Enhancing operational performance of AHUs through an advanced fault detection and diagnosis process based on temporal association and decision rules

Enhancing operational performance of AHUs through an advanced fault detection and diagnosis process based on temporal association and decision rules

Marco Savino Piscitelli[a], Daniele Mauro Mazzarelli[a], Alfonso Capozzoli[a*]

[a] *Dipartimento Energia "Galileo Ferraris", Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy*

* Corresponding author: Tel: +39-011-090-4413, fax: +39-011-090-4499, e-mail: alfonso.capozzoli@polito.it

Abstract

The pervasive monitoring of HVAC systems through Building Energy Management Systems (BEMSs) is enabling the full exploitation of data-driven based methodologies for performing advanced energy management strategies. In this context, the implementation of Automated Fault Detection and Diagnosis (AFDD) based on collected operational data of Air Handling Units (AHUs) proved to be particularly effective to prevent anomalous running modes which can lead to significant energy waste over time and discomfort conditions in the built environment. The present work proposes a novel methodology for performing AFDD, based on both unsupervised and supervised data-driven methods tailored according to the operation of an AHU during transient and non-transient periods. The whole process is developed and tested on a sample of real data gathered from monitoring campaigns on two identical AHUs in the framework of the Research Project ASHRAE RP-1312. During the start-up period of operation, the methodology exploits Temporal Association Rules Mining (TARM) algorithm for an early detection of faults, while during non-transient period a number of classification models are developed for the identification of the deviation from the normal operation. The proposed methodology, conceived for real-time implementation, proved to be capable of robustly and promptly identifying the presence of typical faults in AHUs.

Keywords: HVAC systems; Air Handling Units; Fault Detection and Diagnosis; Temporal Association Rules Mining; Intelligent energy management

Highlights

- A fault detection and diagnosis process is applied on AHU monitoring data;
- A novel methodology tailored on transient and non-transient operation is proposed;
- Faults in the transient period are detected with multivariate association rules;
- Temporal associations are exploited during the start-up period of the system;
- Decision rules are extracted for fault diagnosis in non-transient regime of AHU;

1 Introduction

Recent years have seen an increasing interest of the scientific community in exploring solutions to improve energy efficiency in buildings by implementing advanced data-analytics based energy management strategies. The application of these strategies is supported by the increasing penetration of ICT (Information and Communication Technologies) and EMSs (Energy Management System) in buildings, which may enable the adoption of data analytics based procedures for the exploitation of collected energy-related data and the extraction of hidden knowledge in an automatic way [1]. Building Energy Management System (BEMS) are mainly used for tracking and managing the operation and energy performance over time of Heating Ventilation and Air Conditioning (HVAC) systems. The optimal management of HVAC systems, which accounts in the developed countries for 10-20% of the total energy share in buildings [2], is a crucial task, considering that such systems account for 50% of the energy demand in commercial buildings [3].

However, due to lack of proper maintenance, failure of components or incorrect installation, Air Handling Units (AHUs) are often run in inappropriate operational conditions. A study conducted on more than 55.000 AHUs, showed that a fraction of 90% of them runs with one or multiple faults [4], where a fault is intended as an abnormal system state, an unpermitted deviation of at least one characteristic property of the system from the acceptable, usual, standard conditions. The identification and diagnosis of faults, in the case of HVAC systems, can lead to potential savings of about 30% [5]. This process is also known as Fault Detection and Diagnosis (FDD) where *fault detection* consists in the recognition of a fault occurrence, and *fault diagnosis* corresponds to the identification of the causes and the location of the fault [6]. Advanced methods of *fault detection* are based on mathematical models and on methods of system and process modelling to generate fault *symptoms* (e.g. residuals). *Fault diagnosis* methods use causal fault-symptom-relationships by applying techniques from statistical decision, artificial intelligence and soft computing [6].

Although currently underutilized, FDD is a powerful tool for ensuring high efficiency in building operation and FDD products represent a very fast-growing market in the context of building analytics technologies [7]. According to [8], over 30 FDD products are available in U.S. that may be delivered through different implementation models [7]. The algorithms behind FDD tools may be integrated into server-based software, desktop software, or software directly embedded in equipment controllers. FDD algorithms are based on historical data that can be gathered from different sources such as Building Automation Systems (BAS), equipment controllers, external sensors and meters, or mixed sources. Despite the existing differences in the way tools are implemented and integrated with the monitoring system, the main tool classification can be performed according to the approach employed for conducting the FDD analysis.

The methods used for performing an FDD analysis can be classified in quantitative model-based, qualitative model-based and data driven-based, as done in [9].

The quantitative model-based approach includes all the methods involving engineering models with different levels of detail in the physical description of the system (e.g. white box models). The qualitative model-based methods exploit the system knowledge derived from domain expertise (e.g. rule-based, qualitative models). The last category includes data-driven methodologies exploiting collected operational data of the system under investigation (e.g. Artificial Neural Networks, Association Rules Mining, grey box models). While rule-based methodologies (qualitative approach) are largely used, vendors are beginning to use data driven methodologies for addressing FDD tasks [7].

The next section provides an overview of the use of data-driven approach for FDD, with a specific focus to AHU systems.

1.1 Data-driven approach for FDD analysis in AHUs

In the last few years, the data-driven approach for the FDD analysis gained more and more interest, thanks to its applicability even in the case engineering models of the building and systems are inadequate or difficult to be developed, or the physics-based knowledge is not wide enough [9]. In this context, particularly promising appears the implementation of machine-learning techniques, which include both supervised and unsupervised algorithms. As pointed out in [10], the main advantages of the artificial intelligence-based data-driven approach, in comparison to traditional approaches, rely on the opportunity to:

- Learn automatically patterns from system operational data without the use of physical models. The data-driven approach does not require an a-priori understanding of the relationships that exist among faults and their symptoms.
- Achieve higher fault-detection and fault-diagnosis accuracy than qualitative methods (rules based on expert knowledge), also for faults of low severity levels.
- Perform FDD analysis exploiting a limited number of variables. It means that approach can enable an optimisation of sensor installation and then significantly reduce the number of required sensors.

More in detail, the supervised approach uses the domain expertise to develop useful prediction tool, since monitored data include variables for both input and output of the model (i.e. regression or classification methods). On the other hand, the unsupervised methods (e.g., cluster analysis, association-rule mining) are capable to extract hidden knowledge without a pre-defined target (i.e. data used do not have any output values) and are particularly effective in case of poor-information systems or when the objective of the analysis is not a-priori constrained [11].

The methods involving the construction of supervised estimation models mainly consider the implementation of a residual analysis in order to perform an FDD process. The residual in that context is the difference between the estimated and the measured value of a specific target variable: the estimation is performed by means of a fault-free supervised model, while actual data may be related to faulty conditions. Therefore, the residual analysis is used as a way for assessing the severity of the deviation from the fault-free conditions during operation [12].

Many applications of supervised and unsupervised techniques for Automated Fault Detection and Diagnosis (AFDD) are reported in literature, particularly for detecting faults during the non-transient operation of AHUs [13].

Even though each component of an AHU can be potentially corrupted by a fault, the most common faults can affect sensors (e.g. offset in the measurement), controlled devices (e.g. blockage or leakage of air damper or coil valves), equipment (e.g. coil fouling or reduced capacity, duct leakage, fan complete failure or deviation in the pressure drop or belt slippage) and controllers (e.g. unstable or frozen control signal for dampers, coils or fan) [14]. In [15] was proposed a methodology to identify faults related to the fans and the air dampers of an AHU. The methodology uses a Multi-Class Support Vector Machine (MC-SVM), for the identification of both pre-labelled faults and new ones. In [16] and [17], a Bayesian Network (BN) was adopted for the diagnosis of faults related to air dampers, cooling coil valve stuck and return fan failure. The BN exploited in input the residuals obtained from a set of limit-checking rules and statistical models, capable of estimating air temperature, water flow rate, air flow rate and fan power consumption. Mulumba et al. proposed in [18] a methodology to diagnose the presence of several faults affecting air dampers, cooling coil valve and return fan by using a SVM in combination with an autoregressive model with exogenous inputs. Yan et al. proposed in [19] a combination of two supervised techniques to diagnose the blockage of air dampers and coil valve, the duct leakage and the return fan failure. In [19] a Classification Tree (CT) was developed, which used in input both monitored data (i.e. air temperature and flow rate, fan speed and power, and cooling coil valve position) and residuals obtained from a regression model of the fan speed, while in output the labels of different faults were considered. The methodology developed in [19] made it possible to accurately perform fault diagnosis, but without taking into account transient periods of operation. Different classification models for fault detection were also compared in the work of McHugh et al. [20] and the Classification (decision) Tree model was selected as the best choice for the detection of steam or chilled water leakage.

The unsupervised methods proved to be particularly flexible for their nature in exploring data set without any *a priori* constraint, as opposed to the supervised models [11].

Yu et al. proposed in [21] an unsupervised methodology to identify energy wastes and faults of a fan in an AHU, by exploiting Association Rules Mining (ARM). This type of algorithm requires a strong expertise by the analyst for the interpretation of the results, considering that the rule set extracted could include also not-interesting information for the identification of anomalous operation of the air conditioning system [21]. Many studies make use of ARM for the identification of faults in different types of HVAC systems (e.g. district heating substation, AHU, chillers)[10]. ARM has been adopted also for the analysis of a district heating substation in order to identify inefficient operation and sensor faults by searching anomalous correlation expressed by association rules [22]. In order to help the domain expert in the interpretation of the results, in [23] a methodology was proposed to reduce the number of rules to be analysed and to effectively group them for distinguishing the faulty from the normal operation. Furthermore, the temporal relation among the energy consumption of different HVAC components was studied in [24] and [25] to determine the presence of faults and prevent a reduction of energy performance over time.

A combination of a supervised and unsupervised methods (e.g., decision tree and clustering analysis) was proposed in [26] and [27] for the detection of anomalous energy consumption in a group of smart office buildings. Furthermore, Dey et al. achieved in [28] high values of accuracy in the automatic FDD on fan coil units operation by combining MC-SVM and cluster analysis.

Du et al. in [29] proposed a methodology to identify faults of temperature, flow rate and pressure sensors in a VAV system by implementing Artificial Neural Networks (ANNs) in combination with a signal decomposition technique (i.e. Wavelet analysis). In [30], an ANN was combined with clustering analysis to diagnose faults related to cooling coil valve, air damper and temperature sensors in an AHU. In the first step, the ANN was used for the estimation of supply air and water temperature to perform a residual analysis, then the methodology leveraged on clustering analysis for the fault diagnosis stage. Guo et al. used a Hidden Markov Model (HMM) for the fault detection phase and a cluster analysis for the identification of various types of faults such as the blockage of dampers, frozen fan or unstable cooling coil valve control signal [31]. In [32] and [33], an unsupervised data-driven approach was used to identify the presence of cooling coil valve blockage, heating coil valve leakage and air damper blockage, by analysing the error generated from the reduction of variables by means of Wavelet Transform and Principal Component Analysis. Successively, the fault diagnosis was performed by analysing the trend of each variable during faulty conditions, in order to identify the variable mostly influenced by the fault source.

Liang et al. in [34] proposed a methodology to diagnose the stuck of the recirculation damper and of the cooling coil valve in an AHU, as well as the decreasing of the supply fan speed. In that work, an SVM was used in combination with a white box model, exploiting the residuals obtained by comparing actual and simulated fault-free values of supply and mixed air temperature, and cooling coil outlet water temperature. Wu et al. in [35]combined a quantitative model-based method with an unsupervised data-driven method to diagnose sensor faults, air damper blockage or frozen fan. In that work, first the variables considered were reduced (i.e., by means of Principal Component Analysis), then the presence of faults was investigated comparing actual monitored data with the estimation of airflow rate and energy calculated by using simplified balance equations for energy and pressure-flow. In other works, a qualitative-based approach was used to perform automatic FDD in combination with the data-driven approach. In [36], the detection of faults occurring in an AHU was performed by exploiting “IF-THEN” expert rules related to the residuals of mixed air temperature, return air flow rate, supply air static pressure and cooling coil valve control signal, generated with different General Regression Neural Networks. In [37] the integration of expert rules with Bayesian Networks was pursued, in order to better isolate faults in AHU. Such approach made it possible to exploit the violation of expert rules, to better detect the co-occurrence of multiple faults at the same time.

The above reported literature review demonstrated how much the scientific research has been active in the field of artificial intelligence for FDD in AHU and HVAC systems. However, the opportunity to approach this well-known task (i.e., FDD) from this innovative point of view was mainly due to the growing availability of huge amount of monitored data, related to the actual performance of buildings and energy systems. In this context some projects, supported by the American Society of Heating, Refrigerating and Air- Conditioning Engineers (ASHRAE) made very comprehensive field surveys, laboratory tests and performance evaluations on the performance of HVAC systems also in faulty conditions. The outcomes of such projects (e.g., ASHRAE Project 1312-RP and 1043-RP) enabled a great spread of FDD methodologies which exploit experimental data.

Among the reviewed papers, several published studies focused on the ASHRAE RP-1312 data set for developing and testing FDD methodologies for AHUs [16,17,38–43]. Despite those papers discuss the results of FDD methodologies on the same data set, not always the assumptions behind the analysis are the same. The main differences are related to the operation mode considered (cooling, heating, spring), the number and the type of faults analysed, the regime of operation considered (transient, non-transient). However, from the analysis of these works, some general considerations can be made:

- In most of the cases the analysis is performed for the summer period achieving high values of accuracy in diagnosing faults (over 90% of accuracy),

- The analysis is performed for data collected with sampling frequency of 1-min (original granularity of the dataset),
- Data-driven models used for characterizing the normal behaviour of the AHU lack of interpretability (SVM, ANN)
- In most of the cases, the fault diagnosis is performed through interpretable classifiers (decision trees, Bayesian networks).

In this context, the present paper aims at introducing an FDD methodology for AHU systems (based on data of ASHRAE RP-1312) that is data-driven, fully interpretable and rule-based. Indeed, the rule-based approach can satisfy the user need of simplicity and interpretability while the data-driven nature of the methodology can enable the automatic learning of system operational patterns. Another objective is also to reduce the granularity of the dataset while maintaining good performance in fault diagnosis. In fact, analyse data with a high sampling frequency could expose the FDD tool to instabilities when deployed for operating in real time (presence of punctual anomalies, missing values, sensor network latency).

