Figure 1 is a schematic diagram of a VAV AHU. The AHU consists of fans, dampers, coils, sensors, and controllers. The simulation model used in this study also includes three rooms and three VAV boxes (not shown). The static pressure in the main supply duct was maintained at a constant setpoint value of 249 Pa (1.0 in. of water) by controlling the supply fan speed. The supply air temperature was maintained at a varying setpoint value by adjusting the cooling water control valve. The setpoint temperature was determined from a reset schedule based on the room temperatures and varied from 11°C (51.8°F) to 18°C (64.4°F). The difference between the supply and return airflow rates was maintained at a constant setpoint value of 472 L/s (1000 ft^3/min) by controlling the speed of the return fan. The room air temperature was maintained at a constant setpoint value of 24°C (75.2°F) by controlling the damper position of the VAV box serving that room. A PID algorithm was used to control the cooling coil control valve and the supply fan, and PI algorithms were used to control the return fan and the VAV box dampers.

A simulation model of the VAV AHU was used to generate the data used in this study. The model is based on steady-state characteristic equations and approximate first-order dynamics. Component models (Wang 1992) were calibrated to fit experimental data obtained from a laboratory AHU. Lee et al. (1997) provides an overview of the simulation model.

RESIDUALS

Definition

Residuals are defined as the difference between the actual and expected values of a variable or parameter. An expected value could be a set point or a model prediction. Lee et al. (1996, 1997) identified patterns of residuals to use as signa-

tures for various faults. The approach is the same in this study, although some of the residuals have changed. The residuals used here are:

$$R_{Ts} = T_s - T_{s,ev} \quad (1)$$

$$R_{Tm} = T_m - T_{m,ev} \quad (2)$$

$$R_{Tr} = T_r - T_{r,sp} \quad (3)$$

$$R_{Ps} = P_s - P_{s,ev} \quad (4)$$

$$R_{Qd} = Q_d - Q_{d,sp} \quad (5)$$

where R denotes residual, T is temperature, P is pressure, and Q is volumetric airflow rate. Subscript s denotes supply air parameters, m denotes mixed air, r denotes room air, d denotes flow difference, sp denotes setpoint values, and ev denotes expected values.

To compute residuals R_{Ts}, R_{Tm}, and R_{Ps}, models are needed for the expected value (under normal operating conditions) of the supply air temperature, mixed air temperature, and supply air pressure, respectively. The expected values of T_s, T_m, and P_s are obtained from steady-state regression equations. Details of the regression equations are not presented because the same set of residuals is used to compare the various classifier techniques. It should be noted, however, that the coefficients of the regression equations were determined using simulation data for two separate Washington, D.C., weather days. A third weather day was also simulated to obtain data that could be used to test the sensitivity of the regression equations and classifier techniques to a variability of this type. Because the classifier techniques classify patterns of residuals obtained using models, models are an important aspect of FDD. More accurate models will allow smaller faults to be detected but may also require more modeling and tuning efforts.

Normalization

Residuals defined by Equations 1 through 5 are normalized so that the dominant symptom residuals have nearly the same magnitude for all fault cases. The residuals are normalized as follows:

$$\bar{R} = \left| \frac{R}{R_{mean}} \right| \quad \text{If } \bar{R} > 1 \text{ then set } \bar{R} = 1 \quad (6)$$

where R_{mean} is the mean value of the considered residual computed using fault data for which that residual is dominant. Values of the normalized residuals are truncated at one to compress the fault data into a smaller region in residual space. The mean value for each residual was determined using only the first day of the three-day data set.

Steady-State Detector

The residual pattern used to classify each fault was developed assuming steady-state conditions prevail when classification is performed and no other fault that could affect the residual pattern exists simultaneously. One advantage of using steady-state patterns is that it is possible to determine the state to which the system will evolve, even when faults exist. The dynamic evolution is harder to predict. Thus, a steady-state detector is used to eliminate transient data. The drawback of using a steady-state detector is that faults causing persistent dynamic operation (unstable controller) are difficult to find because the system does not reach steady state.

Five variables were used in the steady-state detector, namely, T_s, T_m, T_r, P_s, and Q_d. Using least squares regression, straight lines were fit through data from the past minute for each of these variables. If the absolute value of the slope of each line is less than its associated threshold value, the system is deemed to be in steady state.

FAULT DESCRIPTION

Seven faults are considered. At normal operating conditions, the residuals defined by Equations 1 through 5 are expected to be nearly equal to zero. Faults cause one or more of the residuals to increase, thereby making it possible to distinguish normal operation from faulty operation. Ten-hour simulation runs were performed with data computed at 10-second time intervals. Data used for analysis were collected starting at $t = 10{,}000$ s, where t denotes time. The initial 10,000 seconds of data were discarded from each data set because they demonstrated obvious transient behavior. Unless otherwise noted, faults were introduced at the beginning of a simulation run.

The first fault is a stuck cooling coil valve. This fault becomes evident as varying loads cause VAV boxes to open and close. This affects the amount of air flowing across the cooling coil, and the control signal to the cooling coil valve will modulate in an attempt to maintain the supply air temperature at the setpoint value. Eventually the control signal will saturate at one of its limits. Because the estimated value of the supply air temperature $T_{s,ev}$ is a function of the cooling coil valve control signal, $T_{s,ev}$ will be driven to a value well above or below the actual supply air temperature T_s. Thus, the primary symptom of this fault is seen in the value of R_{Ts}.

The second fault is a fouled cooling coil. Deposition of dirt and scale on a coil surface can increase the resistance to heat transfer. This thermal resistance can be represented by a fouling factor. Because fouling occurs gradually, performance changes are difficult to detect for some time. Rather than changing the fouling factors gradually, a step change in the values was introduced to expedite data collection. Specifically, the air-side fouling resistance r_a was changed from 0 to 0.4 $m^2 \cdot K/kW$ (0.00227 $h \cdot ft^2 \cdot °F/Btu$), and the water-side fouling resistance r_w was changed from 0 to 0.5 $m^2 \cdot K/kW$ (0.00284 $h \cdot ft^2 \cdot °F/Btu$). When the cooling coil is fouled, the cooling coil control signal increases in order to increase the

flow rate of water, thereby achieving the heat transfer necessary to maintain the supply airflow rate at the set point. Because the expected value of the supply air temperature is a function of the control signal to the cooling coil valve, $T_{s,ev}$ decreases relative to T_s. Thus, R_{T_S} is once again the main feature of this fault.

