A Probability Extension of PCA to Detect and Diagnose Sensor Faults in Air Handling Units

Zhengwei Li
Ph.D Student
Georgia Institute of Technology
Atlanta, GA

Christiaan J. J. Paredis
Professor
Georgia Institute of Technology
Atlanta, GA

Godfried Augenbroe
Professor
Georgia Institute of Technology
Atlanta, GA

ABSTRACT

Due to sensor faults, it is a challenge to successfully detect and diagnose component faults in HVAC systems. The Principal Component Analysis (PCA) method has become a popular method to tackle this problem in recent years, but PCA is not capable of isolating sensor faults, such as sensor bias or sensor noise. The intention of this paper is to take sensor noise into account. This is accomplished by including sensor noise and sensor drift into a Bayesian probability calculation framework. In this approach, both of these potential faults are associated with a probability score once the system detects a fault. Component faults are not taken into account because we assume the PCA method is only the first step in detecting and diagnosing faults in HVAC systems. The sensor location effect has already been eliminated in the training process, so it is not considered either. This paper firstly discusses the drawbacks of applying traditional PCA method. For instance, we show that a sensor drift fault diagnosed by this method could actually be caused by sensor noise instead. The second part of the paper shows that by applying Bayesian probability calculations within the PCA analysis process, the false alarms caused by sensor noise can be partially excluded.

Keywords: AHU, Fault Detection and Diagnostics (FDD), sensor, probability

1. LITERATURE REVIEW

Correctly working sensors are the key in successfully detecting faults in HVAC systems, however, sensor can fail as well. Dexter and Pakanen [1] divided sensor faults into three categories: location faults – wrongly placed, electrical installation faults – bad joints, incorrect power supply, etc. and sensor related faults – drift, no signal, etc.

Among all the sensor faults, sensor drift and sensor bias are among the mostly difficult detectable faults, and have adverse impact on the system operational efficiency. Therefore, many researchers have contributed in detecting sensor faults, which can be categorized based on if the sensor faults are detected separately from component faults ([2, 3, 4, 5, 6, 7]) or together with component faults([9, 10]).

These work can also be categorized based on the FDD methods. While a model based method is quite popular in HVAC component fault detection, it is rarely used to detect sensor faults. Among all the researches reviewed, only Wang and Wang [6] used this method. Rule based methods have been used by researchers for more than ten years. One of the pioneer work done by Yang and Jiang [2] compares the neighboring sensors, finding out the faulty sensor by checking through all the sensors against physical constraints. This method is not practical because it needs more sensors than necessary for control purpose. Schein et al. [9] developed a rule based method (APAR) to detect common faults in air handling units, including some sensor faults. Yang et al. [3] developed a set of rules to detect faults for only temperature sensors in air handling units.

The third category of method is machine learning. This method just emerged in recent ten years, but has quickly been adopted and used by many researchers. Lee et al. [10] used Artificial Neural Network (ANN) to detect sensor faults in AHU, Hou et al. [5] combined Rough Set (RN) with ANN detect sensor faults. Principal Component Analysis (PCA) is a new method emerged in 2004 [7], and since then developed by Wang and Qin [11], Xiao et al. [12], Du and Jin [4,13].

PCA has been suggested as a quick and effective method in detecting sensors in air handling units [7]. It has certain advantages compared to other methods. It is not as computationally intensive as ANN, it is effective towards more sensors in the system as opposed to rule based method, it can separate component faults from sensor faults to some extent, and its results are reasonably good. However, the current analysis method to PCA results (Q-statistics, [7]) does not take into account the possibility of sensor noise, therefore, the false alarm rate could be relatively high.