In the approach proposed in this paper two rule-extraction methods (association rule mining, decision tree) were employed for conducting FDD analysis in AHU system, by exploiting the reduction and transformation of multiple time series related to the operation variables of the system. In the next section a discussion is provided about the automatic extraction of rules in multiple time series (Section 210) and the work novelty is explained (Section 2.1); the case study analysed in the paper is presented in Section 3; a focus on data analysis methods exploited in the analysis is provided in Section 4 and a description of the proposed methodology is provided in Section 5. In Section 6, the results obtained from the application of the methodology are presented. Eventually in Section 7 the discussion of results and concluding remarks are provided.

2 Rule extraction in multiple time series for FDD in AHUs

The analysis reported in section 1.1 on the most relevant works published in the last few years, showed how FDD strategies in AHUs can widely benefit from the implementation of data mining and machine learning techniques, especially when they are supported by a robust expertise in building physics for an effective exploitation and interpretation of the extracted knowledge. However, the complexity of an AHU with multiple operational parameters and the temporal interactions among them makes the effective characterisation of its behaviour challenging.

The operation of an AHU system is characterized by two major time-regimes, namely transient and non-transient. The transient operation typically occurs when the AHU is started-up and is approaching the steady state conditions, or when it is shutdown or disturbed from its non-transient regime. The disturbances could be caused by either variation of thermal loads or by feedback controls. During transient periods some variables can exhibit strong variation in short time and a significant temporally lagged response with respect to the control signals. In addition, the behaviour of an AHU system varies as its mode of operation changes during the day and the year i.e., off mode, heating mode, free cooling mode, and cooling mode. Therefore, a robust data-driven-based FDD tool should be able to automatically determine the mode of operation of the system, to prevent false alarms from being generated. For example, normal behaviour during summer season may be faulty if the system is operating in heating mode (winter season). In order to avoid that condition, FDD tools in AHU systems are characterized by a hierarchical architecture that makes it possible to exploit only the portion of knowledge that is consistent with the specific operation mode considered. In this perspective, when using data-driven based FDD tools it is necessary for the training data to be exhaustive as possible for each operation mode.

However, given their complexity, data-driven-based FDD tools often lack in interpretability. In this context, the use of rule-based data-driven methods for FDD can satisfy the user need of simplicity in terms of understanding the FDD tool, using, commissioning and integrating it with existing BAS, and updating it. For this reason, great attention has been paid in this study to the application of advanced supervised and unsupervised rule extraction methods (i.e., decision trees, association rule mining) with reference to multivariate problems.

The operation of an AHU is a perfect case that can be effectively described through the analysis of multiple time series (defined as series data points indexed in time order) associated to each operational variable of the system. However, the large number of time series with high sampling frequency could significantly increase the complexity and computational cost of the analysis, often making necessary a proper reduction (aggregation in the time domain) and discretization (quantization of the signal value) of data. This is a challenging task, considering that each variable has its own behaviour and distribution and, as a consequence, the optimal time aggregation and value discretization of the signal need to be identified with the aim of minimizing the information loss and of maximizing the mining performance. Such preparation of the time series is an essential step in FDD methodologies based on rule extraction techniques (e.g., based on association-rule mining algorithms or decision trees) that, in the literature, have been used for effectively mining co-occurrences or implications between discrete values and events in the time domain during HVAC operation [19,24,44].

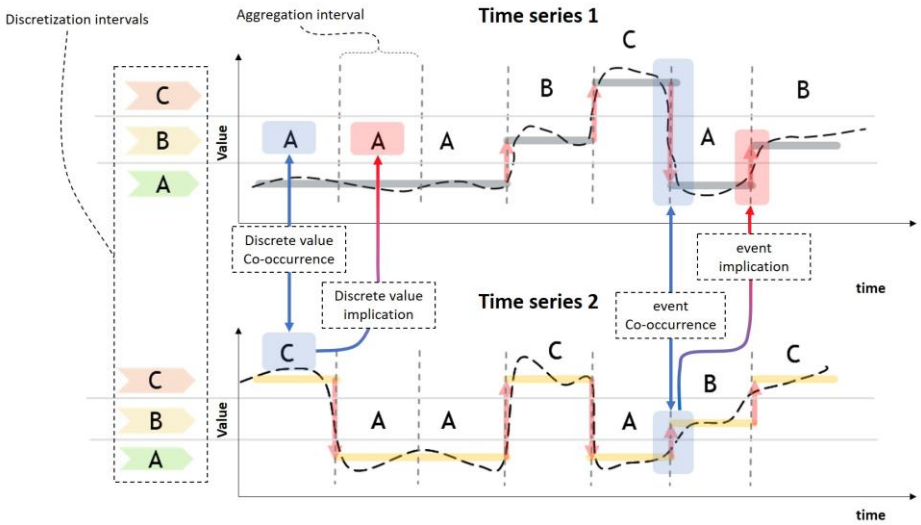

Figure 1. Graphical representation of co-occurrence and implication between discrete values and events among multiple time series.

Figure 1 depicts in graphical form the concepts, of discrete value, event (change of discrete value between two contiguous aggregation intervals), co-occurrence and implication of discrete values and events with reference to two time series encoded in symbols by means of Symbolic Aggregate approXimation (SAX) [45].

When multiple time series are considered, rule extraction techniques can be categorized in intra-transactional and inter-transactional respectively. The first type of extraction aims at discovering co-occurrences between discrete values and events that frequently happen at the same time among different time series (Figure 1). The second type of rule extraction is more complex, considering that the occurrences of discrete values and events among different time series, in that case, are searched taking into account the existence of a time lag (Figure 1). During transient periods of AHUs operation, the latter approach is particularly favourable in describing phenomena that are characterized by temporal dependences among variables representative of the system operation (e.g., change of status in a fan speed and the corresponding effect on supply air temperature).

In order to develop an FDD process capable to be flexible in relation to different conditions of operation in AHUs, the present study proposes the application of two rule extraction methodologies tailored for both transient and non-transient periods.

The developed framework aims at preventing anomalous running modes of AHUs, which could lead to significant energy waste over time and/or discomfort conditions in the built environment.

The analysis relies on temporal abstraction as a pre-processing stage. Temporal abstraction is aimed at reducing and transforming time series in discrete-time and discrete-value signals through aggregation on the time axis and discretization of the value in order to perform the extraction of interesting co-occurrences and implications. In this study, an adaptive process based on a Symbolic SAX is employed for conducting the temporal abstraction.

Furthermore, strong relations between events (i.e., change of discrete value between contiguous aggregation intervals) are automatically mined by means of temporal IF-THEN association rules in the transient period of AHU operation (i.e., start-up phase), considering an intra-transactional approach for characterizing the fault-free behaviour of the system. Similarly, during the non-transient period of operation, a set of classification trees are developed for extracting reference patterns in the form of decision rules. Potential faulty conditions are then detected when the discovered association and decision rules are violated over time. Successively, the identified anomalous patterns (during the non-transient period) are exploited for performing a diagnosis of the most probable faults associated to a specific kind of rule violation by means of a classification algorithm.

2.1 Novelty of the paper

The present work introduces an automatic methodology for performing an FDD analysis in AHUs by using experimental data obtained in the framework of the ASHRAE project RP-1312. The entire process relies on the application of data mining-based algorithms in order to develop a tool capable to detect and diagnose operational faults in AHUs which can determine energy waste over time and the occurrence of discomfort conditions in the built environment.

Based on the FDD literature review presented in section 1.1, the main innovative aspects introduced by the present paper are the following:

- An adaptive process of data reduction and transformation is employed to develop a robust FDD methodology. In complex systems as AHUs, the number of monitored variables and their sampling frequencies could be very high. Extracting only key information from large data set is essential for reducing redundancy, complexity and computational cost. In this study, the methodology makes it possible to achieve good performance in FDD (comparable to other studies focused on the same dataset [16,17,38–43]) leveraging only on the analysis of significant discrete intervals of the operational variables over time.
- The start-up period of AHU operation is isolated and treated separately by developing a tailored analytics module (instead of being filtered out as happened in other studies focused on the same dataset [16,17,38–43]). During transient period of operation time lags occur for example between a change of status in the fan speed and the corresponding effect on supply

air temperature. For this reason, temporal association rules are extracted, following an intra-transactional approach, for discovering associations between events during transient periods, across multiple time series, that frequently occurs within a time lag.

- The characterization of normal behaviour during the non-transient period is completely automated and performed by using a set of estimation models based on decision trees. In comparison to other studies focused on the same dataset [16,17,38–43], the reference behaviour of the AHU is evaluated estimating the most probable discrete value of each influencing operational variable in relation to all the others monitored. In that way, all the existing relations between variables are exploited through several estimation models, for detecting potential faulty conditions. Such approach exhibits high flexibility and generalizability in the formulation of the FDD problem.
- A fault diagnosis during non-transient period of AHU operation is performed by employing a decision tree, capable to extract rules for the classification of typical faults. The diagnosis process exploits the residuals evaluated by means of a set of estimation models as input attributes for the classification of the most probable faults.

In that perspective, this study was aimed at conceiving, developing and testing a methodological framework that introduces the aforementioned novelties in automatic FDD, in as robust a way as possible. As previously stated in the literature review, several studies considered the RP-1312 data set in the analysis, achieving an accuracy in fault diagnosing over 90%. As a consequence, the main objective of this study is not to improve the (already high) FDD performance achieved on the RP-1312 dataset, but rather to demonstrate the opportunity to achieve high performance as well through a fully interpretable and simplified data-driven approach, based on rule extraction techniques.

3 Case study

In order to test the validity and the effectiveness of the proposed methodology, operational data related to two AHUs collected in the framework of the Research Project ASHRAE RP-1312 [14] were analysed. The system investigated is a Variable Air Volume (VAV) AHU. A VAV system is able to modulate the air flow rate according to the variation of the building load and it is typically made up of 4 subsystem controllers, acting on supply air temperature, dampers and valves, supply air static pressure and return air flow rate. Specifically, the control logic maintains the supply air temperature set-point acting on damper and valve positions, according to the mode of operation (i.e. heating, cooling with partial mixing of outdoor air, cooling with 100% of outdoor air, cooling with minimum outdoor air).

Furthermore, also the static pressure of the supplied air and the difference between the supply and return air flow rate is controlled. The return air flow rate is modulated acting on the mixing dampers and the return fan speed, while the system maintains the static pressure set point for the supply air. As a result, the difference between the supply and return air flow rate is kept constant [12].

The dataset used in this paper is particularly interesting as it includes several running conditions for two AHUs in faulty and fault-free operation. The faulty operation was obtained by artificially implementing a number of different faults. The site, where the monitoring data have been collected, is a test facility simulating a typical schedule of occupancy in commercial building.

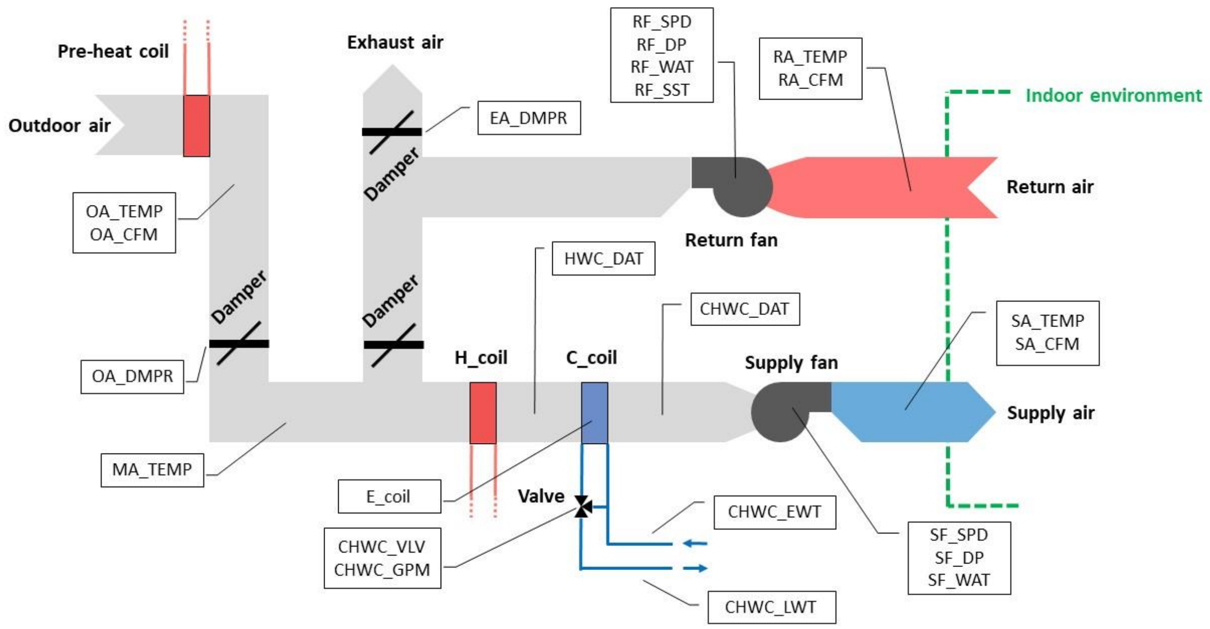

Figure 2. Scheme of the AHU analysed (refer to Table 2 for variable encoding).

The monitoring data were gathered from two AHUs of the facility (AHU-A and B), which are perfectly identical from technical and operational points of view and serve specular zones. The zones served by AHU-A and B face east and west orientation, respectively, in order to be comparable also under the aspect of the thermal loads. The AHUs are characterised by a mixing chamber, to mix return air with outdoor air by means of dampers. Each AHU is equipped with heating and cooling coils and VAV devices to locally adjust the supply air temperature. However, the control volume considered in this work excludes the local VAV devices.

Figure 2 shows a schematic configuration of the system with the indication of the monitored variables (a description of the variables is provided in Table 2).