The third fault is a leak in the heating coil valve. In the presence of this fault, the air temperature at the exit of the heating coil is higher than the mixed air temperature, and the cooling coil valve control signal must increase to compensate for the fault. Because the expected value of the supply air temperature is a function of the mixed air temperature and not the temperature at the exit of the heating coil, the value of $T_{s,ev}$ will be lower than the actual supply air temperature. Thus, R_{T_S} is also the main feature of this fault.

The fourth fault is a stuck VAV box damper. The fault is simulated by causing the damper to stick in the position it occupies at $t = 10{,}000$ s. The fault causes the room temperature to drift away from the setpoint value, causing R_{T_r} to increase or decrease, depending on the room load.

The fifth fault is a performance degradation of the supply fan that might be caused by belt slippage or a decrease in the motor efficiency. For a fixed control signal to the fan, the fault causes the rotational speed of the fan to decrease relative to its value for normal operation. To maintain the supply air static pressure at the setpoint value, the control signal to the supply fan must be increased. Because the expected value of the supply air pressure $P_{s,ev}$ is a function of the control signal to the supply fan, $P_{s,ev}$ will be greater than P_s in the presence of this fault. Thus, R_{P_S} is the main feature of this fault. To expedite the collection of data, a step reduction in the rotational speed of 2% for a given control signal was introduced at $t = 0$.

The sixth fault is a failure of the return fan controller. The fault causes the fan to stick at a fixed speed at $t = 10{,}000$ s. Because the fan is unable to respond to the control signal, the flow difference between the supply and return airstreams will either increase or decrease depending on the operation of the supply fan. Thus, R_{Qd} is the main feature of this fault.

The seventh fault is a failure of a linkage in the mixing box dampers. This fault affects the airflow rates in the mixing box and, therefore, the expected value of the mixed air temperature $T_{m,ev}$ is different from the actual value T_m. Thus, R_{Tm} is the main feature of this fault.

c-MEANS CLUSTERING METHODS

Clustering algorithms are typically used to assign unlabeled data (data for which the class of operation is unknown) to one of c classes, where c is two or more. Data points clustered using a hard c-means clustering algorithm (Bezdek 1981) have crisp membership functions. That is, if data are clustered into two classes, each data point will have a membership of unity in one class and zero in the other. Furthermore, each of the c classes can be represented by a single prototype data point, which is typically just the mean value of all members of the class.

Unlabeled data can also be clustered using a fuzzy c-means clustering algorithm (Bezdek 1981). Data clustered using this algorithm have fuzzy membership functions. In this case, each data point will have membership ranging from zero to unity in each class, with the sum of the membership values for a given data point being unity and the sum of the membership values for a given class being greater than zero and less than the total number of clustered points n. Data points that fall midway between the cluster centers tend to have nearly equal membership in each class. The fuzzy c-means algorithm yields prototype data points for each of the c classes; however, the prototype data points are weighted mean values of the class members. This point will be further developed later in this section. The membership functions obtained from the hard c-means and fuzzy c-means clustering algorithms can be used in nearest neighbor algorithms to classify new data points. Likewise, the c prototype data points can be used in nearest prototype algorithms. The membership functions and prototypes can also be used to train an artificial neural network (ANN) algorithm. These algorithms are described in the next section.

In this study, the training data are labeled, that is, the class of operation of each data point is known a priori. Thus, the utility of a clustering algorithm may be questioned. It is true that crisp labels can be assigned to each data point without any analysis and that prototype data points for each class can be determined by simply computing the mean value of the members of each class. However, if fuzzy membership functions are to be assigned to each data point, it is necessary to use a fuzzy c-means algorithm to determine these membership functions. Fuzzy membership functions may be desirable if there is significant overlap of the data between classes. In this scenario, it may be more meaningful to assign the data point to partial membership in more than one class than to assign it to complete membership in a single class.

Fuzzy membership functions and prototype data points are determined by minimizing the objective function J_m (Bezdek 1981):

$$J_m(U,V) = \sum_{j=1}^{n} \sum_{i=1}^{c} \left(u_{ij}\right)^m \left\| x_j - v_i \right\|^2 \qquad (7)$$

where x_j is the jth q-dimensional training data point (for data characterized by five residual values, $q = 5$), v_i is the ith q-dimensional fuzzy prototype data point, u_{ij} is the membership of the jth training data point in the ith class, U is a matrix of membership functions containing a c-dimensional membership function for each of the n clustered points, V is a matrix containing a q-dimensional fuzzy prototype data point for each of the c classes, m is any real value greater than or equal to unity, and $\|\cdot\|$ is any inner product-induced norm on q-dimensional real space. In this study, $\|\cdot\|$ is the 2-norm, defined by

$$\left\| x_j - v_i \right\| = \sqrt{\sum_{\ell=1}^{q} (x_{j\ell} - v_{i\ell})^2} \; . \qquad (8)$$

From Equation 7 it is clear that the larger the distance between x_j and v_i, the smaller the membership function u_{ij}. Also, if $m = 1$, J_m simplifies to the hard c-means objective function. Finally, as m increases, J_m becomes insensitive to distance and membership functions become more "fuzzy" (i.e., for $c = 3$, membership in each class approaches 0.333 for all data points).

CLASSIFICATION METHODS

k-Nearest Neighbor Classifier

Nearest neighbor classification (Bezdek 1981; Schalkoff 1992) is based on the premise that data from a specific class of operation should fall within the same region of feature (residual) space. Using training data that has been assigned either a crisp or a fuzzy membership label (or membership function), the distance from a test data point to each training data point is determined. The training data points are then sorted based on their distance to the test point. Classification of the test point is performed by computing the average membership function of the k-nearest neighbors (k training data points closest to the test point) in the training data set and assigning the test point to the class having the largest average membership function value.