The initiative of this work is to explicitly take sensor noise into the fault diagnosis calculation algorithm, so that the diagnosis results have fewer false alarms. Although there are filters (such as exponential moving average filter) that can decrease the impact of random noise, and they are effective in negating the adversary effects of sensor noise, but the function they provide in practice are fuzzy, because the extent to which sensor data gets disturbed and smoothed by the filter are left unknown. Besides sensor noise, sensor location and component faults can also affect the sensor value. But sensor location effect has already been eliminated in the training process, component faults are not taken into account because we assume PCA analysis is just the first step in detecting and diagnosing faults in AHU.

2. INTRODUCTION TO PCA

Principal Component Analysis (PCA) method is a multivariate statistical analysis tool that can be used to reduce interdependent variables, so that the independent variables – principal components (PC) can be found and conserved. Since it appeared in Pearson [14], it has been used as a data analysis tool in bioinformatics [15], artificial intelligence [16], and HVAC system [7].

Suppose we have an original matrix x ($x \in \mathrm{R}^{m\times n}$) ($m$ is the number of data points n is the number of variables of each data). After normalizing x to x_n, the interdependence between all the variables can be found by calculating covariance matrix Σ ($\Sigma \in \mathrm{R}^{m\times n}$).

$$\Sigma = \frac{{x_n}^T x_n}{n-1} \qquad (1)$$

The loading matrix U ($U \in \mathrm{R}^{n\times k}$, $k<n$) is then composed by the eigenvectors corresponding to the k largest eigenvalues of Σ. When a new sample of data X_{new} comes, its principal component is calculated using the following equation

$$y_{new} = U\, U^T X_{new} \qquad (2)$$

The error between new data and its principal component is based on

$$e_{new} = \|y_{new} - X_{new}\|^2 \qquad (3)$$

The sum of e_{new} for all features is the indicator as fault, each individual e_{new} is the indicator for that specific sensor, which is proportional to the likelihood the sensor is drifty.

3. BAYESIAN PROBABILITY CALCULATION

Assume there is observed data y, which is based on hypothesis θ. In order to make probability statement about θ, firstly we need to assume a prior probability for θ as $p(\theta)$. Then the joint probability of θ and y is calculated by

$$P(\theta, y) = P(\theta)P(y|\theta) \qquad (4)$$

Then the Bayes' rule is used to calculate the posterior probability for $P(\theta|y)$

$$P(\theta|y) = \frac{P(\theta)P(y|\theta)}{P(y)} \qquad (5)$$

Now, if we have two different hypothesis θ_1 and θ_2, both have an effect on the observed data y, to calculate the posterior probability for θ_1 and θ_2, we need two important assumptions: (1) θ_1 and θ_2 have no causal relationship (2) Besides θ_1 and θ_2, there is no other hypothesis that will affect the data, note that this assumption holds true here if θ_1 is 'sensor is drifty' and θ_2 is 'sensor is noisy' (we don't consider component faults at this point). The calculation is thus the following:

$$P(\theta_1|y) = \frac{P(\theta1)P(y|\theta1)}{P(\theta1)\mathrm{P}(\mathrm{y}|\theta1) + P(\theta2)\mathrm{P}(\mathrm{y}|\theta2)} \qquad (6)$$

4. SENSOR DRIFT FDD SCHEME

Based on PCA method and Bayesian probability calculation, we have come up with following scheme to detect and diagnose sensor drift (shown in Fig.1).

The difference between this scheme and the one used in [7] is the way training data is used, and the probabilistic extension of the result using Bayesian calculation.

5. DESCRIPTION OF SYSTEMS

To test this FDD scheme, we have developed an Air Handling Unit (AHU) system on Dymola [8], which