In the context of the ASHARE project, a number of different faults were implemented one per time, each for a whole day, only in the AHU-A, in order to analyse independently the effects of each fault. The AHU-B was always run in fault-free conditions to have a reference of the normal operation. The data collection was conducted over three seasons and only the monitoring data of the summer season were considered in this study.

Table 1. Tags and descriptions of faults.

Fault Tag	**Description**	**Number of days**
CCVS15	Cooling coil valve stuck at 15%	1
CCVS65	Cooling coil valve stuck at 65%	1
CCVSFC	Cooling coil valve stuck fully closed	1
CCVSFO	Cooling coil valve stuck open	1
EASFC	Exhaust air damper stuck fully closed	1
EASFO	Exhaust air damper stuck fully open	1
Normal	Normal operation	22
OAS45	Outdoor air damper stuck 45%	1
OAS55	Outdoor air damper stuck 55%	1
OASFC	Outdoor air damper stuck fully closed	1
RFCF	Return fan complete failure	1
RFF30	Return fan at fixed speed (30%)	1

The dataset consists of multiple time series (one for each monitored variable) with a length of 33 days and a sampling time of 1 minute. In particular, 22 out of 33 days are tagged as fault-free days while the remaining 11 days correspond to different faulty conditions. Table 1 summarizes the number of fault-free and faulty days, the description of each fault and the tags used for labelling each day included in the monitoring campaign.

A feature selection was preliminarily performed on the basis of expert considerations to focus the analysis only on relevant variables.

As a result, the variables considered for the implementation of the FDD methodology are: the electrical load, the pressure drop and speed of fans, the flow rate and temperature of the air measured in different parts of the system, the damper position, the valve position, the water flow rate and energy transferred in the cooling coil.

Table 2 reports the list of the 23 variables considered for the analysis, together with the specification of labels, description, ID number and unit of measurement for each variable.

Table 2. List of variables considered in the analysis.

Variable	**Description**	**ID n°**	**Unit**
SF_WAT	Supply fan power	1	W
RF_WAT	Return fan power	2	W
SA_CFM	Supply air flow rate	3	m^3/h
RA_CFM	Return air flow rate	4	m^3/h
OA_CFM	Outdoor air flow rate	5	m^3/h
SA_TEMP	Supply air temperature	6	°C
MA_TEMP	Mixed air temperature	7	°C
RA_TEMP	Return air temperature	8	°C
HWC_DAT	Heating coil air temperature	9	°C
CHWC_DAT	Cooling coil air temperature	10	°C
SF_DP	Supply fan pressure drop	11	Pa
RF_DP	Return fan pressure drop	12	Pa
SF_SPD	Supply fan speed	13	%
RF_SPD	Return fan speed	14	%
OA_TEMP	Outdoor air temperature	15	°C
CHWC_EWT	Cooling coil input water temperature	16	°C
CHWC_LWT	Cooling coil output water temperature	17	°C
CHWC_GPM	Cooling coil water flow rate	18	m^3/h
E_ccoil	Cooling coil power	19	kW
CHWC_VLV	Cooling coil valve position	20	%
EA_DMPR	Exhaust air damper position	21	%
OA_DMPR	Outdoor air damper position	22	%
RF_SST	Return fan start/stop signal	23	-

For the application of the proposed methodology, the data sample was split into two datasets. The first one was used for the characterization of the normal operating condition of the system, while the latter was used for the fault detection and diagnosis. The first dataset is composed of 20 days tagged as “Normal” (training dataset), while the second by the rest of the days including 2 “Normal” days and 11 “Faulty” days (testing dataset).

4 Description of the data analysis methods

In this section, the overview is presented of the techniques used for the proposed FDD methodology, describing their main features in relation to the FDD problem under investigation.

4.1 Adaptive symbolic aggregate approximation

The Symbolic Aggregate approXimation (SAX) is a temporal abstraction technique capable to reduce the dimension of a time series of length n in a time series of length m with m < n, and to transform it in a symbolic string. The reduction of the time series is performed through a Piecewise Aggregate Approximation (PAA), which segments the time axis in equally sized non-overlapping time windows (i.e., aggregation intervals). The PAA approximates the original time series replacing the values within the same aggregation interval with their mean value.

The transformation of the time series is then performed by substituting the values of the PAA with symbols. To this purpose, the y-axis is discretized in a pre-defined number of regions and a symbol is associated to each of them. Lin et al. in [46] proposed a simple procedure to perform the SAX, employing a Z-score transformation (i.e., $Z(t) = \frac{x\,(t) - \mu}{\sigma}$ where μ is the mean value of the sample and σ the standard deviation) before the data reduction and identifying the desired range of each symbol assuming a-priori a Gaussian distribution of data.

A variation of SAX technique, called adaptive Symbolic Aggregate approXimation (aSAX), was proposed in the literature [45]in order to improve the quality of the discretization in the case of non-normal distribution of data. The adaptive symbolic aggregate approximation introduced by Pham et al. [45] is based on the original SAX method, but an adaptive process is used for the identification of breakpoints (i.e. the position of boundaries of each symbol range). The position of the adaptive breakpoints is evaluated through a univariate clustering procedure, that minimises the total representation error after the SAX transformation.

In this paper, the aSAX algorithm has been employed to improve the results of the automated and unsupervised discretization process, which represents the most important analysis for ensuring the robustness of the "event" extraction from time series.

4.2 Temporal Association Rules Mining

Association Rule Mining (ARM) is an unsupervised data mining method for identifying all associations and correlations between attribute values in a set of categorical/discretized data [47]. The output is a set of association rules that are used to represent patterns of attributes that are frequently associated together (i.e., frequent patterns).

Let $I = \{i_1,\ i_2, \dots i_d\}$ be the set of all items in a dataset and $D = \{d_1,\ d_2, \dots d_d\}$ be the set of all transactions. Each transaction d_i contains a subset of items chosen from I. In association analysis, a collection of items is named *itemset* and the transaction width is defined as the number of items present in a transaction. A transaction d_j contains an itemset X if X is a subset of d_j. An important property of an itemset is its support count, that corresponds to the number of transactions that contain a specific itemset. The support count, σ(X), for an itemset X can be expressed as follows [47] (eq.1)

$$\sigma(X) = |\{d_i | X \subseteq d_i, d_i \in D\}| \tag{1}$$

Association rules are usually represented in the form X → Y, where X (also called antecedent) and Y (also called consequent) are disjoint item sets (i.e., X ∩ Y = ∅). Rule quality is usually measured through rule support and confidence. Rule support is the fraction of the total number of transactions in which both the item sets X and Y occur while confidence determines how frequently items in Y appear in transactions that contain X. According to [47], Support $s(X \rightarrow Y)$ and Confidence $c(X \rightarrow Y)$ can be calculated with the following equations (eq. 2 and 3):

$$\text{Support}, s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N} \tag{2}$$

$$\text{Confidence}, c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} \tag{3}$$

where N is the total number of transactions. Therefore, given a dataset D, the generic record of which is a set of items, an ARM process discovers all association rules with support and confidence greater than, or equal to, minimum thresholds defined a-priori by the analyst (i.e, MinSup and MinConf).

In the context of discrete-value-transactions, association rules can be used as an efficient method for mining co-occurrences or implications also between events in the time domain (Temporal Association Rule Mining (TARM)). TARM is an extension of sequential pattern mining, which is an important data mining method with broad applications, capable to extract frequent itemset sequences while maintaining their order. Many sequential pattern mining algorithms, such as GSP [48], PrefixSpan [49,50], SPADE [51], and SPAM [52], have been proposed. However, those sequential pattern mining algorithms consider only the itemset occurrence order, but do not consider the time intervals between successive item sets (temporal constraint of event association). To that purpose, in the literature several sequential pattern mining algorithms were proposed, to deal with the extraction of sequential patterns considering the existence of interval between item sets (in terms of item gap and time interval) [53–56]. Such algorithms extract rules, satisfying not only user-specified minimum support constraints, but also user-specified gap constraints. The minimum and maximum gap values should be defined as constraints by the user. For those rules, the search space in the time domain is represented by a sliding window, the length of which is set in advance by the analyst. In detail, this kind of rules can be represented in the following form: $X \xrightarrow{t} Y$. Therefore, the occurrence of the antecedent itemset X implies the occurrence of the consequent itemset Y within a time *t*.

In this paper, the extraction of temporal association rules is performed by means of the cSpade, algorithm based on [51]. The algorithm was implemented in R [69], including the rule extraction phase which was performed by using the “cSpade” function of the “arules” package [70].

That algorithm extracts sequential rules, considering some constraints defined by the user according to his/her needs. The constraints may drive the mining of frequent patterns from the database of transactions, for instance by setting the length of the sliding window, or a minimum time gap between antecedent and consequent of the rules.

However, since the database of transactions considered in the present study is generated by using a sample-by-sample sliding window approach, the number of the transactions N results to be very high with items mostly overlapped. For this reason, the calculation of rule support s(X→Y) cannot be performed with the canonical formulation. In fact, the value of support s(X→Y) calculated according to eq. 2 can be affected by the high value of the denominator (i.e., the total number of transactions), suggesting the use of a formulation less sensitive to the sample size [57].

In this study, according to [58], the support of an association rule is defined as the ratio between the number of transactions that include both antecedent and consequent, and the number of transactions that include at least the consequent itemset (eq. 4).

$$\text{Support}, \mathrm{s}(\mathrm{X} \rightarrow \mathrm{Y}) = \frac{\sigma(X \cup Y)}{\sigma(Y)} \tag{4}$$

The support calculated with eq. 4 has the denominator dramatically lowered in comparison to the one in eq. 2 and makes it possible to have high values of support also for large transaction datasets, obtained through a sliding window. The support calculated through Eq. (4) assesses the frequency of X ∪ Y on a smaller portion of the total number of transactions (i.e., only the transactions that include the consequent itemset Y). The support is in the range (0-1) and allows an easier extraction of rules to be assumed as reference patterns (i.e., with high support) of the occurrence of a specific condition over time (i.e., consequent itemset Y). However, the confidence can be still calculated according to

Eq. (3) only if the consequent itemset Y occurs in a transaction not violating the chronological order respect to the antecedent itemset X.

In general, the mining of association rules can be summed up as a two-step's procedure. In a first phase, the frequent itemset with a support greater than the MinSup are extracted, then the confidence is considered for filtering out rules that consist in weak implications [59]. In this paper, the same two-steps procedure is followed but additional metrics are also considered in the rule filtering phase (as discussed in section 5.3).

4.3 Classification And Regression Tree

Decision trees are machine-learning algorithms that are used to develop descriptive and/or predictive models from a collection of records. Each record can be expressed as a tuple (**x**,y), where **x** represents the explanatory attribute set while y is the target attribute. The type of target attribute is the key factor that distinguishes classification from regression trees (i.e. discrete attribute in the first case and continuous attribute in the second one) [47]. In this work, the Classification And Regression Tree (CART) algorithm, based on recursive partitioning algorithm [60], has been selected to conduct a predictive modelling task, as it is able to easily handle categorical attributes as both explanatory and target attributes. The CART is a specific machine learning technique that is based on a recursive binary splitting of the whole feature space into finite disjoint sets, and its output can be translated into a hierarchical tree structure composed by nodes and directed edges (i.e., branches). The leaves (i.e., final nodes) represent the predicted class labels of the target attribute, while the branches represent the conjunctions of the explanatory attributes that lead to the class labels

The development of a classification tree unfolds over two steps: training and testing of the model.

Each decision tree developed has been pruned following a cost-complexity approach and validated through a k-fold cross validation process as explained in [61]. The development of a classification tree unfolds over two steps: *training* and *testing* of the model. Firstly, all the records are grouped in the root node and the CT algorithm iteratively evaluates the best partitioning of the dataset, using the explanatory attribute that minimises the average impurity measure (e.g., Gini index, Entropy) of the child nodes after each split. If no stopping rules are set by the analyst, the classification tree grows continuously until the impurity in the leaf nodes of the target variable is zero. In order to avoid this condition of model overfitting, various types of appropriate early stopping criteria can be set in advance by the analyst (e.g., minimum number of cases in parent and child nodes, maximum tree depth, minimum reduction in node impurity after splitting). Even when the early stop criteria have been satisfied, the tree may continue to be quite large and/or complex completely losing its interpretability. For this purpose, it is possible to define a cost-complexity parameter (*cp*) during the model validation phase, for optimising the trade-off between the cost of misclassification and the tree complexity. Therefore, the *cp* allows the analyst to set the right tree size by pruning branches and leaf nodes that do not significantly increase the model performance.

Starting from the fully grown tree, the cost-complexity pruning procedure is repeated iteratively, and smaller and smaller subtrees are found until the root node is reached. At the end of the iterations, the final pruned tree can be evaluated by plotting the relative errors of the subtrees versus their *cp* values. This kind of plot usually shows an initial sharp drop, followed by a relatively flat region. When the decision tree is subject to a validation procedure (e.g., k-fold cross-validation), it is also possible to compute a standard error for each relative error of the sub-tree. The choice of the best subtree starts from the flat region of the subtree errors that includes the minimum cross validated error that has been achieved. In fact, the values falling within one standard error of the achieved minimum risk (i.e., 1-SE rule) identify statistically equivalent sub-trees [60]. The simplest model (with the minimum number of final nodes) of all the identified sub-trees in the flat region is then chosen.

K-fold cross-validation has been used in this paper. For this kind of method, the original sample of data with M objects is divided into k equal sized subsamples. A single subsample is selected for the evaluated k subsamples as a validation dataset for testing the model, and the remaining $(k-1)$

subsamples are used for the training. This process is then repeated *k* times, using a subsample at a time for the testing

In this paper, all the classification trees developed for extracting decision rules have been subjected to the previously described procedure of validation and pruning (as discussed in section 5.4).

5 Methodological framework of analysis

The methodology relies on the application of both supervised and unsupervised algorithms to perform robust fault detection and diagnosis in AHUs.

