SimBuild
2024

Eleventh National Conference of IBPSA-USA
Denver, Colorado
May 21 – 23, 2024

Adaptive Fault Detection and Diagnosis Based on Growing Gaussian Mixture Regressions for Passive chilled Beams system

Sujit Dahal[1], Liping Wang[1], and James Braun[2]
[1]University of Wyoming, Laramie, WY
[2]Purdue University, West Lafayette, IN

Abstract

The conventional fault detection and diagnosis (FDD) methods are limited in their ability to identify faults not present in their training data. To address this limitation, a learning-based FDD strategy using a Gaussian mixture regression (GMR) is proposed. This approach can detect and incorporate information about unknown faults into the model by adapting its structure. The performance of this FDD scheme was assessed using experimental data from a passive chilled beams (PCB) system, an area where FDD research is lacking. The model was trained and tested with data from the winter season and further evaluated with data from the spring and summer seasons. Common faults in PCB systems were identified through literature review and from experience with the system operation and were implemented via the building automation software. A feature selection method is employed to isolate the features that show notable symptoms for normal operation and each implemented fault which is important in distinguishing normal operations and faulty operations. The selected features are then used to develop a GMR FDD model with experimental data for normal operation and known faults during the winter season. During model testing, if the new fault data is identified as an unknown state, the model evolves through updating and the addition of Gaussians to incorporate the information of the unknown state. The performance of the initial model and the evolved model is tested using the normal operation, known fault, and unknown fault data for the winter season. Furthermore, the research examined the performance of the model across seasons, conducting tests with data from the spring and summer periods. The model was found to be less effective when used in its static form during spring and summer testing, as it failed to adapt to changes in certain system features across the seasons. However, when the same data was tested using the evolved model, it proved highly effective in incorporating new fault patterns and achieving accurate fault diagnosis. Notably, the model achieved a minimum fault prediction accuracy of 92.5% for the spring season and 92.4% for the summer season.

Introduction

Hydronic radiant systems, which rely on circulating water through pipes integrated within the building structure to regulate temperature, have seen a growing preference as an energy-efficient and thermally comfortable alternative to conventional heating, ventilation, and air conditioning (HVAC) systems for heating and cooling (Fabrizio et al., 2012; Joe & Karava, 2019). In recent years, chilled ceiling systems have gained growing popularity due to their favorable attributes, which include excellent thermal comfort, superior air quality, reduced energy usage, and highly efficient ventilation (Lai, 1995).

The chilled ceiling systems can be broadly classified into radiant cooling panels, active beams, and passive beams. Passive chilled beams (PCBs) cool the room air by utilizing a natural convection process driven by the temperature and density variation between the room air and the air in proximity to the beams. In contrast, active chilled beams not only induce natural convection within the room but also introduce fresh air to blend with the circulated air. As a result, active chilled beams have the potential to deliver greater cooling or heating output compared to passive beams (Riffat et al., 2004). Chilled beams typically function within water temperatures ranging from 14-18 °C. PCBs are also radiant systems that offer an opportunity to decrease cooling demands by harnessing the improved occupant comfort provided by radiant cooling. This heightened comfort enables the maintenance of slightly higher zone temperatures while upholding the same comfort criteria. Additionally, the need for elevated radiant surface temperatures (to prevent condensation) results in higher operating temperatures compared to traditional cooling methods, thereby enhancing the energy efficiency of chiller units (Kim et al., 2015).

Numerous investigations into passive ceiling cooling systems have been carried out and models have been developed to forecast energy consumption and assess the impact on thermal comfort. Key elements of a PCB system are heating/cooling sources, valves, pumps, sensors, and thermostats (Chakraborty & Elzarka, 2019). Failures in these components, inaccuracies in sensors, or general wear and tear can have adverse effects on system performance, energy efficiency, and equipment longevity (Chakraborty & Elzarka, 2019). Hence, the FDD of the PCB system becomes essential to ensure the optimal and cost-efficient operation of the system. Previous research on hydronic heating and cooling has predominantly centered on heat transfer analysis, quantifying system capacity, computational fluid dynamics, and energy simulation (Feng et al., 2015; Hao et al., 2007; Khan et al., 2015; Kim et al., 2019; Laouadi, 2004; Tian & Love, 2009). The research by Wang et al. 2023 exhibited the application of an evolving learning-based FDD technique using Gaussian mixture regression to identify and diagnose typical faults in PCB systems (Wang et al., 2023). Nevertheless, there is a noticeable research gap in methods for FDD of the PCB system.

Methods for fault detection can be categorized into data-driven, model-based, and rule-based approaches (Katipamula & Brambley, 2005). Rule-based methods employ if-then-else rules derived from expert knowledge but lack a foundation in underlying physics principles (Katipamula & Brambley, 2005; Yu et al., 2014). The primary drawbacks of the rule-based methods include their system-specific nature, the potential for significant failure when applied outside the bounds of their incorporated knowledge, and the challenges associated with updating or modifying them (Katipamula & Brambley, 2005). Physical model-based approaches rely on understanding system behavior based on physical laws, involving prediction of system behavior compared to measured output (Liang & Du, 2007; Wang et al., 2012; Wang & Haves, 2014). These methods demand detailed input data, making them complex and computationally intensive (Katipamula & Brambley, 2005; Wang & Haves, 2014). Data-driven models offer advantages, predicting based on historical data without extensive building information (Gao et al., 2016; Yan et al., 2020). Zhu et al. (Zhu et al., 2019) combined decoupling features with a random forest classifier for fault identification. Van Every et al. (Van Every et al., 2017) proposed a fault detection algorithm using Gaussian process regression and support vector machines. Yan et al. (Yan et al., 2020) employed generative adversarial networks for chiller FDD. Despite their popularity, data-driven models have limitations, staying static once trained and struggling with unknown faults (Karami & Wang, 2018).