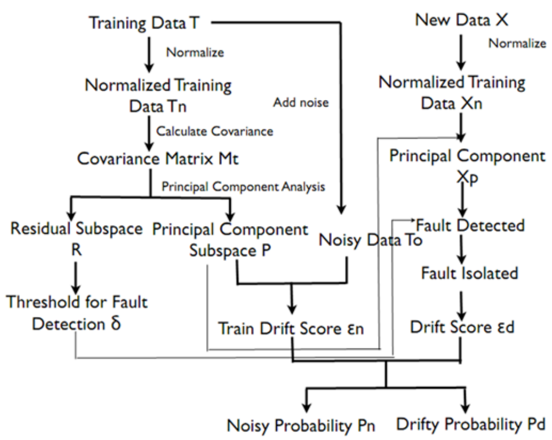

Fig1. Sensor Fault Detection and Diagnosis Schema

is a general purpose physics system modeling and solving environment. In this system, fresh air comes in and mixes with return air, both fresh air mass flow rate and mixed air mass flow rate are measured with sensors. Mixed air goes through a cooling coil, supply fan and to conditioning zone. Cooling coil chilling water mass flow rate is controlled by cooling coil valve, it is in a feedback loop with supply air temperature sensor. The supply air fan is controlled by the static pressure in supply air duct. Both supply air temperature control and supply air static pressure control are PI control.

Return air comes back from conditioning zone and goes through return air fan, part of it mixes with fresh air, the rest of it is exhausted. Return air fan is maintained at a constant rotation speed.

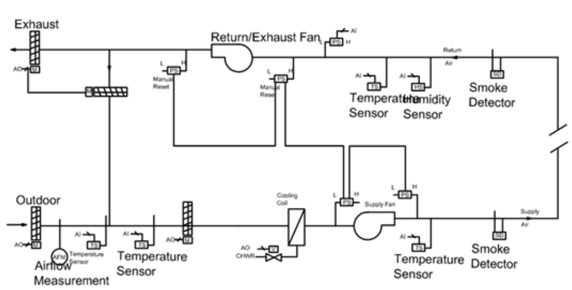

Fig2. Sensor Fault Detection and Diagnosis Scheme

6. DESCRIPTION OF EXPERIMENT

To test how the FDD scheme works for the system, we have chosen 11 sensors, two set points and two control signals, which form an enclosed AHU system, as shown in Table 1.

The simulation is conducted from 8am until 12pm. The conditioning zone is simulated using a 60 cubic meter filled with uniformly conditioned air. To simplify the experiment, the air coming from AHU is fed directly to the conditioning zone. To observe the dynamic behavior in conditioned air, the boundary conditions are changed during the experiment. Fresh air temperature increases from 31°C to 36 °C, the chilled water temperature increases from 5°C to 7 °C, the supply air temperature set point decreases from 28°C to 19°C, zone thermal load increases from 10kw to 20kw.

For the purpose of studying how sensor noise affects the PCA result, we need to assume the sensor accuracy. In the experiment, following accuracies are assumed for sensors (table 1).

Table 1. PCA Matrix Data Points

Index	Name	Accuracy
1	FMF (fresh air mass flow rate sensor)	1%
2	SMF (supply air mass flow rate sensor)	1%
3	RMF (return air mass flow rate sensor)	1%
4	FAT (fresh air temperature sensor)	1°
5	SAT (supply air temperature sensor)	1°
6	RAT (return air temperature sensor)	1°
7	FAE (fresh air enthalpy sensor)	1%
8	RAE (return air enthalpy sensor)	1%
9	SWT (cooling coil supply water temperature sensor)	1°
10	OAP (outside air pressure sensor)	0.1%
11	SAP (supply air pressure sensor)	0.1%
12	SAS (supply air temperature set point)	/
13	SPS (supply air static pressure set point)	/
14	CCV (cooling valve control signal)	/
15	SFR (supply fan rotation speed control signal)	/

The simulation period is from 8am to 12pm (4 hrs). To remove the transient data at the starting stage, data in the first hour (8am-9am) is discarded. Data from 9am to 9:24am is used as training data, which contains 50 samples.