The framework unfolds over different stages as shown in Figure 3. Two different analytics modules are proposed for developing an FDD tool tailored for both transient and non-transient conditions of the AHUs operation. For that purpose, in the methodological framework, a data segmentation phase is preliminarily carried out in order to split the data according to the regime of operation they belong to (i.e. transient or non-transient). In the following sections the pre-processing analysis, applied to the entire dataset, and the two analytics modules, tailored for transient and non-transient periods, respectively, are then described.

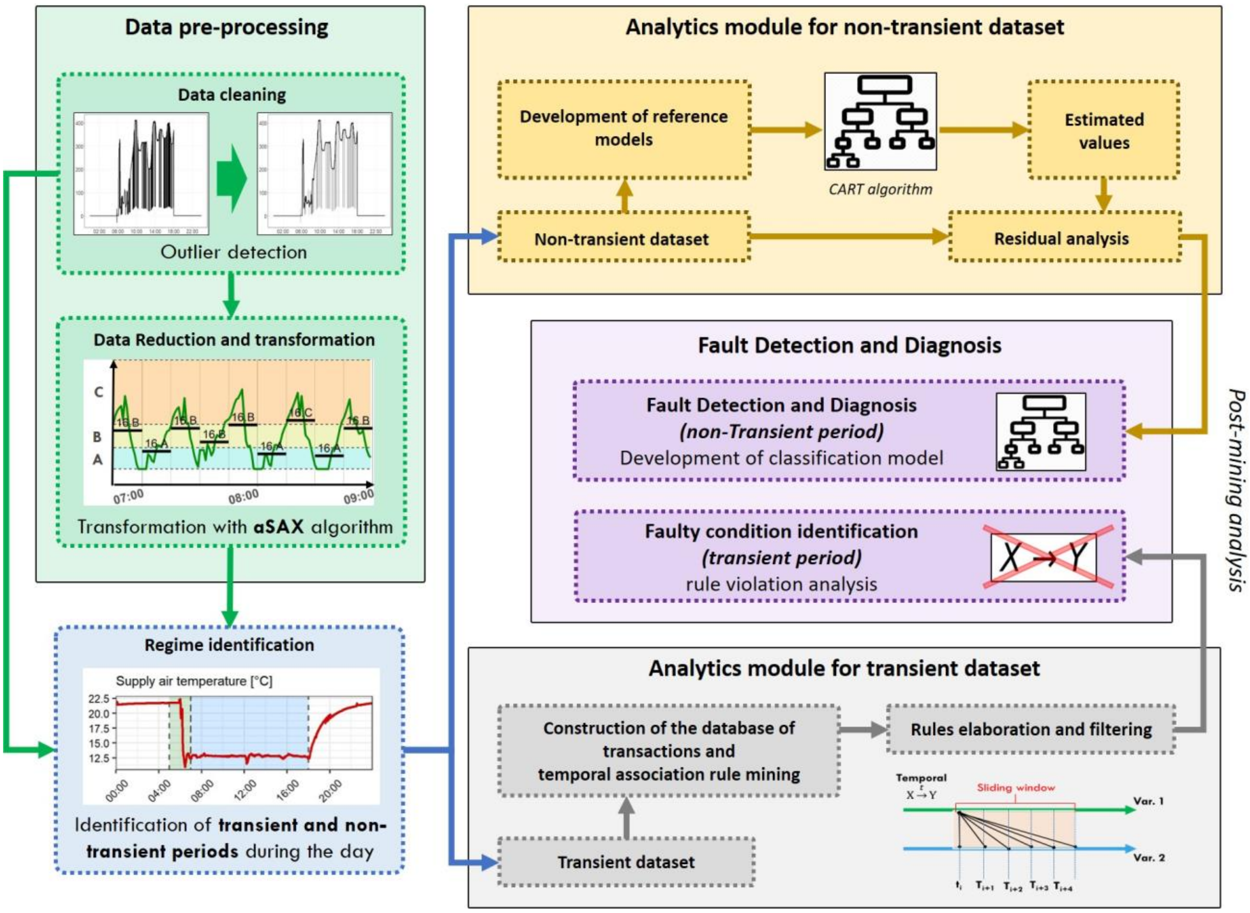

Figure 3. General framework of the analysis.

5.1 Data pre-processing stage

The pre-processing stage consists of three main tasks i.e., cleaning, reduction and transformation, typically accomplished for preparing the data sets. In detail, outlier detection and replacement are firstly performed (for each time series) by using the Hampel filter method [62]. For each data point in the time series, the algorithm computes the median of a window that includes the considered data point and its k surrounding samples. If a data point differs from the median by more than a standard deviation, it is tagged as a statistical outlier and replaced with the median.

The monitoring data were available in time-series with a sampling time of 1-minute, which would make the analysis onerous to be performed. For this reason, in a successive step a data reduction and transformation process is performed by means of the adaptive Symbolic Aggregate Approximation (aSAX) method [45]. This algorithm is employed for reducing the time series through a piecewise technique aggregating data with a fixed length window from 1 minute to 15 minutes and then for transforming it into a symbolic string. The objective is to maximise data compression and minimise the complexity of the time series while preserving important information. The symbolic representation of time series is always subjected to information loss due to the piecewise aggregate approximation (especially information about the slope). However, when the segments are encoded in symbols it is possible to preserve qualitative information about global trends of the time series, and to easily detect important changes of patterns over time.

Figure 4 reports an example of data reduction and transformation through aSAX algorithm for a portion of the time series related to the variable encoded with the ID n° 16 according to Table 2 (i.e., *cooling coil input water temperature* (CHWC_EWT)). The figure shows the time series after the application of the Hampel filter (green curve) and the time series in form of constant approximated piecewise (black lines). Furthermore, Figure 4 also shows the result of the aSAX transformation of the time series into a symbolic string. The variable can assume three discrete values encoded with the symbols 16.A, 16.B or 16.C according to the region the piecewise segments of 15-min fall in.

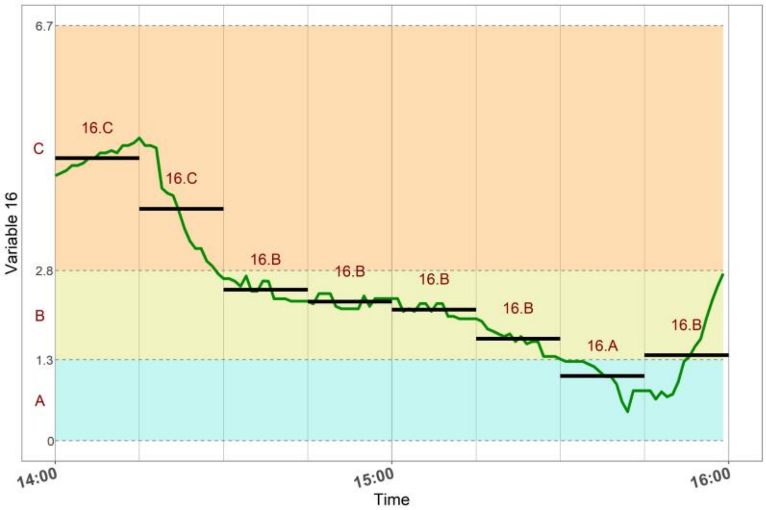

Figure 4. Example of aSAX transformation for a numerical variable.

The obtained symbol sequence is 16.C-16.C-16.B-16.B-16.B-16.B-16.A-16.B from which it is possible to infer that the original time series is characterized by changes in the pattern at times 14:30, 15:30, 15:45, that in this work are intended as events.

As a result, time series are transformed in discrete-time discrete-value sequences of equidistant symbols making it possible to extract events from them.

5.2 Regime identification

At this stage, a regime identification is performed on a daily scale, to detect when transient and non-transient conditions typically occur during the AHU operation.

To that purpose, an automatic regime detector is used to identify the transient period and separate it from the non-transient one. The details of the detector used are the same as that reported in [16][63]. The transient identification is performed on data with sampling time of 1 minute, specifically analysing the *cooling coil valve position* (CHWC_VLV), the *supply air temperature* (SA_TEMP), *supply fan speed* (SF_SPD) and the supply air static pressure. Then, the frequency of transient data points during the day is evaluated for each 15-min aggregation interval, derived from the data reduction phase (Figure 5). Thanks to that analysis, it is possible to establish during which aggregation interval, out of the reduced (15-min-long) daily time series, a transient condition has the highest frequency of occurrence.

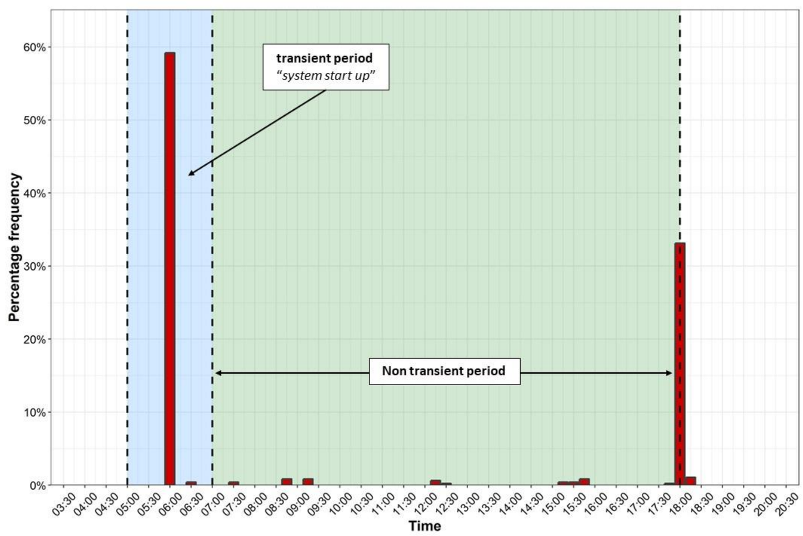

Figure 5. Identification of the transient period.

Starting from such aggregation interval of 15 min, the transient period is evaluated considering a time window of two hours (i.e., blue area of the plot) that includes one hour before and later the aggregation interval considered (Figure 5).

As can be noticed from Figure 5, transients occur at the start-up and the shut-down of the AHU. Among the two transient periods, only the start-up transient is investigated in this paper because during that period the system dynamics affects the successive operation, while in the other case the system is thereafter turned off.

As a result, the non-transient period is supposed to start at the end of the start-up time interval and to end when the system is turned off.

Therefore, excluding the night hours, during which the AHU is certainly not operated, the dataset is segmented as follows:

- From 05:00 to 07:00: transient period labelled as "*system start-up*";
- From 07:00 to 18:00: non-transient period.

In the following sections a tailored FDD methodology for each operation regime of the systems under analysis (i.e., transient, non-transient) is presented.

5.3 FDD methodology for the transient period

The main flow of FDD research reported in literature has been carried out in a steady-state approach [14,17,19,64,65], because the operating characteristics during this operation is relatively more credible and reproducible than in a transient state [64].

Transient data are characterised by great variation in the time domain and require specific data analytics algorithms to be employed to properly reflect the system dynamics. The herein proposed methodology provides, as a main added value to what was already present in literature papers, a tailored approach for such condition of operation.

An overall procedure is developed to obtain temporal association rules that are representative of frequent relationships between events in multiple time series, using a time window and a time lag. As discussed in Section 4.2 temporal association rules are an interesting extension of association rules that include a temporal constraint, which leads to different forms of IF-THEN implication over time. When an event leads to the occurrence of another event, there may be causal relationship or certain correlation between them. The corresponding mining purpose is to find out the reference fault-free association rules between events and time in a temporal transaction dataset, whose violation can suggest the presence of faulty conditions during the start-up period of the AHU system. The extraction makes it possible to find those sequences of events that appear many times among monitored fault-free days and have a high rate of occurrence (i.e., reference rules).

The reference association rules have been searched in the 20 days tagged as fault-free (training dataset) while the remaining 2 fault-free days and 11 faulty days (testing dataset) were used in the successive fault detection phase.

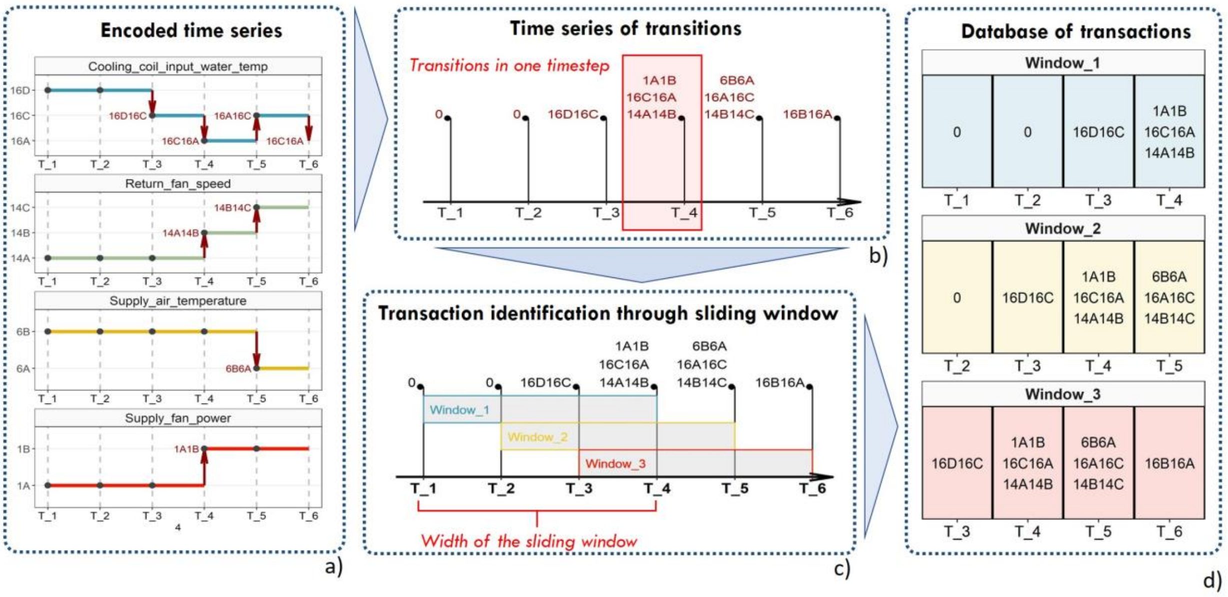

Figure 6. Procedure for the construction of the database of transactions.