The k-nearest neighbor (k-NN) method is a conceptually straightforward method that can be used effectively for pattern recognition. The main drawbacks of the approach are related to training data storage and computational efficiency. Larger training data sets result in a better classifier; however, storage and computational problems are exacerbated.

k-Nearest Prototype Classifier

Nearest prototype classifiers (Bezdek 1981) are similar to k-NN classifiers in the sense that a measure of distance is once again used to assign a test data point to a class of operation. The main difference in the k-nearest prototype (k-NP) and k-NN classifiers is that the training data used in the k-NN method are replaced by prototype data points representing each of the classes of operation. For 1-NP classification, a test point is assigned to the class of operation of the closest prototype data point. This can be extended to a more general case of multiple prototypes for each class of operation. For 1-NP classification, prototype data points have membership in a single class of operation (crisp membership). For k-NP classification, prototype data points can have either crisp or fuzzy membership functions.

The 1-NP classification method alleviates the computational difficulties that limit the application of the k-NN method; however, it seems obvious that some information contained in the training data set is lost with this technique. If significant overlap in the classes occurs, a data point would seem more likely to be improperly classified by the 1-NP method than the k-NN method. Thus, k-NP classification may represent a reasonable compromise between the two methods; however, the results presented in this paper were obtained using a 1-NP method.

Artificial Neural Network Classifier

Artificial neural networks (ANNs) are powerful tools for mapping inputs to outputs, especially when this mapping is nonlinear. For classification purposes, a feedforward ANN can be trained to produce a specific output pattern for a specific input pattern. Lee et al. (1996, 1997) used patterns of residuals as inputs to an ANN classifier and crisp membership functions representing various classes of operation as outputs. Lee et al. (1996) trained an ANN to perform classification using a single pattern for each of the modes of operation considered. Lee et al. (1997) augmented the prototype patterns with patterns generated by the addition of noise.

The approach of Lee et al. (1996) can be followed using either crisp or fuzzy prototype data points as inputs and crisp membership functions as outputs. Alternatively, training data can be input directly to an ANN and the output can be crisp or fuzzy membership functions. Fuzzy prototypes and membership functions would again come from a fuzzy c-means algorithm.

ANNs are effective tools for pattern recognition. In addition, feedforward ANNs are computationally efficient and require little memory. However, ANNs do not extrapolate well, meaning that input patterns unlike those used for training may produce unreasonable outputs. In addition, because ANNs are black boxes, the reasoning behind decisions may be difficult to understand.

Rule-Based Classifier

A rule-based algorithm employing thresholds can also be an effective classifier. IF-THEN type rule-based algorithms are attractive when the patterns representative of a particular class of operation can be easily identified. Otherwise, the complexity of the rules increases and implementation in an algorithm is more difficult. In this study, the faults tend to have a single dominant symptom. This makes it a simple matter to define classifying rules.

Bayes Classifier

A Bayes classifier minimizes the cost or the probability of misclassification (Fukunaga 1990). Thus, in this sense, it is an optimal classifier. Consider the problem of classifying an observation x_o to one of c classes. The cost of misclassification is minimized if x_o is allocated to the class k that minimizes the following expression (Johnson and Wichern 1992):

$$\sum_{\substack{i=1 \\ i \neq k}}^{c} p_i f_i(x_o)\, C(k \mid i) \tag{9}$$

where p_i is the a priori probability of an observation coming from class i, f_i is the conditional density function for class i, and $C(k|i)$ is the cost associated with allocating an observation

to class k, when in fact it comes from class i. If the cost of misclassification $C(k|i)$ is the same for all i and k, the decision rule states that x_o should be allocated to the class k that minimizes

$$\sum_{\substack{i=1 \\ i \neq k}}^{c} p_i f_i(x_o) \quad . \tag{10}$$

This expression is minimum when $p_k f_k(x_o)$ is maximum. Thus, the Bayes decision rule states,

allocate x_o to class k if

$$p_k f_k(x_o) > p_i f_i(x_o) \quad \text{for all } i \neq k . \tag{11}$$

A completely equivalent expression of Equation 11 is

allocate x_o to class k if

$$\ln p_k f_k(x_o) > \ln p_i f_i(x_o) \quad \text{for all } i \neq k . \tag{12}$$

If the data in each class can be represented by a multivariate normal density function,

$$f_i(x) = \frac{1}{(2\pi)^{q/2} \left|\Sigma_i\right|^{1/2}} \exp\left[-\frac{1}{2}(x - \mu_i)' \Sigma_i^{-1} (x - \mu_i)\right], \tag{13}$$

where μ_i is the mean vector and Σ_i is the covariance matrix, Equation 12 then becomes the following:

allocate x_o to class k if

$$d_k^Q(x_o) = \text{largest of } d_1^Q(x_o), d_2^Q(x_o), \ldots, d_c^Q(x_o) \tag{14}$$

where d_k^Q denotes the quadratic discrimination score (or quadratic score) for class k given by

$$d_k^Q(x_o) = -\frac{1}{2}\ln\left|\Sigma_k\right| - \frac{1}{2}(x_o - \mu_k)' \Sigma_k^{-1} (x_o - \mu_k) + \ln p_k - \frac{q}{2}\ln(2\pi). \tag{15}$$

The final term in Equation 15 is the same for all classes and can be dropped. The quadratic score for class i can be approximated by replacing the population mean μ_i and covariance matrix Σ_i with the sample mean $\bar{x}_i$ and covariance matrix S_i.