7. DATA TRAINING

7.1 Loading Matrix

As mentioned above, we have in total 15 variables in the studying matrix. Since we have 50 samples in the training data, which is therefore a 15×50 matrix. Its generated eigenvector and eigenvalue matrix are 15×15 matrix. The last six eigenvalue matrix is presented below:

$$\begin{matrix} 1.17e-7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 6.45e-6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 8.25e-6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.62e-5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0013 & 0 \\ 0 & 0 & 0 & 0 & 0 & 14.99 \end{matrix}$$

Fig3. Last Six Eigenvalue Matrix

As introduced in [7], there are many ways to decide the number of principal components. Here we select the last vector as principal component, since its norm value is much larger than the other vectors. Then the rest vectors are regarded as residual components. The residual matrix is then multiplied by training data to get the error matrix, from which the threshold for fault detection can be calculated. Again, as introduced in [7], there are many ways to determine the threshold. Here the sum of norm value of each feature residual is used as the threshold, which is 0.32 in this case.

7.2 PC Update Scheme

The first test case uses normal operational as test data to see what the error score is like. Data after 9:24am are sampled every 12 minutes (in total 13 samples). During the period, the Principal Component (PC) is not updated. The result could be seen in Fig 4.

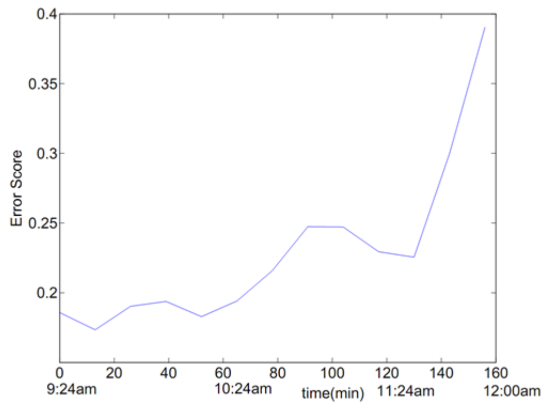

Fig4. Error score of normal operational data

The error score jumps at the last 12 minutes because during this period the boundary condition changes. The boundary condition has been changed at a constant speed from 9:24am to 11:53am. At 11:53am, the boundary conditions stop change and maintain at constant values. Therefore, the error score has been low before the last sample, and it jumps to high just at the last sample when constantly changing boundary condition is changed to a constant boundary condition.

Therefore, it is concluded that the rate at which the boundary condition changes has to be tracked. If the rate changes significantly, then the PC has to be updated in real time.

8. Experimental results

8.1 Fresh air mass flow rate sensor

In this test case, we test the fresh air mass flow rate sensor with possible noise and drift. Based on table 1, we added to FMF with noise that has zero mean and 0.16kg/s standard deviation. The data is then fed to PCA analysis. The error score in this case can be seen as the dotted line in Fig 5. As shown, due to the noise, the error score is between 8 and 14 during most of the time.

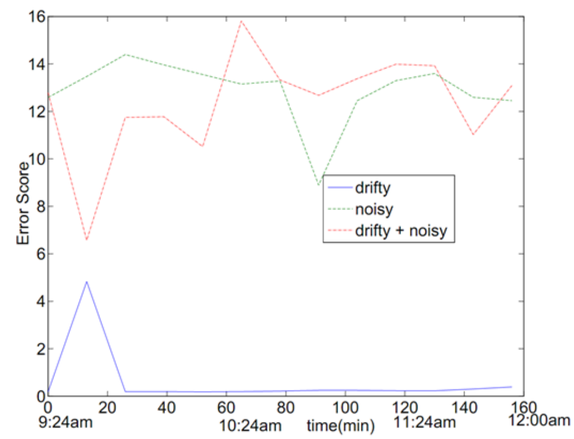

Fig5. Test Case – Fresh Air Mass Flow Rate Sensor

In the second case, we inserted a bias of 0.5kg/s at 9:40am, which is shown as solid line in Fig 5. As we can see, at around 9:40, the error score jumped to over 4 as a result of this drift, however, after 9:48am, due to the normalization of sample data, the drift error is no longer visible.