Before extracting reference temporal association rules from data, it is necessary to create the database of transactions T following the framework shown in Figure 6.

The first step consists of putting together all the transitions that occur in each time series into a unique multivariate time series of transitions.

In particular, according to the symbolic transformation performed during the pre-processing stage a transition in a time series is a kind of event that corresponds to the change of symbol (i.e., encoded discrete values of the variable) in a specific timestep across two consecutive aggregation intervals.

As an example, Figure 6 (a) shows six timesteps of four time series (i.e., *cooling coil input water temperature* (CHWC_EWT), *return fan speed* (RF_SPD), *supply air temperature* (SA_TEMP), *supply fan power* (SF_WAT)). The time series *supply fan power* (SF_WAT) corresponds to the operation variable of the AHU encoded with the ID n° 1 and assumes only two discrete values (encoded with the symbols 1A and 1B) along the six timesteps considered. In the same way the time series *return fan speed* that corresponds to the operation variable of the AHU encoded with the ID n ° 14, assumes three discrete values (encoded with the symbols 14A, 14B and 14C) among the six timesteps. If two consecutives aggregation intervals are encoded with the same symbol, no transition (i.e., event) is detected. Otherwise, during a specific timestep, a transition (i.e., event) is encoded reporting the ID n° of the variable and the two symbols included in the change of discrete value. For example, according to Figure 6 (a), at the first timestep T_1 for any time series, a transition does not occur and then 0 is stored in the time series of transitions (Figure 5 (b)). On the contrary, at the fourth timestep T_4, a transition occurs for the time series 1, 14 and 16. In particular, for time series 1 and 14, occurs a change from symbol “A” to symbol “B” (events encoded as “1A1B” and “14A14B” respectively) while for time series 16 the variable changes symbol from “C” to “A” (event encoded as “16C16A”).

Once the encoded events are stored in the multivariate time series of transitions (Figure 6b), the database of transactions is constructed by chunking this time series considering a fixed-length sliding time window (Figure 6 (c)). Figure 6 (d) shows how the encoded transitions for each timestep are stored in the database of transactions. For example, assuming a sliding window that includes four timesteps, the database T can be represented by a $4\ x\ n$ transition matrix where n corresponds to the maximum number of sliding windows which can be contained in the time series of transitions.

Considering that the time windows are sliding a timestep by time, two consecutive rows in the database T (Figure 6 (d)) differ only for a single item. As a reference considering a time series of transitions with 6 timesteps and a sliding window that includes 4 timesteps, the database of transactions is a $4\ x\ 3$ transition matrix given that the maximum number of complete time windows is equal to 3 (Figure 6 (d)). After the construction of the database T, the temporal association rules are searched among transactions.

The cSpade algorithm [51] has been selected for the extraction of the rules from the inter-transactional database, setting in advance three fundamental parameters: minimum confidence, minimum support, and maximum time lag between antecedent and consequent item sets (equal to the sliding window length).

According to the proposed methodology, the first two parameters (i.e., confidence and support) should be as high as possible~~,~~ to ensure that the extracted rules are much frequent as possible and then representative of the normal behaviour of the system.

Once the reference rule set has been identified, it is used for detecting the presence of potential faults in a testing dataset.

In particular, a temporal association rule is expressed as a logical IF-THEN implication where the presence of an event (i.e., antecedent) implies the occurrence of another event (i.e., consequent) within a certain time lag. According to this formulation, three potential violations can occur when such rules are applied on a testing set of data:

i) absence of the antecedent itemset,

ii) absence of consequent itemset,

iii) absence of antecedent and consequent item sets.

In that perspective, the violation analysis helps physical interpretation of rules making it possible to assess their sensitivity to the presence of specific faults or group of them.

In section 6.2, the results of the transient methodology herein described are presented and discussed providing further details about the setting of the input parameters and the post-processing of the extracted association rules.

5.4 FDD methodology for the non-transient period

The methodology employed for performing the FDD analysis during non-transient period relies on three fundamental phases that can be generalized as follows:

- Development of reference models through classification trees, representative of the normal behaviour (fault-free condition) of the system under analysis;
- Comparison between the estimated behaviour of the system and the actual one (i.e., evaluation of model residuals) for detecting potential faulty conditions;
- Analysis of the model residuals for diagnosing the most probable cause associated to a specific fault (fault diagnosis).

The first step of the process consists of a robust characterization of the fault free operation of the AHU during the non-transient period (i.e., from 07:00 to 18:00). To this purpose, several estimation models (i.e., classification trees) have been developed on a portion of the available non-transient dataset. In detail 20 days tagged as fault-free were considered at this stage (training dataset) while the remaining 2 fault-free days and 11 faulty days (testing dataset) were used in the successive diagnostic phase.

For the development of the estimation models (i.e., classification trees), all the variables related to the operation of the AHU (e.g., *supply fan power* (SF_WAT), *return fan power* (RF_WAT), *supply air flow rate* (SA_CFM)) have been selected once at a time as target attribute while the remaining ones have been used as input attributes. However, features related to external forcing variables to the AHU system (i.e., *cooling coil input water temperature* (CHWC_EWT), *outdoor air temperature* (OA_TEMP)) have been used only as input attributes.

In that way, 21 classification trees are developed for providing a robust benchmark of the fault-free operation. To that purpose, a CART algorithm is employed as a supervised classifier in the study. The developed classification trees estimate for each target variable and for each 15-min aggregation interval included in the non-transient period the most probable discrete value (encoded as symbol) according to the relationship that exists between all the input variables and the dependent attribute. Successively all the classification trees developed are put together in the same estimation layer as shown in Figure 7. At this stage, the estimation process can be summarized as follows:

- At each aggregation interval (i.e., 15 min.) the monitored variables are encoded into symbols through the aSAX method (i.e., pre-processing stage);
- The set of encoded variables goes through the estimation layer (that consists of 21 classification trees) providing an estimation of each target variable for the considered aggregation interval;
- The actual symbols are compared with the estimated ones.

The latter step consists in the evaluation of the model residuals.

In this study, the difference between two equal symbols is assumed to be zero, while the residual differs from zero if the symbols are at least one alphabet apart. For example, if the estimated and actual symbol for a variable is equal to “A” and “B” respectively, the residual between those symbolic discrete-values is equal to 1 (Figure 7).

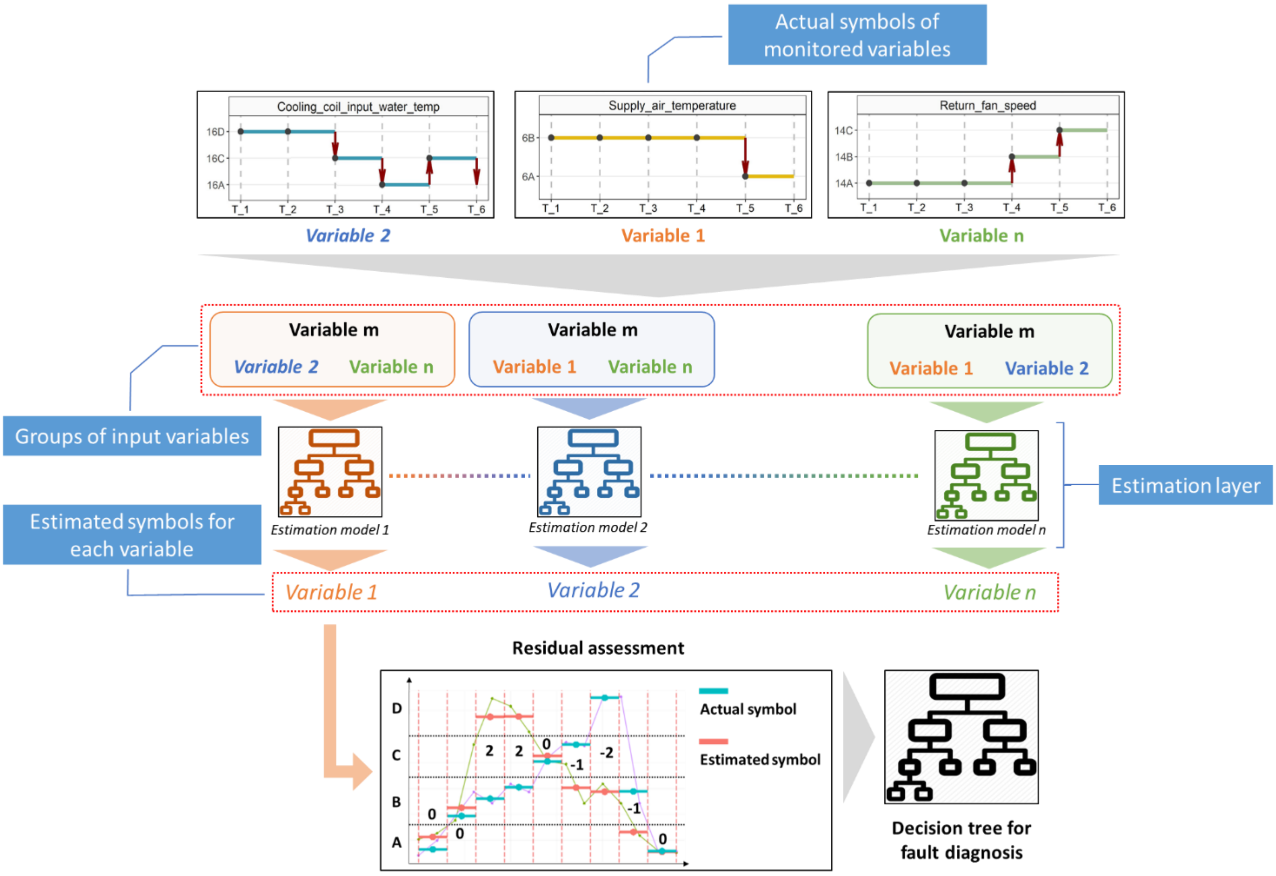

Figure 7. Analytics module for the non-transient period.

Considering that the estimation models are trained on fault-free data, at the end of the estimation process it is possible to assess how much the input data differ from the reference fault-free behaviour of the AHU through the analysis of residuals. Understanding which variables are out of range and assessing the severity of those deviations enables the detection of possible faulty conditions. In order to test this FDD procedure, all the days excluded from the training set of the reference models (i.e., 2 fault-free days and 11 faulty days) have been considered. In particular, each day included in the testing dataset is labelled as "Normal" or with the tag of one of the faults reported in Table 1.

The time series of the 13 days are pre-processed (aggregated in intervals of 15-min and encoded in symbols) and put through the estimation layer (i.e., 21 classification trees) generating a dataset of residuals as shown in Figure 8. At this stage, a further classification tree has been developed to predict the label of each faulty or normal condition (Figure 8) for performing the fault diagnosis. This classification tree estimates the most probable label (e.g., CCVSFC, EASFC, RFCF or Normal) according to the residuals evaluated for each variable as an outcome of the estimation layer.

Aggregation interval	*Day*	*Variable 1 Residual*	*Variable n Residual*	*Variable 21 Residual*	*Fault label*
15:00 – 15:15	1	0	…	0	Normal
15:15 – 15:30	1	0	…	0	Normal
15:30 – 15:45	1	0	…	0	Normal
…	…	…	…	…	…
15:00 – 15:15	5	1	…	3	CCVSFC
15:15 – 15:30	5	0	…	-1	CCVSFC
15:30 – 15:45	5	-2	…	0	CCVSFC
…	…	…	…	…	…
15:00 – 15:15	10	2	…	1	EASFC
15:15 – 15:30	10	1	…	0	EASFC
15:30 – 15:45	10	0	…	-2	EASFC
…	…	…	…	…	…
15:00 – 15:15	13	-3	…	1	RFCF
15:15 – 15:30	13	1	…	-2	RFCF

Set of input variables of the classification tree

Target variable of the classification tree

Figure 8. Structure of the database used for developing the classification tree of fault diagnosis

In the dataset reported in *Figure 8* the target variable is the fault tag, and the same tag is assigned to all of the 44 aggregation intervals of 15-min that belong to the same day (included in the 11 hours of "non transient" operation of the AHU from 7:00 to 18:00), generating a total amount of 572 instances on which develop the classifier. The present methodology exploits the CART algorithm for developing the decision trees since it proved to be a good choice for fault diagnosis [19][66].

As already mentioned above, in this paper the developed FDD tool is trained and tested on real data of an AHU operated in non-transient cooling mode for 33 non-consecutive days during the summer season (22 "normal" days and 11 "faulty" days). Note that the decision and association rules extracted through the proposed supervised and unsupervised approaches can be considered valid only for the operation mode under consideration. In this perspective, rule-based tools can be easily integrated in FDD process with hierarchical architecture capable to exploit only the useful knowledge during specific conditions. For instance, the use of automatic detector makes it possible to call specific sets of rules depending on the operating mode of the AHU: off mode, heating mode, free cooling mode, and mechanical cooling mode [67].

In Section 6.3, the results of the non-transient methodology herein described are presented and discussed providing further details about the performance, reliability and generalizability of the entire process.

6 Results

6.1 Pre-processing stage

According to the methodological framework introduced in Section 5, a data preparation stage was preliminarily implemented. Firstly, outliers were filtered out by implementing the Hampel filter on the 1-minute time series. For each data sample of the time series, the filter computes the standard deviation and the median of a window composed of the current sample and $\frac{Len-1}{2}$ adjacent samples on each side of the current sample. *Len* is the window length and in this study is set equal to 31 minutes. A window with a length of 31 minutes could be not too much sensitive to the presence of outliers considering that the sample on which the standard deviation is computed is quite large. However, such window length proved to be suitable for identifying extremal values that certainly are related to problems of the sensing system. In the performed analysis, the filter does not take into account the first and the last *(Len-1)/2* data points of each daily time series. Such data points are always related to measurements during the hours when the AHU system is turned off (time intervals from 00:00 to 00:14 and from 23:45 to 23:59) and do not affect the results in any ways.

After data pre-processing (i.e., cleaning and replacement of outliers) a data reduction was performed by means of a PAA process with the aim of approximating the time series of each considered variable to the mean value calculated in non-overlapped time intervals with a fixed length of 15 min. The time interval length of 15 minutes was chosen as the best trade-off between approximation accuracy and data size reduction. Successively, the encoding of the reduced variables in symbols was carried out by implementing the aSAX algorithm [45].