To address these limitations, the development of evolve-learning-based FDD methods becomes essential, allowing the detection and incorporation of information about unknown faults (Wang et al., 2023). Adaptive learning enables real-time adjustments and improved handling of system variations (Karami & Wang, 2018). Costa et al. (Costa et al., 2014) proposed a two-stage unsupervised approach for fault detection and diagnosis. Gaussian mixture models (GMMs) have been used in various fields, such as predicting air mass in combustion engines and robot motion learning. Nakamura et al. introduced a batch-incremental adaptive methodology for FDD using parsimonious GMMs (Nakamura et al., 2017). Adaptive GMR integrated with an Unscented Kalman filter successfully detected and diagnosed faults in a water-cooled multi-chiller plant system (Karami & Wang, 2018).

This research paper employs an evolving learning-based GMR model for the detection and diagnosis of potential faults within PCB systems. This approach demonstrates adaptability as it responds to variations in building system performance and component behavior. The methodology leverages an adaptive Gaussian mixture regression algorithm to construct a data-driven model capable of accurately diagnosing known faults established during training, identifying unknown states, and evolving its models to incorporate information about these unfamiliar states and pinpoint new faults. Moreover, if the building operational conditions and parameters vary seasonally, the model can dynamically adjust to these changes through continuous evolution. There is no need to create separate models for each season, as a single model proves effective in detecting and diagnosing faults across multiple seasons. The baseline GMR is trained using normal operation data from the winter season to predict system flow, heating, and cooling capacity. Selected features essential for FDD model development are utilized to construct the GMR FDD model. The model performance of the model is evaluated using both known and unknown fault data, and if an unknown fault is detected, the model undergoes further evolution using the newfound fault data. Subsequently, the prediction accuracy of the model is retested post-evolution. Furthermore, the model is subjected to testing without and with evolution using faulty data from the spring and summer seasons, and the resulting prediction accuracy is thoroughly analyzed.

Proposed FDD methodology

The FDD process and evolving mechanism are shown in Figure 1. The experimental dataset utilized in this study encompasses both faulty and normal operation data, meticulously organized into distinct training and testing sets. Within this framework, the training set serves as the foundation for developing the GMR FDD model.

Meanwhile, the testing dataset presents a multifaceted evaluation ground that includes known fault scenarios, normal operation states, and a category known as "unknown faults."

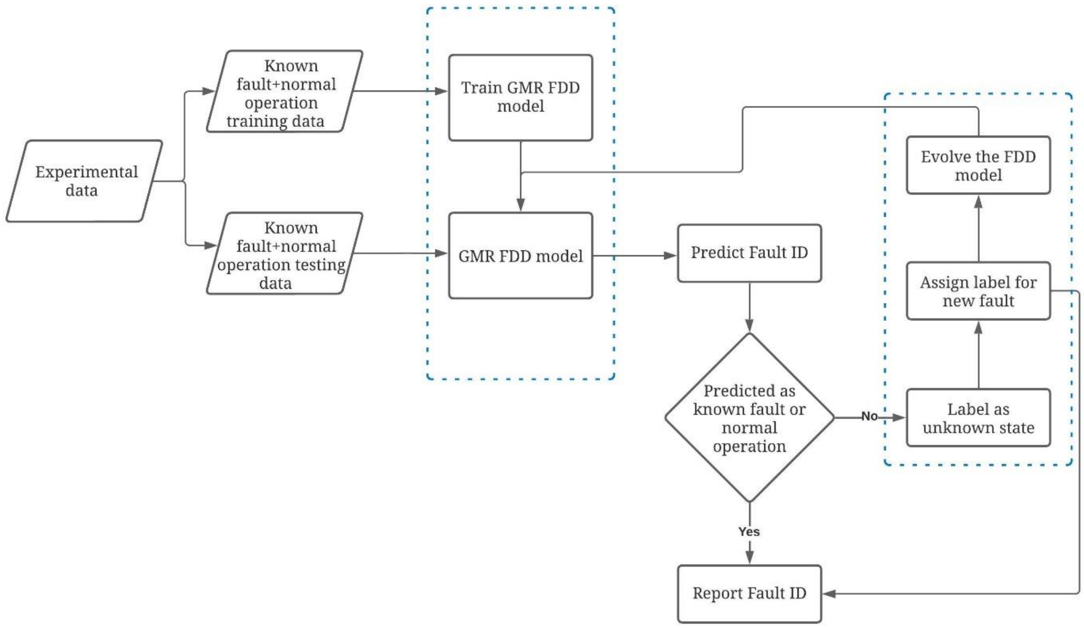

Figure 1. Schematic diagram of FDD model development and evolving

During the GMR FDD model testing phase, when a prediction corresponds to a label denoting a known fault or normal operation, this outcome is reported. However, a challenge arises when the prediction does not align with any of the predefined labels for known faults or normal operation. In such instances, the data is examined by an expert. The objective of this examination is twofold: firstly, to ascertain whether the unclassified data represents a previously unidentified operational state within the realm of normal operation or any known fault condition. Secondly, it seeks to determine if the data signifies an entirely new fault scenario not encountered before. In the event that this unknown data is categorized as a new faulty condition, an appropriate label is assigned to accurately represent this newfound fault.

Crucially, this process sets in motion the evolution of the GMR FDD model. This evolutionary mechanism ensures that the model continues to learn, adapt, and refine itself with each testing data point it encounters. As a result, the performance of the model continually improves over time, making it increasingly adept at detecting and diagnosing faults, even those that were previously unknown.

Gaussian Mixture Regression and Growing Gaussian Mixture Regression

The Gaussian mixture model (GMM) is a potent tool for dissecting complex datasets, revealing underlying patterns, and illuminating the distinct clusters that exist within the data population. Each of these clusters is thought of as a Gaussian distribution, providing a comprehensive understanding of how data points are grouped and distributed across different regions of the dataset. This technique proves invaluable in various domains, from machine learning to statistical analysis, where the goal is to uncover the inherent structure and relationships within the data. The GMM effectively partitions the data into K discrete components thereby treating it as a diverse population. Each of these components is characterized by a Gaussian distribution, featuring a mean value that encapsulates the central tendency of the cluster, along with an associated variance parameter. To grasp the distribution of the data, the probability density function of the GMM is constructed as a summation of these Gaussian probability density functions, each of which is weighted according to its significance within the model. This formulation can be precisely expressed using the mathematical representation provided in Equation (1).

$$f(\mathbf{x}, y|\pi, \boldsymbol{\mu}, \delta) = \sum_{j=1}^{k} \pi_j N(\mathbf{x}, y|\boldsymbol{\mu}_j, \delta_j) \quad (1)$$

Where $N(\mathbf{x}, y|\boldsymbol{\mu}_j, \delta_j)$the multivariate Gaussian density function, k is the number of Gaussian mixtures, $\boldsymbol{\mu}_j$ is the mean vector, δ_j is the covariance matrix of j-th Gaussian component and π_j is the weight of the j-th component (mixing coefficient).