In the third case, we added both noise and drift to fresh air mass flow rate sensor (FMF). The result is surprising. At the time both drift and noise happens, the error score actually drops. Which means if the result is only interpreted through the absolute value of error score, the conclusion maybe the opposite to the real case. This finding leads to our next step, using Bayesian method to calculate the probability for both sensor noise and drift, which is in the section 9.

A last note to the third case, this situation happens because the noise and drift are on the opposite directions (one is positive, the other is negative). In case they occur on the same direction, the error score is expected to increase.

8.2 All sensors noise

After we studied fresh air mass flow rate sensor, we tested the noise for all the other sensors. All the noise was with zero mean and standard deviation chosen from table 1. For each test case, we calculated the mean and average of the error score, which is listed in table 2.

Table 2. Error Score Caused by Sensor Noise

Index	Sensor Name	Score Mean	Score Std
1	FMF	12.9	1.34
2	SMF	13.08	1.27
3	RMF	13.12	1.61
4	FAT	12.43	1.23
5	RAT	12.03	1.48
6	FAE	11.97	1.19
7	RAE	11.81	1.21
8	SWT	12.48	1.63
9	OAP	13.06	1.4
10	SAP	12.9	1.27
11	TOT	65.44	1.36

It is observed that although each different sensor has varying accuracy range, the error score caused by each noisy sensor is close, with a mean between 11 and 13, and standard deviation smaller than 2.

The case name ‘TOT’ stands for the case in which all sensors are added with noise. In this case, the total error score is larger than each specific sensor, but smaller than the sum of score for all the other cases, which suggests the total error score is a useful indicator of the extent of faults. But a more effective approach to detect multiple sensor faults is through error score of each individual feature.

8.3 Error Score Caused by Drift

Error score caused by drift is studied separately from noise. Two factors that affect the analysis are studied: (1) the time when the error occur (2) the direction of drift (positive or negative). The results are shown in Fig 6.

It was found that the error score in this case depends more on the drift direction than the occurring time. This finding could change depending on the extent of the drift as well. But it is concluded that both the occurring time and drift direction could affect the error score. Another finding is that the error score caused by sensor drift in this experiment arranges from 4 to 17, which is comparable to the error score caused by sensor noise (average is between 11 and 13). In another word, by purely looking at the error score, it is impossible to differentiate sensor drift from sensor noise.

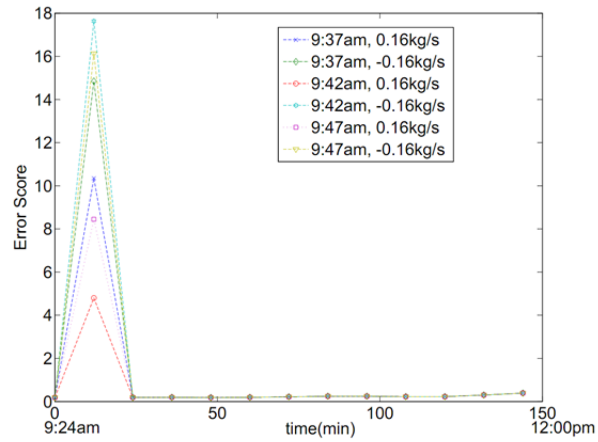

Fig6. Error score caused by drifty fresh air mass flow rate sensor

9. Probability Extension

To extend the deterministic analysis to a probability based analysis, firstly we define notations shown in the following table – table 3.

Table 3. Notations

Index	Symbol	Stands for
1	P_{npr}	prior noisy probability
2	P_{dpr}	prior drifty probability
3	P_{en}	observation evidence that the sensor is noisy
4	P_{ed}	observation evidence that the sensor is drifty
5	P_{npo}	posterior noisy probability
6	P_{dpo}	posterior drifty probability
7	ε_{en}	expected noisy score
8	ε_{ed}	expected drifty score
9	ε_{cf}	current feature error score
10	σ_{ns}	history noisy score variance
11	σ_{ds}	estimated drifty score variance
12	δ	threshold to employ drifty probability calculation

We use the following algorithm to calculate P_{npo} and P_{dpo}.