The algorithm was initialised for each variable by identifying the number of symbols (i.e., discretization intervals) and the initial positions of the breakpoints (i.e., borders of the discretization intervals) with a hierarchical cluster analysis using the Ward linkage method [47]. Through the clustering algorithm, it was possible to obtain the optimal number of discretization intervals (i.e., number of symbols) by computing several cluster validation metrics. This process was completely automated and performed through Nbclust package [68] available in the statistical software R. The number of discretization intervals was constrained from 2 to 4 considering only data referred to the period of operation of the system (i.e., ON-hours of the system).

When the optimal positions of the adaptive breakpoints were found and each variable was encoded in symbols, the operation conditions of the AHU were considered fully characterised. Then, the data related to OFF-hours of the system operation were analysed to find possible additional intervals. In particular, if during OFF-hours a variable typically assumes values that are out of the identified ranges of discretization, a new lower or upper half-open interval was appended to the previous ones.

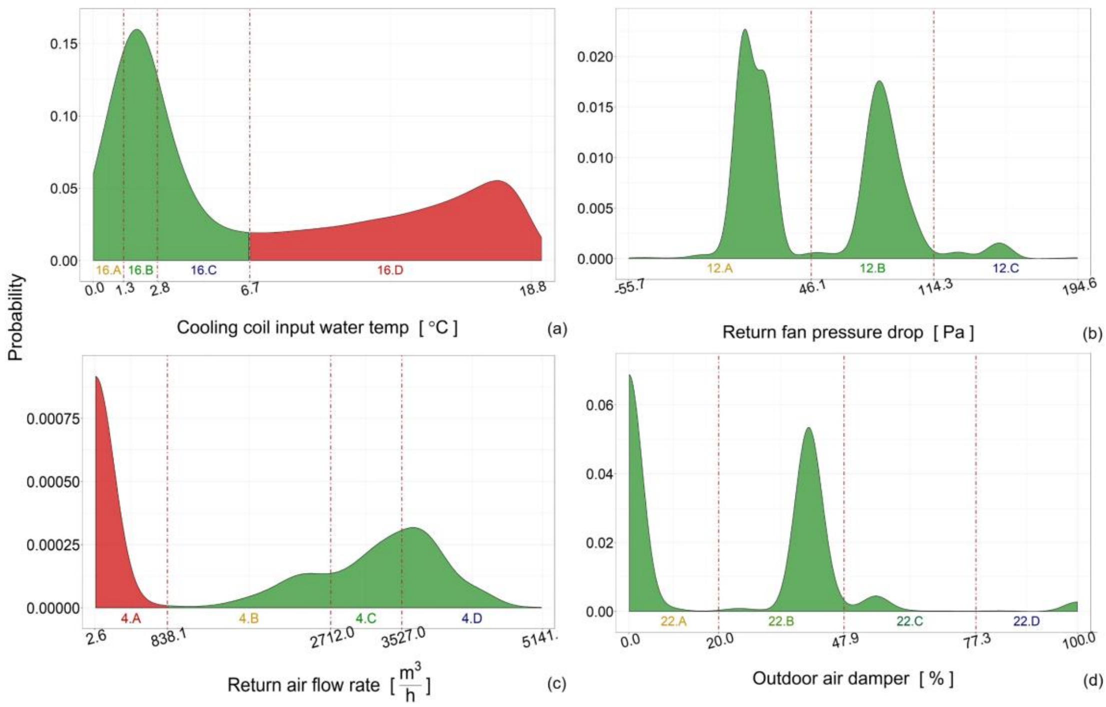

Figure 9. Distributions and breakpoint identification for some variables.

Figure 9 shows the encoding process performed for 4 variables (i.e., *cooling coil input water temperature* (CHWC_EWT), *return fan pressure drop* (RF_DP), *return air flow rate* (RA_CFM), *outdoor air damper position* (OA_DMPR)) randomly selected from the set of inputs. It can be observed that for two variables an additional OFF-hours discretization interval (i.e., red area of the distributions in Figure 9 (a) and (c)) was added to the other ranges of values for the symbol encoding (i.e., ID n° = 16, symbol = D and ID n° = 4, symbol = A).

As a reference, Table A in Appendix A summarizes the transformation results obtained, with the specification of the numerical range corresponding to each symbol for all the analysed operational variables.

At this stage, according to the procedure described in Section 5.2, transient and non-transient periods were identified and the data set was consequently segmented. In particular, the time interval between 5:00 and 7:00 was labelled as transient start-up period, while the period from 7:00 to 18:00 was considered as non-transient period. The results obtained from the application of the methodological framework are in the following presented and discussed separately for transient and non-transient periods.

6.2 Fault detection analysis for the transient period (system start-up)

According to the methodological process introduced in section 5, the encoded time series were analysed for extracting temporal association rules in the start-up period of system operation. In detail, the transitions of the variables (i.e., change from a symbolic discrete-value to another one) were preliminarily encoded and the inter-transactional database was created considering a sliding window of 60 minutes. The width of the sliding window was chosen to be large enough to include any effect of the system dynamics, but tight enough to ensure that the occurrence of a consequent itemset was related to a physics-based implication with its antecedent itemset.

Considering that the fault detection methodology was conceived for extracting reference association rules of normal operation, the inter-transactional database was created from the fault-free dataset, by selecting rules with high values of support and confidence.

Typically, the main issue related to association rules mining consists in handling and filtering the large number of rules extracted and eventually identify those that are of interest [20]. To tackle this problem and facilitate the mining of useful knowledge from extracted rules, a post-mining phase was performed.

The post-mining phase was aimed at solving various practical issues, such as interestingness, redundancy, generalization, visualization and interpretability of association rules.

To this purpose, additional quality metrics were introduced: the daily support of the rule (i.e. SUPP.DAY) calculated for both fault-free ($\mathrm{SUPP.DAY_{NORMAL}}$) and the faulty days ($\mathrm{SUPP.DAY_{FAULTY}}$) and the actual time lag between the antecedent and consequent of a rule (ACTUAL TIME LAG). In more detail, the $\mathrm{SUPP.DAY_{NORMAL}}$ is defined as the percentage of fault-free days during which a single association rule (R_i) occurred, while $\mathrm{SUPP.DAY_{FAULTY}}$ is calculated for the faulty days (Eq. (5) and Eq. (6), respectively).

$$SUPP.DAY_{NORMAL}, R_i = \frac{\text{N° of Free−fault days with of the occurrence of the rule } \mathrm{R}_i}{\text{Tot. N° of Free−fault days}} \quad (5)$$

$$SUPP.DAY_{FAULTY}, R_i = \frac{\text{N° of Faulty days with of the occurrence of the rule } \mathrm{R}_i}{\text{Tot. N° of Faulty days}} \quad (6)$$

However, according to ASHRAE project RP-1312, during the faulty day tagged as CCVSFO (i.e., cooling coil valve stuck open), the blockage of the cooling valve in fully open position was implemented from 8:00 to 18:00 and hence out of the start-up period of the system. For this reason, the day tagged as CCVSFO has been not considered in the calculation of $\mathrm{SUPP.DAY_{FAULTY}}$.

The ACTUAL TIME LAG was introduced to evaluate the most frequent temporal distance between the first occurrence of an antecedent and the last occurrence of the corresponding consequent of a specific rule. Consequently, even though the rules are searched with a sliding window of 60 minutes, the user can have a feedback about the most frequent time interval within a consequent occurs given the presence of its antecedent.

The ACTUAL TIME LAG was calculated for each rule by computing the cumulative frequency of occurrences of the temporal distance between antecedent and consequent. For each rule a cumulated frequency threshold of 80% was considered in order to evaluate this metric.

Figure 10 shows the frequency distribution of the ACTUAL TIME LAG for two rules. The rule on the left (i.e., rule 1077) occurs for more than the 80% of the time with an actual time lag between the antecedent itemset and consequent itemset of 15 minutes, while for the rule on the right (i.e., rule 15268) the 80% of occurrences has a characteristic time lag lower or equal to 30 minutes.

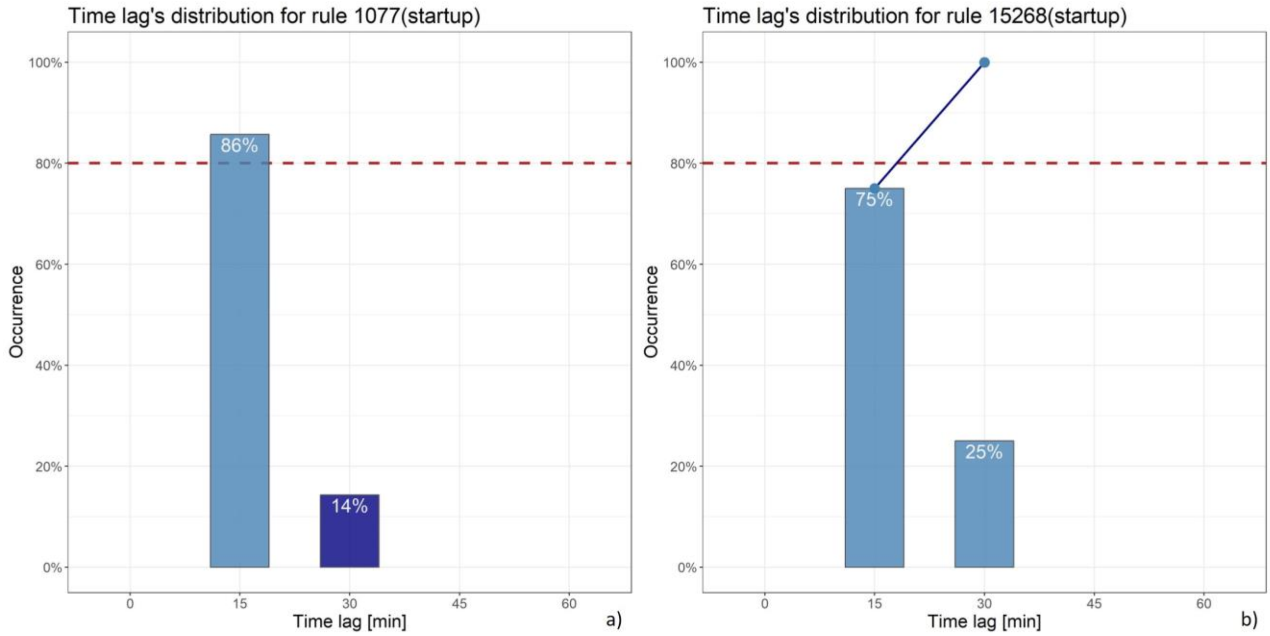

Figure 10. Distribution of the time lags for rule 1077 (a) and rule 15268 (b) – (refer to Table B in Appendix A for the description of the rules).

At this stage, more than 15,000 rules were extracted from the start-up dataset of fault-free days (in more or less 10 min.), assuming minimum support and minimum confidence equal to 0.7 and not including drivers of system's operation as potential consequent events (i.e. *outdoor air temperature* (OA_TEMP) and *cooling coil input water temperature* (CHWC_EWT)).

After the rule extraction, the values of support and confidence were recalculated considering only the occurrences of each rule within the evaluated ACTUAL TIME LAG (instead of the window of 60-min), reducing the set of rules to 7,419 rules.

Since the rules extracted should be representative of the fault-free operation of the system, only the rules, which in the testing dataset frequently occur in normal days and rarely in the faulty ones, are of interest for the problem under investigation. To this purpose, after the application of the 7,419 temporal association rules to the testing dataset, only the rules with a $SUPP.DAY_{NORMAL}$ equal to 1 (i.e., the rule occurring for each day labelled as “normal” included in the testing dataset) and a maximum value of $SUPP.DAY_{FAULTY}$ equal to 0.3 were considered with the final result of obtaining 465 reference rules (SUPP.DAY values are set by the user).

As a general approach, the parameters were set in order to obtain a limited number of interesting rules, which respect the following conditions i) each rule occurs during fault-free condition with high support and confidence, ii) each rule has high probability to be violated during faulty conditions regardless from the fault type.

In this perspective, general rules that are sensitive to more fault types at the same time were preferred to those violated only for specific faults.

The introduced metrics allow an enhanced comprehension of the rule set, making it possible to discriminate rules with high support and confidence occurring during both fault-free and faulty days, from the rules, robust as well, occurring only during the normal operation of the system.

Figure 11 shows for each day in the testing dataset (composed by 11 different faulty days and 2 Normal days) the percentage of rules (out of the 465 considered) which occurred and/or have been violated, with specification of the kind of violation detected. In particular, the label “*presence*” indicates that the rule occurred with its antecedent and consequent while the labels “*antecedent*”, “*consequent*” and “*absence*” indicate three different types of violation. In detail, the label “*antecedent*” denotes that a rule was violated because of the only presence of the antecedent; the label “*consequent*” indicates that a rule was violated because of the only presence of the consequent; the label “*absence*” indicates the complete violation of a rule because of the absence of both antecedent and consequent. The characterisation of the rules in terms of type of violation helps the interpretation of the path which determines a specific fault. In fact, the presence of the only antecedent, the only consequent, rather than the absence of both item sets, correspond to different behaviours of the system in relation to the presence of the considered faults.

The results obtained can be described according to the severity of rule violation for each day representative of a specific fault implementation or normal operation. To this purpose four different groups of days were identified and in the following described.