Equation 15 can be simplified for the special case where all the population covariance matrices are equal (or are taken to be equal). In this case, Equation 15 simplifies to

$$d_k^Q(x_o) = -\frac{1}{2}(x_o - \mu_k)' \Sigma^{-1} (x_o - \mu_k) + \ln p_k . \tag{16}$$

The first and last terms in Equation 15 have been dropped in Equation 16 since they are the same for all classes. The quadratic score of class i can be approximated by replacing the population statistics (μ_i and Σ) with the sample mean $\bar{x}_i$ and a pooled estimate S_{pooled} of Σ given by

$$S_{pooled} = \frac{(n_1 - 1) S_1 + (n_2 - 1) S_2 + \ldots + (n_c - 1) S_c}{n_1 + n_2 + \ldots + n_c - c} \tag{17}$$

where n_i is the sample size for class i (Johnson and Wichern 1992). Finally, if S_{pooled} is equal to the identity matrix and all classes have equal prior probabilities, the allocation rule simplifies to that of the nearest prototype classifier.

From a standpoint of minimizing classification errors, the Bayes classifier is optimal. In addition, it is not overly complex from either a computational or a conceptual point of view. However, its performance is sensitive to departures from normality in the data.

FAULT DETECTION

Clustering

One of the first tasks associated with developing a training data set for fault detection was to decide whether the data should be clustered into two classes (normal and faulty) or three classes (normal, faulty, and unknown). To this end, 700 data points were drawn randomly from the data set representing normal operation, and 100 data points were drawn randomly from each of seven fault data sets (all training data come from the first day of the three-day simulation). Using only the two largest residuals for each training point, the training data were clustered in two classes using the fuzzy c-means algorithm (Demuth and Beale 1992). The clustering results are shown in Figure 2a. It is important to understand that the two largest residuals are not necessarily the same physical features for all data points. Selecting the two largest residuals simply reduces the dimension of the fault detection clustering and classification problems from five residuals to two.

The main feature distinguishing the data points in the two classes is the largest residual; however, the classes are not well separated. A considerable number (112) of training data points have the largest residual in the range from 0.3 to 0.7; however, only 47 data points have fuzzy membership function values between 0.3 and 0.7. This implies that the membership functions are somewhat crisp and the transition from the normal class to the faulty class is rather abrupt.

Note that many data points have the largest residual in this ambiguous middle range from 0.3 to 0.7. This characteristic of the data is easily explained. Although all training data were taken from data sets for which the status of operation was known to be either normal or faulty, the residuals at any given time were not always consistent with what was expected for a particular operating status. For instance, faulty operation due to a stuck valve may look like normal operation if the position at which the valve sticks happens to satisfy the current load on the system. The opposite situation, where data obtained at normal operating conditions have residuals that indicate the presence of a fault, was essentially eliminated through the use of the steady-state detector.

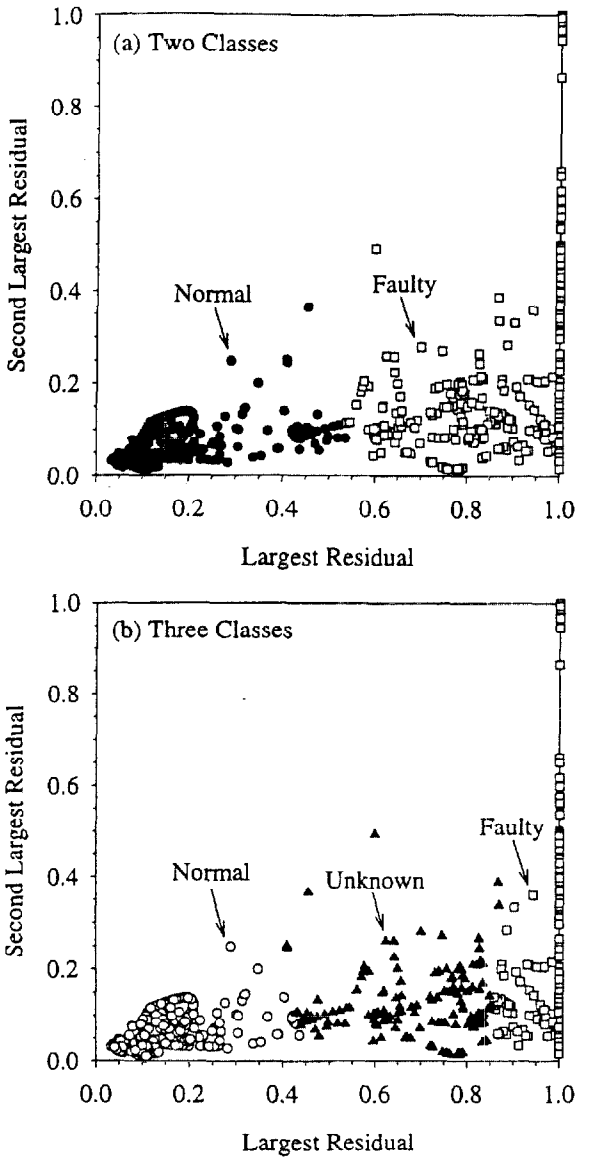

Figure 2 Results of clustering the training data.

Because the training data do not separate into two well-defined classes and because the fuzzy membership functions do not adequately reflect this characteristic, the clustering exercise was repeated using $c = 3$. The labeled training data for this case are shown in Figure 2b. Now the normal and faulty classes are well separated. In addition, 204 of the 1400 training data points have at least one membership function value in the range from 0.3 to 0.7. Most of these data points have two membership function values in this range. Thus, the labeling of the data more accurately reflects the overlapping nature of the classes.

The use of the unknown class has two effects. First, it is less likely that a false diagnosis will be made (faulty data labeled normal and vice versa), and second, it is more likely that faulty and normal operation will be labeled as unknown operation. The first effect is desirable, while the second is tolerable. Thus, for fault detection, the training data clustered into three classes will be used as the basis for classification of the testing data. It could be argued that the unknown classification is artificial because the status of operation can only be normal or faulty. However, if the features of the current operation are ambiguous, then so is the operating status.

Training

The output of the clustering algorithm included fuzzy membership functions for the training data set and fuzzy prototypes for each of the three classes of operation. Crisp membership functions were also assigned to each data point in the training data set, and crisp prototype data points were determined by computing the mean value of the members of each class of operation.