To determine the unknown parameters of the mixture model (π_j, $\boldsymbol{\mu}_j$, δ_j), the GMM is trained using a dataset. During this training, the Expectation Maximization (EM) algorithm is used for deducing the maximum likelihood estimates of these critical GMM parameters. The EM algorithm unfolds through two pivotal steps: the Expectation (E) step and the Maximization (M) step. In the E-step, the algorithm calculates the responsibility function by assessing the current parameter values. This function essentially quantifies the degree of contribution of each component (or cluster) to the observed data points within the dataset. In the M-step, the algorithm undertakes the critical task of updating the parameter values based on the information gathered from the responsibility function. This iterative process of expectation and maximization continues until the model converges to optimal parameter values. When it comes to predicting the dependent variable Y when provided with a new input (x), the Gaussian Mixture Regression (GMR) method steps into the spotlight. Here, the objective is to estimate the conditional expectation over [Y│X=x], a task facilitated by GMR. This estimation process is outlined in Equation (2). The predicted value of the dependent variable Y is represented by $\hat{y}(\mathbf{x})$.

$$\hat{y}(\mathbf{x}) = E[Y|\mathbf{X} = \mathbf{x}] = \sum_{j=1}^{k} w_j(\mathbf{x}) m_j(\mathbf{x}) \quad (2)$$

The mechanisms, mathematical equations, and methodologies underpinning both the GMM and GMR are explained in detail in (Sung, 2004).

The evolution of the GMR model unfolds through several stages during training with baseline data and continuous updating with new data. The following steps outline this process, and a comprehensive understanding of the evolution process can be found in the research conducted by (Bouchachia & Vanaret, 2011).

Step 1: Compute the probability density for the new data sample across all Gaussian components, employ distance criteria, and identify the best-fitting component.

Step 2: When a matching Gaussian component is identified, update the parameters of that specific Gaussian.

Step 3: If the new sample data doesn't match any existing Gaussian mixture component, and the current component count is below the maximum threshold, create a new Gaussian component. Initialize the prior, mean, and covariance matrix for the newly generated component.

Step 4: Conclude by normalizing the priors of the new Gaussian mixture model.

Experimental test facility

Living Lab 1, situated within the Herrick Building at Purdue University, encompasses an office space measuring 9.9 meters in width, 10.5 meters in length, and 3.2 meters in height. This space is equipped with a passive chilled beam (PCB) system, comprising a total of 30 PCBs strategically positioned to provide the necessary sensible cooling for the area. The PCBs are affixed to the ceiling and are organized into three distinct sections, each equipped with individual valves for precise control over the flow of chilled water as shown in Figure 2. Specifically, these sections comprise 12 PCBs in the southern segment, another 12 in the middle section, and 6 in the northern part of the room. Each of these PCB units measures 12 centimeters in length, 57 centimeters in width, and 14 centimeters in depth. The PCB system is further enhanced with essential components, including a pump, three control valves, a heat exchanger, temperature sensors, a differential pressure sensor, and a flow rate sensor, ensuring efficient and responsive operation. In tandem with the PCB system, an Air Handling Unit (AHU) plays a crucial role in managing latent and ventilation loads. It achieves this through variable air volume distribution systems, contributing to the overall environmental control within this living lab space.

The supply water setpoint temperature, chilled and hot water valve positions, PCB valve positions, and pump output are all controlled with individual PID loops. The maximum of the space dew point temperature plus 3 °F and the output from the PI loop is used as the supply water temperature setpoint temperature. Space temperature and space humidity are used to calculate the space dew point temperature and 3 °F is added to the space dew point temperature to keep the chilled water setpoint 3 °F above the space dew point temperature. The temperature reset PID loop is used to calculate the supply water setpoint temperature. This PI loop is designed to keep the output between 62 to 73 °F.

Cooling or heating mode is determined by comparing the space temperature (controlled variable) with the space setpoint temperature which is fixed at 73°F. If the space temperature is above 73 °F the system will be in cooling mode and if it is lower than 73 °F the system will be in heating mode. The supply water temperature is controlled through the heat exchanger connected to the district chilled water supply and hot water supply. The chilled water flow and hot water flow are controlled using cool water and hot water valves on the supply side of the heat exchanger. Both the chilled water valve position and hot water valve position are determined using individual PID loops with supply water temperature as the controlled variable and the supply water temperature setpoint calculated in the previous step.