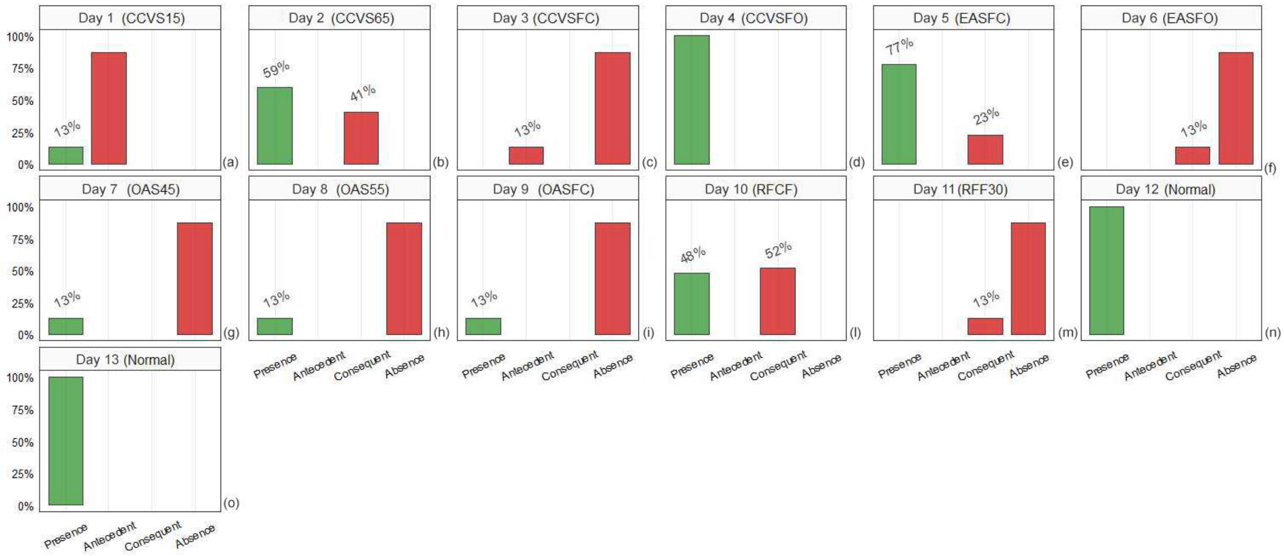

Figure 11. Characterization of the presence or the violation of the extracted rules for the testing days (refer to Table 1 for the encoding of faults).

The first group includes days characterized by the presence of the 100% of the 465 rules tested. This is the case of days in Figure 11 (d), (n) and (o) tagged as Normal and the faulty day tagged as CCVSFO. Such condition suggests, as expected, that during the faulty day CCVSFO the start-up of the system can be considered normal.

The second group instead, includes the days in Figure 11 (a), (c), (f), (g), (h), (i) and (m) that are characterized by a net prevalence of rule violations (more than 70%). Moreover, for those days, the presence of a fault is also associated to a specific kind of violation of the rules. As a reference, in case of CCVSFC, EASFO, OAS45, OAS55, OASFC and RFF30 the rules are violated mainly due to the absence of both antecedents and consequents, while only in the in case of CCVS15 the rule was violated for the absence of consequent.

The third group includes the day in Figure 11 (e) for which, during the start-up period, the percentage of violations is lower than the percentage of valid occurrences of the rules. Such condition suggests that during this day the behaviour of the system is similar to the normal one limiting the number of violations occurred. The main reason is that such fault does not strongly affect the system operation making the detection process less sensible to its presence. This result agreed with the findings of the ASHRAE-RP 1312 project, during which the analysed dataset was generated [14].

The last group includes days in Figure 11 (b) and (l) that are characterized by a similar amount of violated and not violated rules (violation rate between 40% and 60%). These two faults seem to affect the performance of the system differently from other faults respect to which hypothetically should exhibit high similarity (i.e., CCVS15 and RFF30). Regarding the fault CCVS65 (Figure 11 (b)), the cooling coil valve is stuck open at 65% and therefore the supply air flow is overcooled. In this case, the system reacts by opening the heating coil valve and operating in fully recirculation mode for increasing the *supply air temperature* (SA_TEMP). Consequently, the failure of the cooling coil valve does not affect the capability of the system in reaching the supply set-point temperature, but the operation of the other components is different from the normal condition.

On the opposite, during the day (Figure 11 (a)) tagged as CCVS15 (included in group 2) the cooling coil valve is almost closed limiting the heat transfer with the supply air flow that does not reach the set point temperature. Such case is representative of the complete failure of the system in maintaining the desired conditions of the indoor environment, as a matter of fact, justifying a higher rule violation rate for CCVS15 respect to CCVS65.

Regarding the fault RFCF (Figure 11 (l)), the system is operated implementing the complete failure of the return fan despite its speed control signal is correctly elaborated. Instead, during the day in Figure 11 (m) tagged as RFF30 (included in group 2), the return fan is not corrupted, but it is subjected to a faulty control signal. In this case the high number of rules violated for RFF30 suggests a higher sensitivity of the extracted rules to frequent transitions of the fan speed discrete values rather than fan power ones.

Some key figures related to the 465 extracted rules are described below. The rules are characterized by an ACTUAL TIME LAG that lies between 15 and 30 minutes. The evaluation of the ACTUAL TIME LAG can be considered as an essential step for reducing the intrinsic latency of the FDD process during real implementation. Indeed, the ACTUAL TIME LAG gives the opportunity to check the occurrence of a rule within a time interval smaller than the width of the sliding window used for the rule extraction (in this case study, equal to 60 min.).

The transitions in the antecedent and consequent item sets are reported in Table 3 with the corresponding occurrence frequency. In particular, the number of different consequent item sets is 13, resulting from a combination of 4 different events, while the antecedent item sets are 99, resulting from the combination of 12 different events.

Table 3. Occurrence frequency of each event included in the antecedent and consequent item sets

Itemset	**Variable**	**Event**	**Frequency**
Antecedent	Return Fan Speed	RF_SPD [A-B]	87%
	Cooling coil input water temperature	CHWC_EWT [D-C]	31%
	Return fan power	RF_WAT [A-B]	24%
	Exhaust air damper position	EA_DMPR [A-B]	23%
	Cooling coil output water temperature	CHWC_LWT [C-B]	22%
	Supply fan speed	SF_SPD [A-B]	22%
	Cooling coil input water temperature	CHWC_EWT [C-A]	21%
	Supply fan power	SF_WAT [A-B]	21%
	Return fan start/stop signal	RF_SST [A-B]	18%
	Return air flow rate	RA_CFM [A-B]	16%
	Return fan pressure drop	RF_DP [A-B]	12%
	Return fan pressure drop	SF_DP [A-B]	9%
Consequent	Return Fan Speed	RF_SPD [B-C]	87%
	Supply Air Temperature	SA_TEMP [B-A]	49%
	Cooling coil output water temperature	CHWC_LWT [B-A]	46%
	Cooling coil air temperature	CHWC_DAT [B-A]	36%

The obtained rule set, including the most representative rules, is reported in Table B in Appendix A. The rules extracted are meaningful since they can be interpreted as chains of events that characterise the normal operation of the AHU in reaching the set-point conditions during the start-up period. Indeed, extracted rules can be expressed as IF-THEN implications to be verified within a specific time interval. As a reference, the rule n° 8661 (included in Table B in Appendix A) can be written and interpreted as follow: IF (RF_SPD [A-B] and CHWC_LWT [C-B] and EA_DMPR [A-B]) occur THEN (CHWC_DAT [B-A] and RF_SPD [B-C]) will occur within 30 minutes with the 100% of confidence during a normal day.

In detail, the antecedent itemset includes transitions related to the return fan speed (RF_SPD), the *cooling coil output water temperature* (CHWC_LWT) and the *exhaust air damper position* (EA_DMPR) that imply the occurrence of consequent transitions related to *cooling coil output water temperature* (CHWC_LWT) and *return fan speed* (RF_SPD).

In order to further improve the interpretability of the rules a novel visualization was proposed in this work. An example of this visualization is showed in Figure 12, where the profiles of the variables involved in rule n°1253 (see Table B in Appendix A).

Figure 12 shows the trend of the variables in terms of real profile (i.e. green curve) and PAA (i.e. black segments). Regardless of the approximation introduced by the PAA, the behaviour of the variables during the transient period is preserved, as can be seen by looking at the *supply air temperature* trend (SA_TEMP). In fact, during the start-up period the *supply fan speed* (SF_SPD) initially ramps up and then it is reduced to a constant level. The transitions of the antecedent itemset are reported in red, while the consequent itemset in blue. The PAA is represented in a window of 60 minutes, while with a darker shade of grey the length of the ACTUAL TIME LAG (i.e., 15 minutes) is reported. On the y-axis are shown the values used for the discretization of each variable.

The rule in Figure 12 shows a typical behaviour of the system at the start-up period, in terms of the variation of *supply fan speed* (SF_SPD), *exhaust air damper* (EA_DMPR), *return fan power* (RF_WAT), and *supply air temperature* (SA_TEMP).

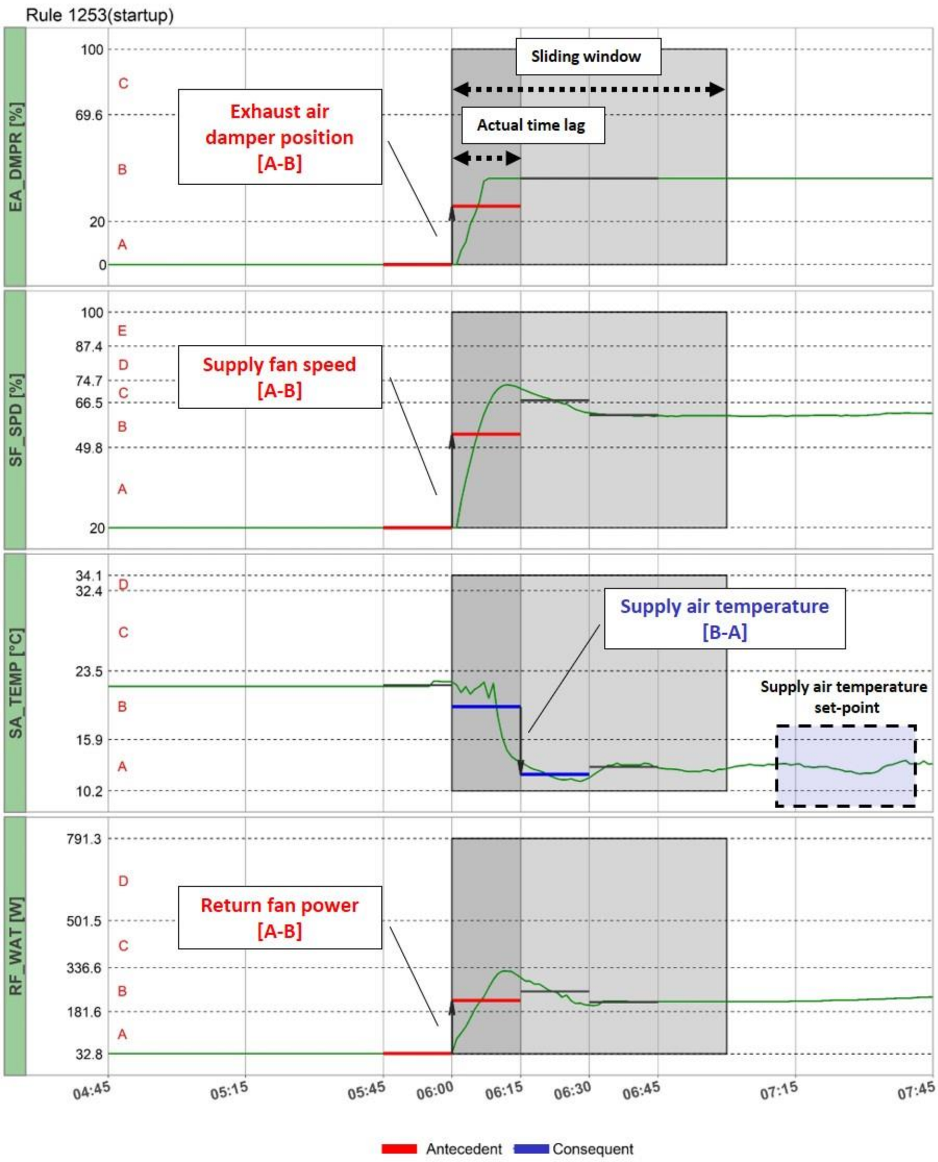

Figure 12. Visualization of an extracted temporal association rule (refer to Table 2 for variable encoding).

According to this rule, usually at the time scheduled for the start-up (i.e. 6:00 a.m.), the supply fan receives the start signal contemporary to the opening of the exhaust air damper while the *return fan power* (RF_WAT) increases (change from A to B). After 15 minutes from the occurrence of the first antecedent transition in the event chain, according to the rule, the *supply air temperature* (SA_TEMP) decreases from symbolic discrete-value B to A until the reaching of the desired set-point.

This proved that the chain of events related to each association rule provides information about the expected behaviour in terms of discrete-value changes among influencing variables of the AHU during normal operation.

6.3 Fault detection and diagnosis during non-transient period

In this section, the results obtained for the application of the methodology during non-transient period described in section 5 are presented. The first step is aimed at developing a CT reference model for each variable to predict the normal operation of the system. For the development of these reference estimation models, all the variables related to the operation of the AHU have been selected once at a time as target attribute while the remaining ones have been used as input attributes. However, features related to external forcing variables to the AHU system (i.e., *cooling coil input water temperature* (CHWC_EWT), *outdoor air temperature* (OA_TEMP)) have been used only as input attributes. As a result, 21 reference models were built for providing a robust benchmark of fault-free operation. Moreover, the variables used as input were also considered with a maximum backward lag of four time steps (i.e. 60 minutes). Indeed, the decision trees are able to predict the discrete values (i.e., symbol) of a target variable considering the discrete values of the input variables both in the same and previous aggregation intervals.