The ANN classifier that used only the crisp or fuzzy prototypes as inputs and crisp memberships as outputs (Lee et al. 1996) was not considered. Instead, the ANNs used the training data residuals as inputs and crisp or fuzzy memberships as outputs (1400 input/output pairs). The ANNs were multilayer feedforward networks with one hidden layer containing four neurons and were trained using a back-propagation learning rule until the sum of squares error was less than 1.0. A commercial ANN software package was used (Demuth and Beale 1992).

The thresholds used in the rule-based method were mean values of the largest residual of the prototypes of neighboring classes. For instance, the threshold between normal and unknown was the mean value of the largest residual of the prototypes for these classes. If the largest residual of a test point was less than this threshold, the test point was assigned to the normal class.

To implement the Bayes classifier, the mean value and covariance matrix for each class of operation were required. Only the allocation rule given by Equations 14 and 15 was considered, with population statistics replaced with sample statistics from the training data. Furthermore, prior probabilities were taken to be equal, so the $\ln p_k$ term in Equation 15 was the same for each class. Finally, the cost of misclassification $C(k|i)$ was taken to be equal for all misclassifications.

No additional training beyond the clustering analysis was necessary for the other classification techniques. It is important to note that the need for training data is a limiting characteristic of all of the considered classifiers, although it is less of an issue for the rule-based classifier. Training data for various fault modes of operation may be difficult to obtain, and it may be difficult to generalize the operating characteristics (residual patterns) of a particular system to others. The approach described in this study is expected to be more applicable to packaged equipment that is factory built than to customized equipment such as built-up AHUs.

Scoring Method

The scoring method defines how the classifiers are evaluated. In this study, a diagnosis of "unknown" is included as

a correct output of the classifiers, whether the data come from a normal data set or a faulty data set. The conclusions reached in ensuing sections of this paper are sharply dependent on this choice of scoring method; however, sufficient detail is provided in the results to allow the readers to draw their own conclusions based on the scoring method of their choice.

Results

Figure 3 shows the fault detection decision boundaries and prototypes (shaded circles) for several classifiers. Boundaries labeled CANN refer to an ANN trained with 1400 training pairs, where the inputs are the two largest residuals for each training data point and the outputs are the crisp membership labels. Boundaries labeled CNP refer to the 1-nearest prototype classifier with crisp prototypes. Those labeled CkNN and FkNN refer to the k-nearest neighbor classifier with crisp and fuzzy membership functions, respectively. For all nearest neighbor results, $k = 5$. Boundaries labeled BAYES and RULES refer to the Bayes and rule-based classifiers, respectively. The boundaries were determined by classifying data that spanned the input space defined by $0 \le x_1 \le 1$ and $x_2 \le x_1$, where x_1 is the largest residual and x_2 is the second largest residual. No data can lie above the diagonal line from (0,0) to (1,1) since this would imply that the second largest residual is greater than the largest residual. Note that when the second largest residual is less than 0.25, there is little difference in the classifier boundaries. The exception to this statement is the BAYES classifier. Although not shown, the boundaries for the nearest prototype classifier with fuzzy prototypes and for the ANN trained with fuzzy membership labels are nearly the same as those of the CANN classifier.

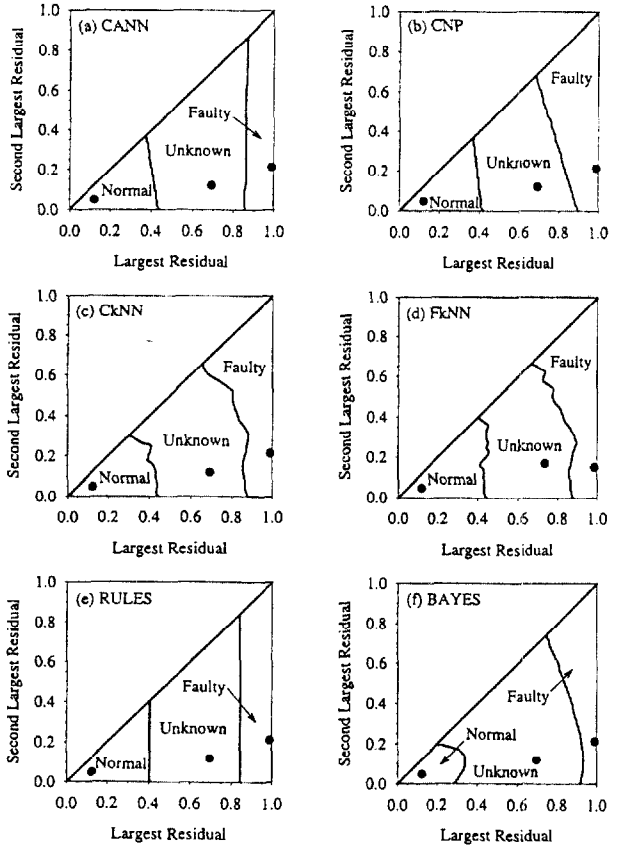

Figure 3 *Fault detection decision boundaries for various classifiers.*

Fault detection results for normal and seven fault modes of operation are presented for six classifiers in Figure 4. The third day of the three-day simulation data was used to produce the results in this section. The shaded area of the bars indicate diagnoses of "normal" for the normal operation case and "faulty" for the seven fault cases. The unshaded area of the bars represent the "unknown" diagnoses. Recall that an "unknown" diagnosis is taken to be a correct diagnosis.

The most striking feature of the results in Figure 4 is the fact that, with the exception of the Bayes classifier, there is very little difference in the performance of the classifiers for a given mode of operation. For instance, each method correctly classified approximately 99% of the data obtained under normal operating conditions. Of these data, 95% to 97% were labeled "normal" and an additional 2% to 4% were labeled "unknown."