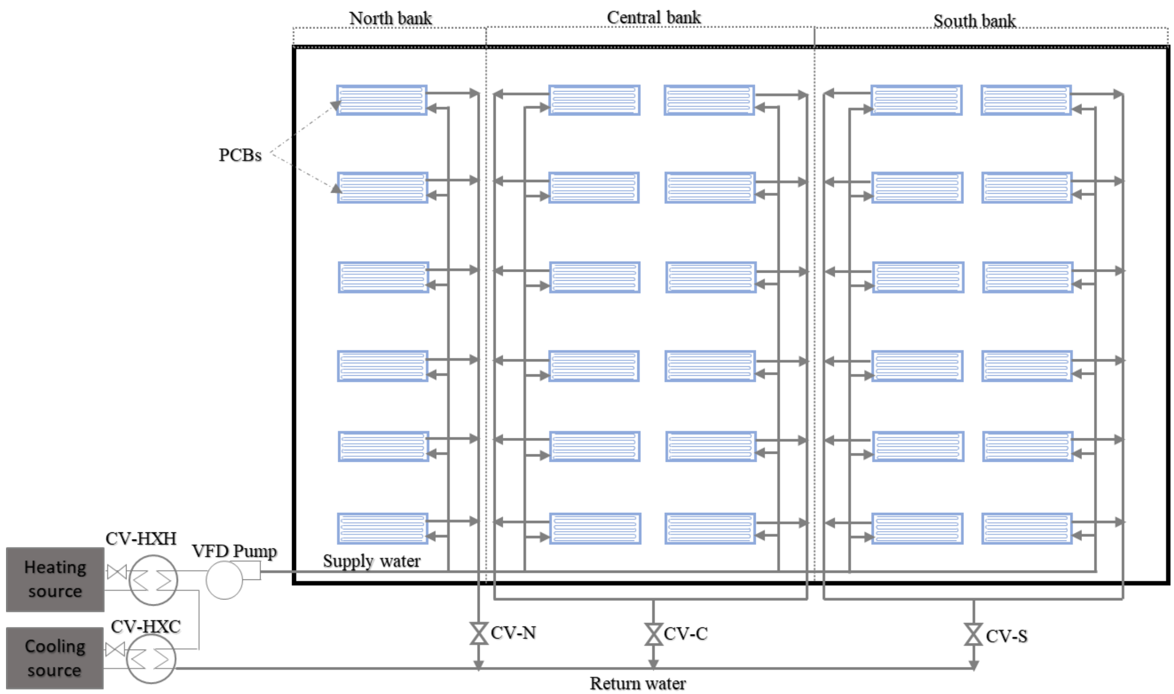

Figure 2. System diagram of PCBs installed in the Living Lab 1 ceiling

The three PCB positions are also calculated using a PI loop with space temperature as the controlled variable and the space setpoint temperature of 73°F. The maximum valve position of all three valves is set at 100% and the minimum for the north PCB valve and south PCB valve are set at 30% and the central PCB valve is set at 0%. The supply water pump is enabled if either the chilled water loop or hot water loop is enabled. The supply water pump position is also controlled based on the output of a PI loop with differential pressure in the supply pipe as the controlled variable and the setpoint pressure of 10 psi. The maximum pump output is set at 100% and the minimum is set at 0%.

Experimental method and data collection

The experiments were carried out within an open-plan office environment equipped with a PCB system. To efficiently manage and monitor the operating conditions of the experimental space, a Building Automation System (BAS) based on the Niagara Framework was employed. This BAS played a vital role in regulating the experimental environment and offered the capability to override control parameters when necessary. Data collection encompassed normal operation data obtained during both winter and summer seasons, as well as faulty data gathered throughout winter, spring, and summer. A total of five distinct fault types, varying in severity, were introduced deliberately. These fault types were selected based on their realistic occurrence likelihood and feasibility of implementation via the BAS. Table 1 provides a comprehensive breakdown of each fault type, including the fault label, a detailed description, the levels of severity, and the dates when these faults were introduced during winter, spring, and summer.

The fault types include stuck hot and cold water flow valves, temperature sensor offsets, and the VFD pump stuck faults. To implement the space temperature sensor offset faults, an offset of either +5 or -5 was introduced to modify the temperature readings recorded by the sensor. Similarly, supply water temperature sensor offset faults were implemented with offsets of +9 and -9. The position of the three PCB valves is based on the output of the PI loop, where the maximum valve position of all three valves is set at 100% and the minimum valve position is set at 30%. The implementation of PCB valve stuck faults involved the insertion of a new block between the block containing the PI loop output and the signal block responsible for controlling the valve positions. This additional block prevented the signal block from receiving values from the PI loop output and could be manually overridden to specific positions (0%, 50%, and 100% in this case). The introduction of cold water valve stuck faults and VFD pump stuck faults followed a similar methodology as that employed for the PCB valve stuck faults, ensuring consistency in fault implementation.

Table 1. Implemented faults and implementation periods

Fault label	Description
SpaceT+ve	Space temperature sensor offset of +5 F
SpaceT-ve	Space temperature sensor offset of -5 F
PCB_V_0	PCB valves stuck at 0% open
PCB_V_50	PCB valves stuck at 50% open
PCB_V_100	PCB valves stuck at 100% open
SWT+9	Supply water temperature sensor offset of +9 F
SWT-9	Supply water temperature sensor offset of -9 F
Cold_V_0	Cold water valve stuck at 0% open
Pump_50	VFD pump stuck at 50% of its maximum speed

Experimental System modeling

The GMR model was used to make predictions concerning the flow and cooling capacity of the PCB system. To identify the input data for the cooling capacity prediction model, a correlation coefficient was employed. This statistical measure gauges the strength of the relationship between pairs of variables. In essence, variables exhibiting a high correlation coefficient were singled out as input parameters for the cooling capacity prediction. The training dataset included normal operation data spanning from March to June 2021. In the subsequent phases, July 2021 was initially utilized to test the performance of the model, after which it played a pivotal role in evolving the model. The evolved model, thus refined through iterative processes, eventually served as the baseline for predicting the PCB system cooling capacity under normal operating conditions. Furthermore, when dealing with collected faulty data, this baseline model came into play, facilitating the estimation of the PCB cooling capacity during normal operational scenarios. In addition to cooling capacity predictions, the GMR model was also instrumental in predicting the water flow rate within the PCB system. This predicted water flow rate served as a crucial input when estimating the PCB cooling capacity. The input parameters thus selected for the prediction of water flow rate and cooling capacity are listed in Table 2.