Calculate the sensor is noisy probability

$$P_{en} = normpdf(\epsilon_{cf}, \varepsilon_{en}, \sigma_{ns})$$

Calculate the sensor is drifty probability

If $\epsilon_{cf} > \delta$

$$P_{ed} = normpdf(\epsilon_{cf} - \epsilon_{en}, \varepsilon_{ed}, \sigma_{ds})$$

Else

$$P_{ed} = 0.001$$

Update sensor drift posterior probability and sensor noisy posterior probability

$$P_{dnp} = P_{dpr}\frac{P_{ed}}{P_{en} + P_{ed}} + P_{npr}\frac{P_{ed}}{P_{en} + P_{ed}}$$

$$P_{dpo} = \frac{P_{dpr}}{P_{dnp}} \frac{P_{ed}}{P_{en} + P_{ed}}$$
$$P_{npo} = \frac{P_{npr}}{P_{dnp}} \frac{P_{ed}}{P_{en} + P_{ed}}$$

10. Test case

In the Bayesian calculation test case, the fresh air mass flow rate sensor (FMF) is used again. In this test case, it is assumed that the sensor noise standard deviation is 0.01 kg/s, and at 9:37am, a drift of -0.16kg/s is seeded into the system. Due to both faults, the error score varies from 10 to 15 during the simulation, as shown in Fig 7.

The Bayesian calculation result is shown as dotted line in Fig 7.

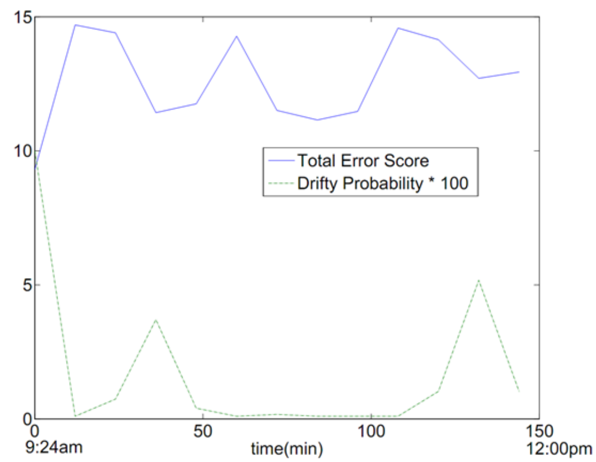

Fig7. Bayesian Calculation – Test case

It can be seen that the drift fault seeded at 9:37 is not really visible in both curves in this case. This is attributed to the closeness of error score caused by drift and sensor noise. But assume the drift is larger, the fault should manifest itself in a probability value higher than the peaks shown here.

In the next case, the extent of noise is decreased, as shown in Fig 8. The error score caused by noise is less than 3. At 11:12 am, a drift of 0.16kg/s is seeded, which causes the error score increased to 10. In this case, the Bayesian algorithm captures the difference immediately and made a high probability prediction (90%).

To conclude, with the prerequisite of training the normal operational data with artificial drift fault and random noise, using the Bayesian algorithm that is shown is this paper, user can now get a quantitative confidence of the diagnostic result. However, how well this algorithm works does depend on the difference between the errors scores caused by drift and noise. In case the difference is small, this algorithm will not report a drift fault when the cause is really a drift fault.