The results for the seven faults clearly demonstrate the difference between the Bayes classifier and the other classifiers. For each of the faults, the Bayes classifier has the largest percentage of correct diagnoses; however, it also has the largest percentage of "unknown" diagnoses and the smallest

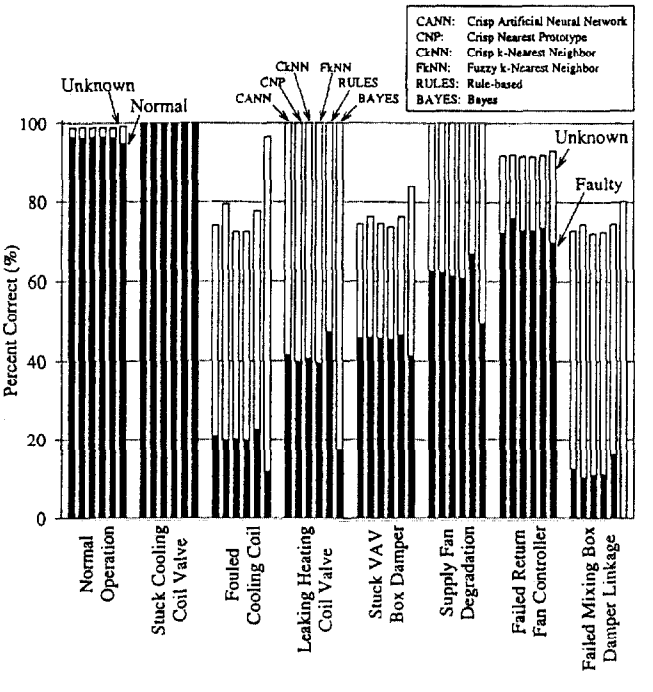

Figure 4 *Fault detection results (Normal Operation: Percent Correct = Normal + Unknown; Faulty Operation: Percent Correct = Faulty + Unknown).*

percentage of "faulty" diagnoses for all but the stuck cooling coil valve fault. For this fault, all methods correctly diagnosed the operation as "faulty" 100% of the time. Results in Figure 4 are consistent with the locations of the classifier decision boundaries.

For the most part, the results in Figure 4 exclude the classifiers that utilize fuzzy membership functions and/or prototypes. This was done to simplify the discussion of the results and is not an indication that classifiers using crisp membership functions and/or prototypes are superior to their fuzzy counterparts. Figure 4 shows that the CkNN and FkNN classifiers yield almost identical results for the data considered. To understand why, consider the training data in Figure 2b. Since the five nearest neighbors are determined by distance, differences in classification must be due to differences in membership functions. The crisp and fuzzy membership functions differ most near class boundaries. Thus, the CkNN and FkNN classifiers differ only near these boundaries. Figures 3c and 3d illustrate the similarity of the classifiers for these training data.

The results presented in Figure 4 correspond to steady-state operation. It is of interest to examine the influence of the steady-state detector thresholds on the performance of the classifiers. Since the residual patterns for each mode of operation were determined with the assumption of steady-state operation, classifier performance is expected to improve as the thresholds on the steady-state detector are tightened. Results of applying different steady-state detector thresholds are shown in Figure 5 for the fouled cooling coil fault and for the stuck VAV box damper fault. For each fault, results obtained with three different sets of steady-state detector slopes are presented. The total number of steady-state points in each set of results is given in Figure 5. The results indicate only a small improvement in classifier performance as the thresholds on the slopes are tightened. Furthermore, the order of the classifiers based on percentage of correct diagnoses does not appear to change. Hence, although the steady-state detector is an important aspect of the fault detection approach used here, it does not account for similarities or differences seen when comparing the results of different classifiers.

Sensitivity of the classifier to fault severity and load conditions should be considered for degradation faults. Figure 6 compares classifier performance for the fouled cooling coil fault with various fouling levels. Fouled coil results in Figure 4 and the associated training data correspond to r_a = 0.4 m^2·K/kW (0.00227 h·ft^2·°F/Btu) and r_w = 0.5 m^2·K/kW (0.00284 h·ft^2·°F/Btu).

For the considered range of r_a, no change in classifier performance is observed. However, as r_w increases, the performance of each of the classifiers improves dramatically. Note that although the number of "faulty" diagnoses is very small for levels of fouling below those used in training, 44% to 69% of the data is assigned to the "unknown" class for r_w = 0.35 m^2·K/kW (0.00199 h·ft^2·°F/Btu). This indicates that the fault symptoms can be detected at lower levels of fouling than were present in the training data. The fact that the BAYES classifier has the highest percentage of correct diagnoses is not surprising, since this classifier has the largest unknown class and it extends well into the normal region of the other classifiers.

The influence of load conditions on classifier performance was not thoroughly investigated. Training and testing data for the fouled cooling coil fault corresponded to coil loading of 50% to 75%. The ability of the classifier techniques to

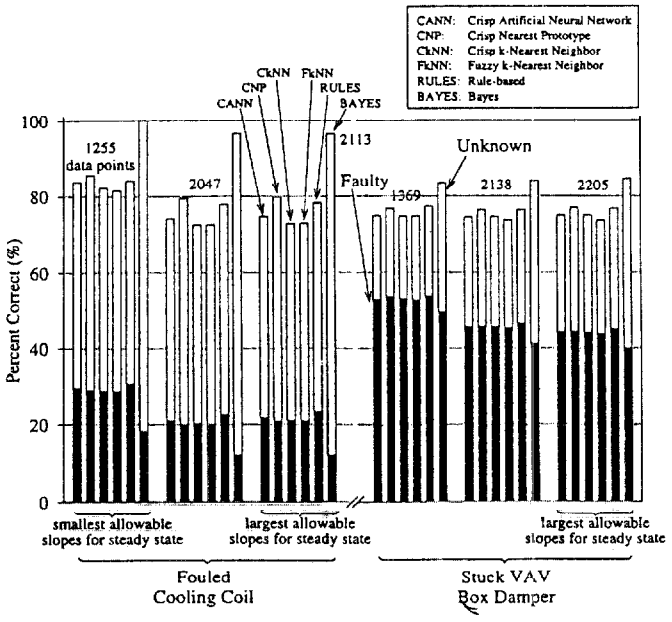

Figure 5 *Influence of the steady-state detector on fault detection results for two faults (Percent Correct = Faulty + Unknown).*