Table 2. Input parameters for PCB cooling capacity and PCB flow prediction models

PCB cooling capacity prediction input	PCB flow prediction input
Outdoor air temperature	Outdoor air temperature
Outdoor air relative humidity	Space temperature
Space temperature	Operation status of the pump
Façade temperature	Average of three PCB valve position
Average of three PCB valve position	
Hot water valve position	
Cool water valve position	
GMR predicted water flow rate	

Once the requisite input parameters were identified, the dataset underwent a normalization process. This crucial step was undertaken to standardize the parameters, ensuring that they all conformed to a common scale. Consequently, for each parameter, the mean value was set to zero, and the standard deviation was adjusted to one. This normalization procedure is vital for creating a level playing field, as it enables the model to effectively process and compare data from parameters with varying scales, promoting accurate and meaningful predictions. Another factor in developing the GMR model is determining the number of components. This parameter plays a significant role in shaping the quality and performance of the model. Specifically, for the prediction of PCB water flow rate, a careful choice was made to employ 30 components. In the case of cooling capacity prediction, a total of 50 components were selected. These numbers were arrived at considering the right balance between model complexity and prediction accuracy. Furthermore, a learning rate of 0.01 and a closeness threshold of 10 were chosen. These values were determined through experimentation and iterative refinement to ensure that the model exhibits robust learning capabilities while maintaining the ability to detect significant deviations. Such parameter tuning is essential in constructing a model that reliably predicts PCB system behavior with precision and accuracy.

The comparison between the measured and predicted PCB flows and cooling capacity during model testing is shown in Figure 3 and Figure 4 respectively. The coefficient of determination R2 between predicted and measured PCB flow during training is 0.98 and between predicted and measured PCB cooling capacity is 0.94 which shows excellent prediction accuracies. As shown in Table 3 the CV-RMSE for the predicted PCB flow is 5.7% and the predicted cooling capacity is 17.8 during the model testing which is within the ASHRAE recommendation of 30% or less for the baseline model.

Table 3. PCB flow and cooling capacity prediction errors during model testing

	PCB cooling capacity prediction	**PCB flow prediction**
CV-RMSE (%)	17.8	5.7
NMBE (%)	-1.7	2.4

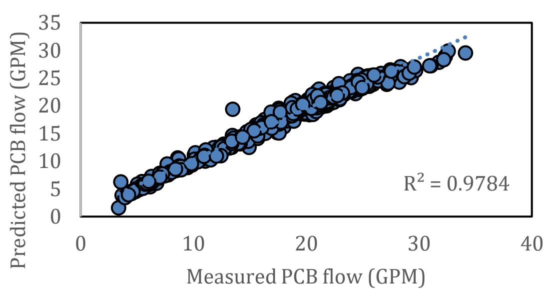

Figure 3. Measured vs Predicted PCB flow during model testing

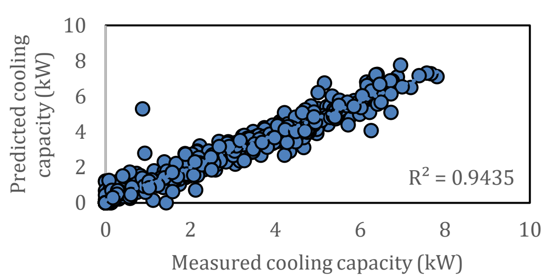

Figure 4. Measured vs Predicted PCB cooling capacity during model testing

Feature selection process

We collected trend data at 5-minute intervals for a range of parameters extracted from the PCB system. The trended parameters are the 3 PCB valve signals (averaged for analysis), space temperature, façade temperature, PCB system water flow rate, PCB supply and return water temperatures, cold and hot water valve signals on the heat exchanger side, and differential pressure of the PCB loop. The PCB system water flow rate and PCB supply and return water temperature were used to calculate the PCB cooling capacity at each timestamp. In addition to these parameters, the PCB system water flow rate and the PCB cooling capacity predicted using the GMR model gave two additional parameters. These parameters are used to develop a set of 7 features which includes deviations between the parameter and the setpoint (space temperature and differential pressure), deviation of the predicted value from the measured value (supply water flow rate and PCB cooling capacity), the difference between the supply and return water temperature, difference between the supply water temperature and the space setpoint temperature, and the rate of change of supply water flow observed over a 12 hour moving window (maximum flow for the 12 hour period– minimum flow for the 12 hour period divided by number of data points in the 12 hour period). Furthermore, we went a step further to derive seven additional features from the ones previously described. These additional features were computed as the cumulative sums of each feature over a 12-hour moving window, providing a comprehensive perspective on the temporal behavior and trends of these parameters across the seasons. This brings the total number of features to 14 which is listed in Table 4 where the features column has the code name given to each feature which is described in the description column.

Table 4. List of the initial set of features

Features	**Description**
ΔT	Space temperature difference from set point
ΔP	Differential pressure difference from set point
ΔF	Measured-predicted flow
ΔC	Measured – predicted Cooling capacity
$\Delta \mathbf{T}_{\mathbf{swt-Ssp}}$	Supply water temperature difference from space setpoint temperature

$\Delta \mathbf{T_{sw-rw}}$	Supply water temperature-return water temperature
Δf	Flow change rate for 12-hour period
CUSUM_ΔT	12-hour cumulative sum of Space temperature difference from set point
CUSUM_ ΔP	12-hour cumulative sum of Differential pressure difference from set point
CUSUM_ ΔF	12-hour cumulative sum of Measured-predicted flow
CUSUM_ ΔC	12-hour cumulative sum of Measured – predicted cooling capacity
CUSUM_ $\Delta \mathbf{T_{swt-sp}}$	12-hour cumulative sum of Supply water temperature difference from set point
CUSUM_ $\Delta \mathbf{T_{sw-rw}}$	12-hour cumulative sum of supply water temperature-return water temperature
CUSUM_ Δf	12-hour cumulative sum of Flow change rate for 12-hour period

The selection of essential features for constructing a GMR Fault Detection and Diagnosis (FDD) model followed a procedure outlined in (Fijany & Vatan, 2006; Wang et al., 2023). This process involves two steps: initially, training data is employed to identify symptoms associated with each fault, and subsequently, an optimization algorithm is applied to establish the minimal set of features required for diagnosing each fault. Faults may induce alterations in correlated features, which serve as symptoms for fault identification. This is achieved by comparing feature values under faulty and fault-free conditions. The feature selection process utilizes data collected during both normal operation and faults occurring during the winter season.