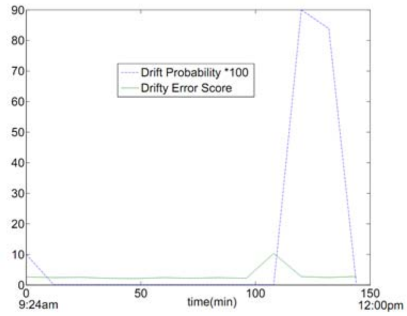

Fig8. Bayesian algorithm performance

11. Conclusion

In this paper, we have tested a PCA method against various situations to check its robustness. We find that (1) Principal Component (PC) does not need to be updated if the system boundary condition changes at a constant speed, however, if the speed changes significantly, then PC has to be updated in real time. (2) Both sensor drift and sensor noise can be detected by PCA, sensor noise manifests itself throughout the whole simulation period, sensor drift is only visible when the fault happens. (3) The error score caused by sensor noise is comparable to the value caused by sensor drift, therefore, by looking at the absolute value of error score, it is impossible to differentiate sensor noise from sensor drift.

To help separate the noisy sensor case from a drifty sensor, we developed a Bayesian algorithm to calculate the probability of both noisy and drifty. It is found that Bayesian algorithm can filter certain alarms caused by sensor noise, although it does not help when the error score caused by noise and drift are close. But most of all, a Bayesian algorithm gives a quantitative confidence level of the diagnostic result to the user.

REFERENCES

1. A. Dexter, J.Pakanen, Fault detection and diagnosis methods in real buildings, Annex 34: Computer-aided evaluation of HVAC System performance, Technical Report, International Energy Agency, 2001.
2. C.H. Yang, Y.Jiang, Sensor fault detection of HVAC system – system constraints and voting, Pan Pacific Symposium on Building and Urban Environmental Conditioning in Asia (1995).
3. H. Yang, S. Cho, C. Tae, M. Zaheeruddin, Sequential rule based algorithms for temperature sensor fault detection in air handling units, Energy Conversion and Management 49 (2008) 16.
4. Z. Du, X. Jin, Detection and diagnosis for sensor fault in HVAC systems, Energy Conversion and Management 48 (2007) 10.
5. Z. Hou, Z. Lian, Y. Yao, X. Yuan, Data mining based sensor fault diagnosis and validation for building air conditioning system, Energy Conversion and Management 47 (2006) 12.
6. S. Wang, J. B. Wang, Robust sensor fault diagnosis and validation in HVAC systems, Transaction of the Institute of Measurement and Control 24 (2002) 33.
7. S. Wang, F. Xiao, AHU sensor fault diagnosis using principal component analysis method, Energy and Buildings 36 (2004) 14.
8. D. Bruck, H. Elmqvist, et al. Dymola for Multi-Engineering Modeling and Simulation. Modelica Conference, Munich. 2002.
9. J. Schein, S. T. Bushby, N. S. Castro, J. M. House, A rule-based fault detection method for air handling units, Energy and Buildings 38 (2006) 8.
10. W. Y. Lee, J. M. House, D. R. Shin, Fault diagnosis and temperature sensor recovery for an air-handling unit, ASHRAE Transactions 103 (1997) 13.
11. S. Wang, J. Qin, Sensor fault detection and validaton of VAV terminals in air conditioning systems, Energy Conversion and Management 46 (2005) 19.
12. F. Xiao, S. Wang, J. Zhang, A diagnostic tool for online sensor health monitoring in air-conditioning systems, Automation in Construction 15 (2006) 15.
13. Z. Du, X. Jin, Detection and diagnosis for multiple faults in VAV systems, Energy and Buildings 39 (2007) 12.
14. K. Pearson, On lines and planes of closest fit to systems of points in space, Philosophical Magazine 2 (1901) 14.
15. A. Sturn, J. Quackenbush, Z. Trajanoski, Genesis: cluster analysis of microarray data, Bioinformatics 18(2002) 2.
16. J. Yang, D. Zhang, A. F. Frangi, J. Yang, Two-dimensional PCA: A new approach to appearance-based face representation and recognition, IEEE Transactions On Pattern Analysis and Machine Intelligence 26 (2004) 7