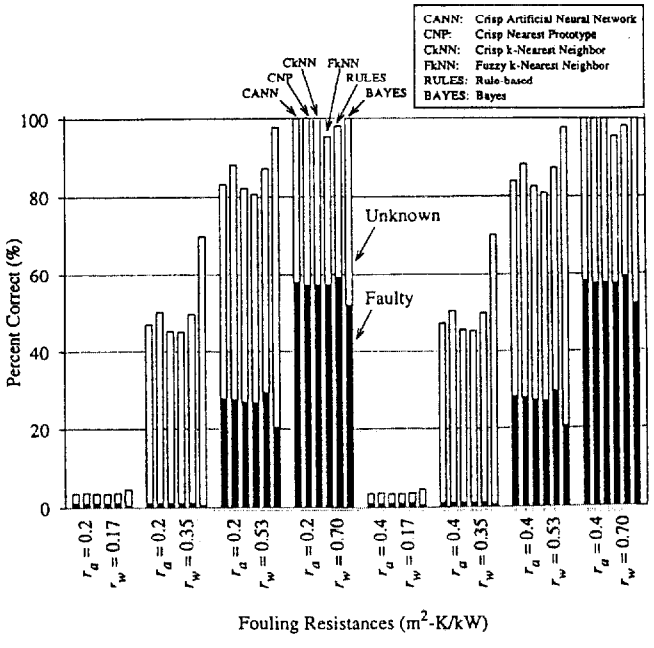

Figure 6 *Fault detection results for different levels of cooling coil fouling (Percent Correct = Faulty + Unknown).*

detect this fault at lower coil loading is highly dependent on the model used to predict T_s. In general, the fault symptoms are expected to be less apparent at lower coil loading, so the performance of all the classifiers would suffer.

FAULT DIAGNOSIS

Training

Training for the fault diagnosis problem is similar to that for the fault detection problem, except that all five residuals are employed. In addition, the training data consist only of data points from the first weather day that were classified as faulty by the rule-based classifier. The objective is to classify the data determined to be faulty into one of five classes ($c = 5$) corresponding to faults associated with (1) control of the supply air temperature, (2) control of the mixed air temperature, (3) control of the room air temperature, (4) control of the supply air pressure, and (5) control of the difference between the supply and return airflow rates. An unknown class is not used for fault diagnosis since the training data separate well into five classes. The introduction of a sixth class tends to divide the cluster corresponding to faults with the control of the supply air temperature into two clusters. Hence, only five classes were used.

The five classes listed above typically have one dominant residual. Thus, thresholds in the rule-based method were specified such that if the largest residual was the normalized value of R_{Ts}, the data point was assigned to the supply air temperature fault class. Similar assignments were made for the other residuals.

Results

Fault diagnosis results for numerous classifiers are presented in Figure 7. Note that the data considered for fault diagnosis are restricted to data points from the third weather day that were classified as faulty by the rule-based classifier. The results in Figure 7 demonstrate that for six of the faults (all but the stuck cooling coil valve), there is very little difference in the performance of the classifiers. For those six faults, the performance of all classifiers was very good.

The classifiers correctly diagnosed the stuck cooling coil valve fault approximately 55% (BAYES) to 78% (RULES) of the time. Dominant normalized residual values for this fault are plotted in Figure 8. Note that at certain times the normalized values of R_{Ts} and R_{Tr} are simultaneously equal to unity, and at other times normalized values of R_{Ts} and R_{Qd} are simultaneously equal to unity. A large percentage of the incorrect diagnoses are data points that are believed to be faults associated with the control of the room air temperature. In addition, for the RULES and BAYES classifiers, a small number of misclassified data points are classified as faults associated with the control of the airflow difference. The correct diagnosis is that all data points come from a fault associated with the control of the supply air temperature.

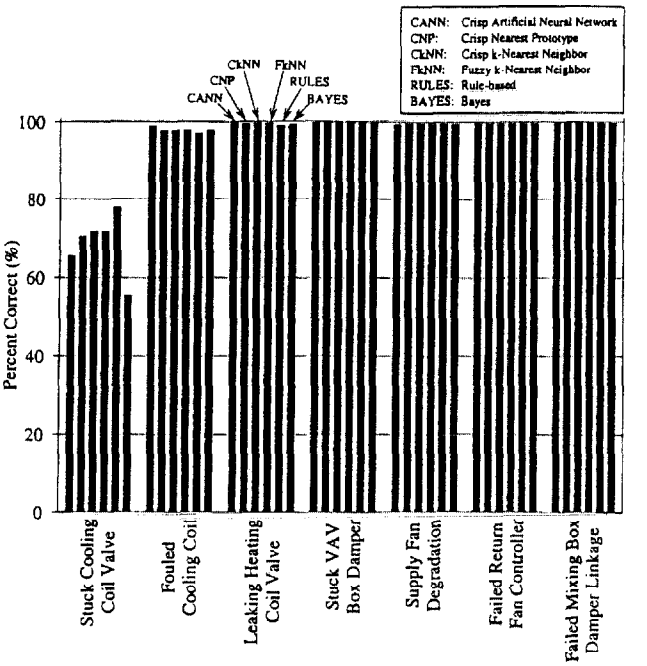

Figure 7 Fault diagnosis results.