A feature matrix was established for each feature by assigning values to individual data points. Symptoms resembling normal operation were designated with a value of 2, symptoms indicating a fault where the measured values exceeded the upper baseline limit were assigned a value of 1, and symptoms indicating a fault where the measured values fell below the lower baseline limit were assigned a value of -1. This matrix effectively captured the distinctive patterns associated with various operational states. The upper baseline limit is determined as the maximum feature value during the baseline period (M) plus a degree of uncertainty (a × sd), while the lower baseline limit is calculated as the minimum feature value during the baseline period (m) minus a degree of uncertainty (a × sd). Here, 'a' represents the coefficient utilized to fine-tune the level of uncertainty, and 'sd' stands for the standard deviation of the feature during the baseline period. Five different values of the coefficient 'a' were employed, including 0, 1, 2, and 3.

In the next step, key signatures were identified for each fault. The symptom was preserved if a feature consistently exhibited the same symptom across all rows for a given fault. However, if the symptom varied across rows, it was represented as 0, indicating that the feature did not present a distinct symptom for diagnosing the fault. This process was repeated for all faults, resulting in the development of Table 5, which contains a key signature for each fault.

Table 5.Signature matrix with key signature for each fault (the rows are the fault names, and the columns are the 14 features provided in Table 4)

Faults	**Signature**													
SpaceT+ve	2	2	2	2	2	2	2	0	2	0	1	0	0	2
SpaceT-ve	2	2	2	2	2	2	0	0	1	1	0	0	2	2
PCB_V_0	2	0	0	0	2	2	-1	2	1	-1	-1	1	1	2
PCB_V_50	2	2	2	2	2	2	-1	2	2	-1	0	2	0	2
PCB_V_100	2	2	0	0	2	2	-1	2	1	1	1	2	0	2
SWT+9	2	2	2	2	2	2	0	2	2	0	-1	1	1	2
SWT-9	2	2	2	0	2	2	2	2	2	2	1	2	-1	2
Pump_50	2	0	0	2	2	2	-1	0	1	-1	2	2	2	2

The resultant matrix derived from the preceding step is referred to as the sensor matrix, denoted as S. Assuming that S(x) represents a vector signifying the number of features in each row of the sensor matrix S, the task of identifying the most suitable sensor placement can be framed as a constrained optimization problem, as outlined in equation 3 and elaborated upon in the research conducted by (Fijany & Vatan, 2006).

$$\text{minimize} \sum_{i=1}^{m} S(x_i) \text{ Constraint: } \quad \widetilde{M}x \geq 1 \quad , x_i = 0 \text{ or } 1 \tag{3}$$

The feature selection was done with 14 initial features, and it resulted in the selection of four features presented in 6. In 6, all the rows and columns are non-identical, meaning the combination of the features will be able to distinguish all the faults.

Table 6. List of features selected with the final signature (the rows are the faults and the columns are the selected features which are Δf, CUSUM_ ΔP, CUSUM_ ΔF, CUSUM_ ΔC)

Faults	Selected features			
SpaceT+ve	2	1	2	1
SpaceT-ve	2	1	1	2
PCB_V_0	-1	1	-1	-1
PCB_V_50	-1	2	-1	1
PCB_V_100	-1	1	1	1
SWT+9	2	2	2	-1
SWT-9	2	2	2	1
Pump_50	-1	1	-1	2

Result analysis and discussion

A total of 9 faults were implemented during the winter season among which 8 faults were used as known faults. One of the faults (the Cold water valve stuck at a closed position) was treated as a new fault. A 5-mins interval dataset was used throughout the model training and testing. The dataset for normal operation and 8 known faults was divided into training and testing datasets in the ratio of 7:3. All the new fault data was used for testing the GMR model.

The selection of the threshold is based on the occurrence of false alarms, which refer to instances where a normal sample is wrongly identified as faulty. The second criterion for the selection of threshold is based on the accuracy of prediction of the normal operation data during testing. The third criterion for the threshold selection is using fault prediction accuracy during testing. Table 7 presents the false alarm ratios associated with different baseline thresholds, defined as the ratio of false alarm samples to the total number of normal condition samples tested. Based on the results shown in Table 7, an uncertainty of '2sd' was found to be suitable for the system with a false alarm ratio of 2.5% and fault prediction accuracy of 97.5%. Although the uncertainty of '3sd' had a lower false alarm rate, the fault prediction accuracy plummeted to 82.9%. In contrast, an uncertainty of '0sd' led to an unacceptably high false alarm ratio of 22.9%. Hence the uncertainty of 2sd was used for all subsequent fault training and testing.

Table 7. Results of false alarm ratio and accuracy for different baseline thresholds for normal operation period (M=Maximum feature value during baseline period, m=minimum feature value during baseline period, sd = standard deviation of feature during baseline period)

Upper baseline limit	Lower baseline limit	False alarm ratio (%)	NO prediction Accuracy (%)	Fault prediction accuracy
M	m	22.9	77.1	92.4
M+sd	m-sd	6.6	93.4	92.9
M+2×sd	m-2×sd	2.5	97.5	97.5
M+3×sd	m-3×sd	0.8	99.2	82.9

The FDD method demonstrates promising performance in the diagnosis of known faults as the accuracy of the true diagnosis is 100% for 7 faults, 94.4% for the SpaceT+ve fault, and 97.5% for the normal operation Also, 100.0% of samples corresponding to the new fault were identified as an unknown state. Upon recognition of the type of the new fault by an expert, the data associated with the new fault was labeled and fed to the evolving mechanism to update the Gaussian mixture regression model in the diagnosis section. Hyperparameter values that resulted in the lowest prediction error were selected while evolving the GMR model. The values for the learning rate and closeness threshold used were 0.01 and 10 respectively. Before evolving the model, the new fault data fell into one of the 16 existing Gaussian components. In the evolved diagnosis model, three new Gaussian components (17, 18, and 19) were added to the model which had the information about the new fault.

The performance of the evolved FDD mode was evaluated using the same dataset used to test the model before evolving. The results shown in Figure 5 confirm that the FDD model remains stable after being evolved since the accuracy of the true diagnosis is between 94.4% and 100.0% for known faults. The evolved model also exhibits excellent performance in the diagnosis of the new fault as the accuracy of the true diagnosis for the new fault is 100.0%.

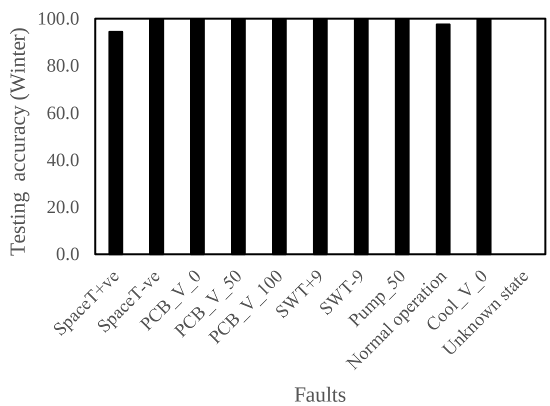

Figure 5. Fault prediction accuracy during the model testing for the winter season after the model evolving

The FDD model was also tested with fault data collected during the spring and summer seasons. A total of 9 faults were implemented for both seasons consisting of both known and unknown faults. First, the FDD model was tested using the spring testing data without evolving the model followed by testing the model with the same testing data with the model evolving. The result of this testing is shown in Figure 6. When tested without evolving the model, seven out of nine faults are predicted with accuracy above 87%. The SpaceT-ve fault was predicted at an accuracy of 72.4% and the SpaceT+ve fault was predicted at an accuracy of 82.5%. We can see that about half of the SpaceT+ve fault was misdiagnosed as an unknown state. This was because of the change in one or more of the feature values due to the change of the season. The signature matrix for diagnosis of SpaceT+ve fault is [2 1 2 1]. However, during the spring season, the signature of the feature changed resulting in additional signature matrices [2 1 2 2] and [2 1 2 -1]. This signature was not encountered by the model while training. Hence, evolving the model to incorporate the newly gained information about the change in signature of the SpaceT+ve fault was necessary for the correct diagnosis of the fault. Similar characteristics were observed for the other misdiagnosed faults as well. When the new signature for the existing fault was discovered and the model was evolved with the new information for each fault, we can see the improvement in the diagnosis of the fault in Figure 6. After evolving the model, the prediction accuracy of the SpaceT+ve fault increased to 92.5% and the true diagnosis of all the remaining faults is between 92.0% and 100.0%.

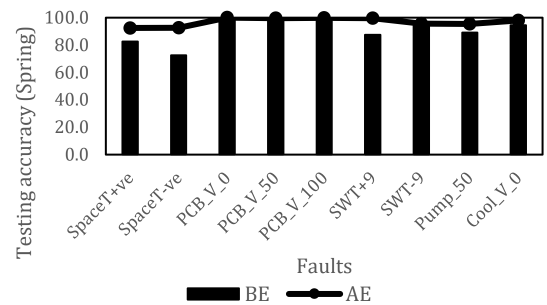

Figure 6. Fault prediction accuracy during the model testing for the spring season

Next, the FDD model, after evolving with the spring data was tested with the fault data collected during the summer season. First, the model was tested using the summer testing data without evolving the model followed by testing the model with the same testing data with the model evolving. The result of this testing is shown in Figure 7. When tested without evolving the model, five out of nine faults are predicted with an accuracy of 100.0%, and a few faults are misdiagnosed with a prediction accuracy of around 80%. The signature matrix for diagnosis of SpaceT+ve fault is [2 1 2 1]. But as the season changed to summer, one of the features was impacted by changing the signature matrix during summer to [2 1 2 -1]. This signature was not seen during model training. Hence, evolving the model to incorporate the newly gained information about the change in signature of the SpaceT+ve fault was necessary for the correct diagnosis of the fault. When the new signature for the existing fault was discovered and the model was evolved with the new information for each fault, the fault prediction accuracy increased which is seen in Figure 7. After evolving the model, the prediction accuracy of the SpaceT+ve fault increased to 99.5% and the true diagnosis of the remaining faults is between 92.4% and 100.0%.

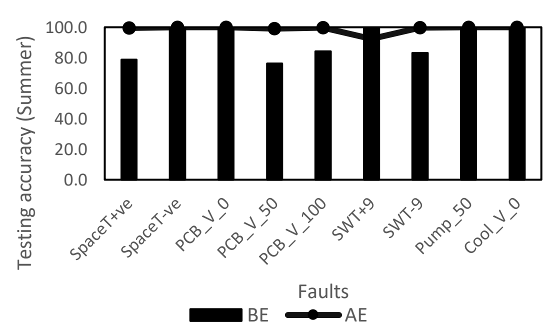

Figure 7. Fault prediction accuracy during the model testing for the summer season

Conclusion

The focus of this study was to develop a learning-based FDD model for PCB systems that can learn a new state and continuously update the model whenever new information is encountered. An experimental study was conducted to investigate the performance of the learning-based GMR model in fault detection and diagnostics of faults. A total of 5 common fault types of different severities (9 total faults) were implemented and were divided into known and unknown faults. A GMR model was developed and trained with normal operation and 8 known faults data and was evolved using new fault data. The performance of the model was tested using both known and new faults before and after evolving the model. The prediction accuracy for each fault using any dataset was extremely high. The algorithm was successful in detecting the new fault as an unknown state before evolving the model and in diagnosing it as a new fault after evolving the model. The accuracy of diagnosing the known faults did not decrease in any after evolving the model. Additional testing was done using the faulty data from the spring and summer seasons. While testing the model with the spring and summer data without evolving the model, the number of faults was misdiagnosed because the signature of the fault had already changed from winter to spring and summer seasons. The static model was not able to account for this change in the fault signature. However, testing the same data while evolving the model, the model incorporated the new signature for the faults and was able to diagnose the faults with very high accuracy. The lowest fault prediction accuracy for the spring and summer seasons were 92.5% and 92.4% respectively.

References

Bankó, Z., Abonyi, J. 2012. Correlation based dynamic time warping of multivariate time series. *Expert Systems with Applications*, *39*(17), 12814-12823.