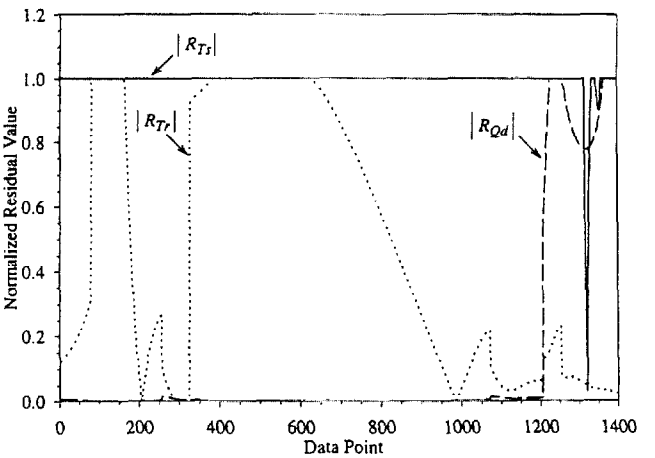

Figure 8 Dominant normalized residual values for the stuck cooling coil valve fault.

The rules used in the RULES classifier are not sophisticated enough to handle the case where multiple normalized residuals are saturated at the maximum value. However, for classification problems such as this, where the classes of operation are well separated and can typically be characterized by one dominant feature, a simple rule-based classifier is expected to be effective.

CONCLUSIONS

The objective of this study was to demonstrate the application of several classification techniques to the problem of detecting and diagnosing faults in data generated by a VAV

AHU simulation model and to describe strengths and weaknesses of the techniques considered. Artificial neural network classifiers, nearest neighbor classifiers, nearest prototype classifiers, a rule-based classifier, and a Bayes classifier were compared for both fault detection and fault diagnosis.

For the faults considered, classifier performance was nearly equal. Because each classifier uses a measure of distance either directly or indirectly, this result is perhaps not surprising. Based on the performance of the methods and the scoring method selected, the Bayes classifier is a good choice for fault detection. It is a straightforward method that requires limited storage and computational effort. In addition, for the faults considered, the Bayes classifier has the lowest percentage of incorrect diagnoses. For fault diagnosis, the rule-based method is favored for classification problems such as this, where the various classes of faulty operation are well separated and can be distinguished by a single dominant symptom or feature.

These conclusions are drawn from a single study that is by no means exhaustive in terms of the number and type of faults considered, training schemes, or classifiers. Nonetheless, the characterization of the strengths and weaknesses of the classifiers is intended to be unbiased and should be useful to researchers and practitioners who are interested in further studying one or more of the classifiers. Furthermore, the findings of this study are supported by Bezdek (1993), who provides numerous explanations of the asymptotic equivalence of several of the classifiers considered here. This work points out that the success or failure of classifiers hinges to a large degree on an ability to separate the different classes in some feature space. Hence, preprocessing of data to extract dominant features is as important as the selection of the classifier.

ACKNOWLEDGMENTS

This research was partially supported by the Office of Energy Efficiency and Renewable Energy, U.S. Department of Energy, and the Ministry of Science and Technology in Korea.

REFERENCES

Anderson, D., L. Graves, W. Reinert, J.F. Kreider, J. Dow, and H. Wubbena. 1989. A quasi-real-time expert system for commercial building HVAC diagnostics. *ASHRAE Transactions* 95(2): 954-960.

Bagby, D.G, and R.A. Cormier. 1989. A heat exchanger expert system. *ASHRAE Transactions* 95(2): 927-933.

Bezdek, J.C. 1981. *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum Press.

Bezdek, J.C. 1993. A review of probabilistic, fuzzy, and neural models for pattern recognition. *Journal of Intelligent and Fuzzy Systems* 1(1): 1-25.

Demuth, H., and M. Beale. 1992. *Neural network toolbox: User's guide*. Natick, Mass.: The MathWorks, Inc.

Dexter, A.L., and M. Benouarets. 1996. A generic approach to identifying faults in HVAC plants. *ASHRAE Transactions* 102(1): 550-556.

Dodier, R.H., P.S. Curtiss, and J.F. Kreider. 1998. Small-scale on-line diagnostics for an HVAC system. *ASHRAE Transactions* 104(1).

Fukunaga, K. 1990. *Introduction to statistical pattern recognition*, 2d ed. San Diego, Calif.: Academic Press.

Haberl, J.S., and D.E. Claridge. 1987. An expert system for building energy consumption analysis: Prototype results. *ASHRAE Transactions* 93(1): 979-998.

Haves, P., T.I. Salsbury, and J.A. Wright. 1996. Condition monitoring in HVAC subsystems using first principles models. *ASHRAE Transactions* 102(1): 519-527.

Johnson, R.A., and D.W. Wichern. 1992. Applied multivariate statistical analysis, 3d ed. Upper Saddle River, New Jersey: Prentice Hall.

Kaler, G.M. 1990. Embedded expert system development for monitoring packaged HVAC equipment. *ASHRAE Transactions* 96(2): 733-742.

Lee, W.Y., J.M. House, C. Park, and G.E. Kelly. 1996. Fault diagnosis of an air-handling unit using artificial neural networks. *ASHRAE Transactions* 102(1): 540-549.

Lee, W.Y., J.M. House, and D.R. Shin. 1997. Fault diagnosis and temperature sensor recovery for an air-handling unit. *ASHRAE Transactions* 103(1): 621-633.

Li, X., H. Vaezi-Nejad, and J.C. Visier. 1996. Development of a fault diagnosis method for heating systems using neural networks. *ASHRAE Transactions* 102(1): 607-614.

Pape, F.L.F., J.W. Mitchell, and W.A. Beckman. 1991. Optimal control and fault detection in heating, ventilating, and air-conditioning systems. *ASHRAE Transactions* 97(1): 729-736.

Pernot, O., P. Coralles, X. Li, and J.C. Visier. 1997. Fault detection and diagnosis for heating systems: Artificial neural network versus threshold. Annex 34 Working Paper, Montreal.

Rossi, T.M., and J.E. Braun. 1997. A statistical, rule-based fault detection and diagnostic method for vapor compression air conditioners. *International Journal of HVAC&R Research* 3(1): 19-37.

Schalkoff, R.J. 1992. *Pattern recognition: Statistical, structural, and neural approaches*. New York: John Wiley & Sons.

Wang, S. 1992. Modeling and simulation of building and HVAC systems—Building and HVAC system and component models used in emulation exercise C.3. IEA ANNEX 17 working paper.