Bouchachia, A., Vanaret, C. 2011. Incremental learning based on growing gaussian mixture models. 2011 10th International Conference on Machine Learning and Applications and Workshops,

Chakraborty, D., Elzarka, H. 2019. Early detection of faults in HVAC systems using an XGBoost model with a dynamic threshold. *Energy and Buildings*, *185*, 326-344.

Costa, B. S. J., Angelov, P. P., Guedes, L. A. 2014. A new unsupervised approach to fault detection and identification. 2014 International Joint Conference on Neural Networks (IJCNN),

Fabrizio, E., Corgnati, S. P., Causone, F., Filippi, M. 2012. Numerical comparison between energy and comfort performances of radiant heating and cooling systems versus air systems. *HVAC&R Research*, *18*(4), 692-708.

Feng, J. D., Chuang, F., Borrelli, F., Bauman, F. 2015. Model predictive control of radiant slab systems with evaporative cooling sources. *Energy and Buildings*, *87*, 199-210.

Fijany, A., Vatan, F. 2006. A new efficient algorithm for analyzing and optimizing the system of sensors. 2006 IEEE Aerospace Conference,

Gao, Y., Liu, S., Li, F., Liu, Z. 2016. Fault detection and diagnosis method for cooling dehumidifier based on LS-SVM NARX model. *International Journal of Refrigeration*, *61*, 69-81.

Hao, X., Zhang, G., Chen, Y., Zou, S., Moschandreas, D. J. 2007. A combined system of chilled ceiling, displacement ventilation and desiccant dehumidification. *Building and Environment*, *42*(9), 3298-3308.

Jeong, Y.-S., Jeong, M. K., Omitaomu, O. A. 2011. Weighted dynamic time warping for time series classification. *Pattern recognition*, *44*(9), 2231-2240.

Joe, J., Karava, P. 2019. A model predictive control strategy to optimize the performance of radiant floor heating and cooling systems in office buildings. *Applied Energy*, *245*, 65-77.

Karami, M., Wang, L. 2018. Fault detection and diagnosis for nonlinear systems: A new adaptive Gaussian mixture modeling approach. *Energy and Buildings*, *166*, 477-488.

Katipamula, S., Brambley, M. R. 2005. Methods for fault detection, diagnostics, and prognostics for building systems—a review, part I. *Hvac&R Research*, *11*(1), 3-25.

Khan, Y., Khare, V. R., Mathur, J., Bhandari, M. 2015. Performance evaluation of radiant cooling system integrated with air system under different operational strategies. *Energy and Buildings*, *97*, 118-128.

Kim, J., Tzempelikos, A., Braun, J. E. 2015. Review of modelling approaches for passive ceiling cooling systems. *Journal of Building Performance Simulation*, *8*(3), 145-172.

Kim, J., Tzempelikos, A., Braun, J. E. 2019. Energy savings potential of passive chilled beams vs air systems in various US climatic zones with different system configurations. *Energy and Buildings*, *186*, 244-260.

Lai, I. 1995. Chilled ceilings: the market potential.

Laouadi, A. 2004. Development of a radiant heating and cooling model for building energy simulation software. *Building and Environment*, *39*(4), 421-431.

Liang, J., Du, R. 2007. Model-based fault detection and diagnosis of HVAC systems using support vector machine method. *International Journal of refrigeration*, *30*(6), 1104-1114.

Nakamura, T. A., Palhares, R. M., Caminhas, W. M., Menezes, B. R., de Campos, M. C. M., Fumega, U., . . . Lemos, A. P. 2017. Adaptive fault detection and diagnosis using parsimonious Gaussian mixture models trained with distributed computing techniques. *Journal of the Franklin Institute*, *354*(6), 2543-2572.

Riffat, S., Zhao, X., Doherty, P. 2004. Review of research into and application of chilled ceilings and displacement ventilation systems in Europe. *International Journal of Energy Research*, *28*(3), 257-286.

Salvador, S., Chan, P. 2007. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, *11*(5), 561-580.

Sung, H. G. 2004. *Gaussian mixture regression and classification*

Tian, Z., Love, J. A. 2009. Energy performance optimization of radiant slab cooling using building simulation and field measurements. *Energy and Buildings*, *41*(3), 320-330.

Van Every, P. M., Rodriguez, M., Jones, C. B., Mammoli, A. A., Martínez-Ramón, M. 2017. Advanced detection of HVAC faults using unsupervised SVM novelty detection and Gaussian process models. *Energy and Buildings*, *149*, 216-224.

Wang, H., Chen, Y., Chan, C. W., Qin, J., Wang, J. 2012. Online model-based fault detection and diagnosis strategy for VAV air handling units. *Energy and Buildings*, *55*, 252-263.

Wang, L., Braun, J., Dahal, S. 2023. An evolving learning-based fault detection and diagnosis method: Case study for a passive chilled beam system. *Energy*, *265*, 126337.

Wang, L., Haves, P. 2014. Monte Carlo analysis of the effect of uncertainties on model-based HVAC fault detection and diagnostics. *HVAC&R Research*, *20*(6), 616-627.

Yan, K., Huang, J., Shen, W., Ji, Z. 2020. Unsupervised learning for fault detection and diagnosis of air handling units. *Energy and Buildings*, *210*, 109689.

Yu, Y., Woradechjumroen, D., Yu, D. 2014. A review of fault detection and diagnosis methodologies on air-handling units. *Energy and Buildings*, *82*, 550-562.

Zhao, Y., Wen, J., Wang, S. 2015. Diagnostic Bayesian networks for diagnosing air handling units faults–Part II: Faults in coils and sensors. *Applied Thermal Engineering*, *90*, 145-157.

Zhao, Y., Wen, J., Xiao, F., Yang, X., Wang, S. 2017. Diagnostic Bayesian networks for diagnosing air handling units faults–part I: Faults in dampers, fans, filters and sensors. *Applied Thermal Engineering*, *111*, 1272-1286.

Zhu, X., Du, Z., Jin, X., Chen, Z. 2019. Fault diagnosis based operation risk evaluation for air conditioning systems in data centers. *Building and Environment*, *163*, 106319.