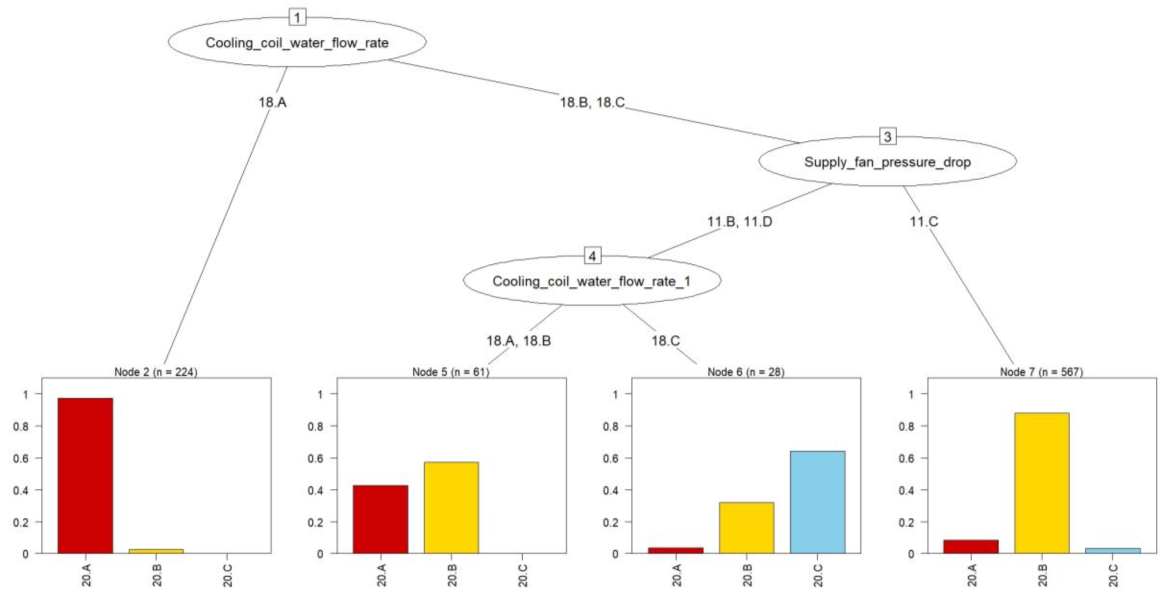

Figure 13. Classification tree for the estimation of the symbolic discrete-values of the cooling coil valve position (CHWC_VLV).

Figure 13 reports as an example the CT model developed for predicting the discrete values (i.e., symbol) of the variable *cooling coil water valve position* (i.e., variable tagged as CHWC_VLV with

ID n° = 20), with an overall accuracy of 88% evaluated as the fraction of correct predictions with respect to the total number of predictions. The algorithm selected as input variables the *cooling coil water flow rate* (i.e., variable tagged as CHWC_GPM with ID n° = 18) and the *pressure drop of the supply fan* (i.e., variable tagged as SF_DP with ID n° = 11). From this CT, it is possible to extract useful decision rules for straightforwardly characterizing all the implications between discrete values (i.e., symbols) that typically occur during the fault-free operation of the AHU. Table 4 reports all the IF-THEN decision rules extracted from the CT shown in Figure 13 with the evidence of the accuracy achieved in each leaf node. The accuracy refers to each single leaf node assuming that the predicted label of the node corresponds to the label of the majority of the objects.

For example, according to rule 4, the value of the response variable *cooling coil valve position* is equal to 20_B (i.e. CHWC_VLV lies in the interval 41 – 75 [%]) if the *cooling coil water flow rate* is equal to 18_B or 18_C (i.e. CHWC_GPM lies in the interval 0,89 – 2.7 [m^3/h]) and the *supply fan pressure drop* is equal to 11_C (i.e. SF_DP lies in the interval 562 – 770 [Pa]).

Table 4. Decision rules for the estimation of the symbolic discrete-value of cooling coil valve position (CHWC_VLV).

Rule number	**Decision rules**	**CHWC_VLV discrete-value**	**N° of objects**	**Leaf node accuracy**
1)	**IF** CHWC_GPM = 18_A	20_A	224	95%
2)	**IF** CHWC_GPM = 18_B or 18_C **AND** SF_DP = 11_B or 11_D **AND** CHWC_GPM (lag -1) = 18_A or 18_B	20_B	61	55%
3)	**IF** CHWC_GPM = 18_B or 18_C **AND** SF_DP = 11_B or 11_D **AND** CHWC_GPM (lag -1) = 18_C	20_C	28	65%
4)	**IF** CHWC_GPM = 18_B or 18_C **AND** SF_DP = 11_C	20_B	567	85%

Once all the estimation models were trained and validated, the residual analysis was performed by using a testing dataset including both faulty and fault-free data (i.e., 2 fault-free and 11 faulty days). Therefore, the difference between the actual status of a variable and that estimated by the CT during a aggregation interval determines the detection or not of a potential faulty condition, since the predicted status should be considered as the reference condition (fault-free). The values of the residuals can be equal to zero in case of absence of deviation from the normal conditions, positive if the actual value is higher than expected, while negative if the actual value is lower than expected. Eventually, in order to perform fault diagnosis, an additional CT model was developed, which uses in input the residuals obtained from the estimation performed through the previously described reference models and as output the tags related to the various faults analysed in this study. Figure 14 shows the classification model obtained, which can classify the faults considered with a set of intuitive rules, reaching an overall accuracy of the 90%.

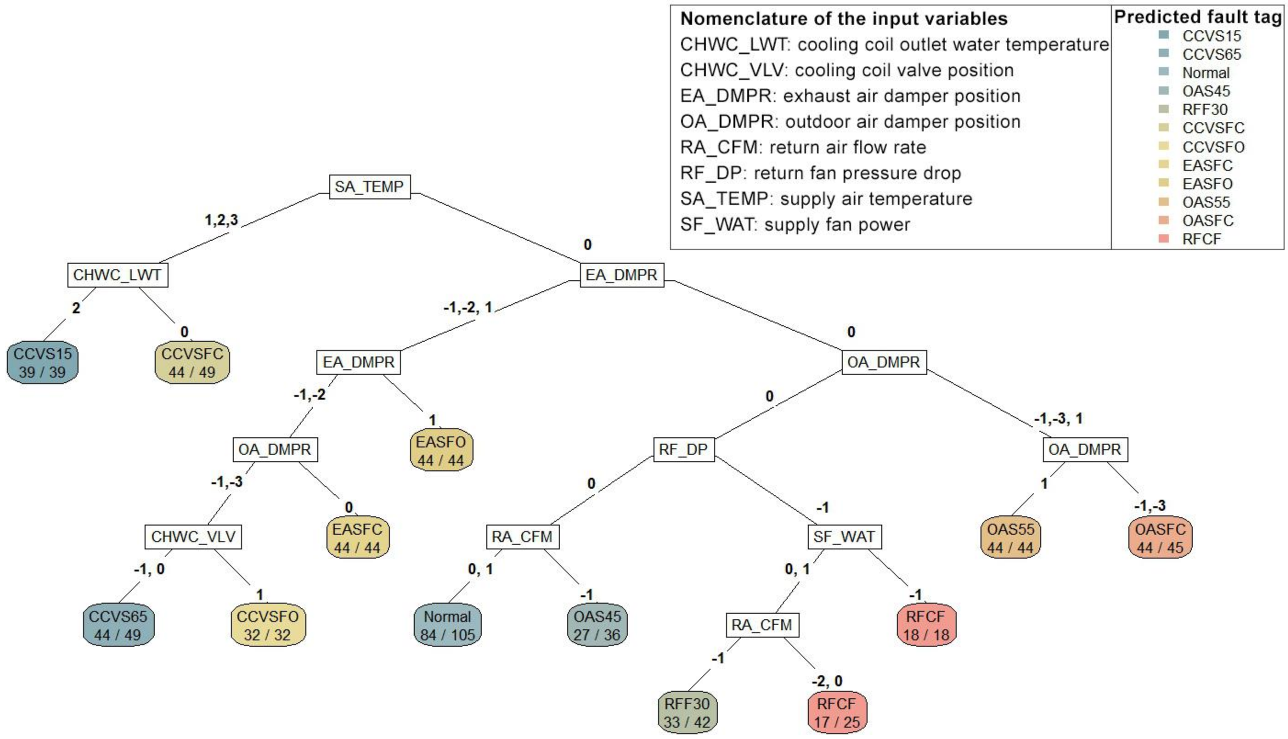

Figure 14. Classification tree for the fault diagnosis during the non-transient period.

The variables involved in input for the classification are the *supply air temperature* (SA_TEMP), the *outdoor air damper position* (OA_DMPR), the *exhaust air damper position* (EA_DMPR), the *cooling coil outlet water temperature* (CHWC_LWT), the *cooling coil valve position* (CHWC_VLV), the *supply fan power* (SF_WAT), the return *fan pressure drop* (RF_DP) and the *return air flow rate* (RA_CFM). The CT developed can diagnose 11 different faults and the normal condition as well. The latter is predicted by following the path of the CT (Figure 14) that includes all zeros (i.e. residual equal to zero) in the splits for the variables SA_TEMP, EA_DMPR, OA_DMPR, RF_DP and RA_CFM. With reference to Figure 14, some other rules are described in the following.

The first split made by the CT algorithm is driven by the *supply air temperature* (SA_TEMP), which identifies the faults due to a blockage of the cooling coil valve at 0% (CCSFC) or at 15% (CCVS15) if the air temperature presents higher values than normal (i.e., SA_TEMP residuals = 1, 2, 3).

In some cases, the faults can be diagnosed by analysing the variables directly related to the corrupted component, such as the blockage of the exhaust and outdoor air dampers at 0%, 55% or 100% (i.e. OASFC, OAS55, EASFC, and EASFO). In other cases, a series of deviation from the normal condition for different variables are considered as symptoms for a specific fault. That is the case, for example, of anomalous energy transfer in the cooling coil due to blockage of the cooling coil valve at 65% (CCVS65) or 100% (CCVSFO). These faults are diagnosed in the case both the air dampers are completely closed (i.e. negative values of residuals), but the *supply air temperature* (SA_TEMP) does not present a deviation from the normal condition. In this case, the system tries to counterbalance the excessive decrease of the temperature of the air by operating in fully recirculation mode.

The effect of a fault related to the return fan (Figure 14) can be easily identified, since the pressure drop at the return fan is reduced, with the absence of deviation, from normal condition, for *supply air temperature* (SA_TEMP) and air dampers.

The discrimination between the *return fan complete failure* (RFCF) and the case when the speed is fixed at 30% (RFF30) can be performed by evaluating the severity of the reduction of the *return air flow rate* (RA_CFM) rather than the reduction of the *supply fan power* (SF_WAT).

The introduced FDD tool is a multiclass classifier and when in operation sorts data into either fault-free (i.e., normal) or faulty classes.

All the evaluation metrics for a multiclass classification model can be understood in the context of a binary classification model (where the classes are “positive” and “negative”). These metrics are derived from the following categories:

- True Positives (TP): Objects labelled as positive and predicted to be positive.
- False Positives (FP): Objects labelled as negative and predicted to be positive.
- True Negatives (TN): Objects labelled as negative and predicted to be negative.

- False Negatives (FN): Objects labelled as positive and predicted to be negative.

The multiclass classification problem can be seen as a set of many binary classification problems and its performance can be assessed labelling as “positive” each class once at time. In the context of the presented multiclass FDD classifiers some metrics have been calculated:

- Accuracy (A): Objects of items correctly identified as either truly positive or truly negative out of the total number of items i.e., $(\mathrm{TP}+\mathrm{TN})/(\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN})$.
- Recall (R): Number of objects correctly identified as positive out of the total actual positives i.e., $\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$. The recall is calculated for each class and then averaged among classes for a global performance assessment of the CT.
- Precision (P): Number of objects correctly identified as positive out of the total items predicted as positive i.e., $\mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$. The precision is calculated for each class and then averaged among classes for a global performance assessment of the CT.
- False Positive Rate (FPR), Type I error: Number of objects wrongly identified as faulty out of the total actual fault-free data i.e., $\mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$. In FDD processes, this error means that data belonging to fault-free class (negative) are incorrectly labelled as faulty (positives) generating false alarms.
- False Negative Rate (FNR), Type II error : Number of objects wrongly predicted as fault-free out of the total actual faulty data i.e., $\mathrm{FN}/(\mathrm{FN}+\mathrm{TP})$. In FDD processes, this error means that data belonging to one of the fault classes (positives) are incorrectly labelled as fault-free (negative) generating missing detection opportunities.

The developed CT exhibits the following performances A = 90%, R = 89%, P = 91%, FNR = 4%, FPR = 4%. The performance of the CT can be also assessed with the detail of each class considered. To this purpose, in Table 5 is reported the Confusion Matrix (CM) of the CT. The CM, in form of table (actual class vs predicted class), allows an effective analysis of the performance of the CT algorithm making it possible to identify confusion between all the considered classes (i.e., mislabelling of objects belonging to a class and classified into another one).

In particular, rows of the table correspond to the actual classes while columns to the predicted ones. At this stage it is possible to evaluate in each class the proportion of prediction actually correct (i.e., Precision) and the proportion of actual values predicted correctly (i.e., Recall).

Table 5. Precision and recall for classification tree of fault diagnosis during non-transient period.

	CCVS15	CCVS65	Normal	OAS45	RFF30	CCVSFC	CCVSFO	EASFC	EASFO	OAS55	OASFC	RFCF	**Total**	**Recall**
CCVS15	**39**	0	0	0	0	5	0	0	0	0	0	0	44	89%
CCVS65	0	**44**	0	0	0	0	0	0	0	0	0	0	44	100%
Normal	0	0	**84**	3	0	0	0	0	0	0	1	0	88	96%
OAS45	0	0	17	**27**	0	0	0	0	0	0	0	0	44	61%
RFF30	0	0	2	1	**33**	0	0	0	0	0	0	8	44	75%
CCVSFC	0	0	0	0	0	**44**	0	0	0	0	0	0	44	100%
CCVSFO	0	5	2	5	0	0	**32**	0	0	0	0	0	44	73%
EASFC	0	0	0	0	0	0	0	**44**	0	0	0	0	44	100%
EASFO	0	0	0	0	0	0	0	0	**44**	0	0	0	44	100%
OAS55	0	0	0	0	0	0	0	0	0	**44**	0	0	44	100%
OASFC	0	0	0	0	0	0	0	0	0	0	**44**	0	44	100%
RFCF	0	0	0	0	9	0	0	0	0	0	0	**35**	44	80%
Total	39	49	105	36	42	49	32	44	44	44	45	43	572	**Average 89%**
Precision	100%	90%	80%	75%	79%	90%	100%	100%	100%	100%	98%	81%	**Average 91%**	

Thanks to the methodology introduced in the present paper the faults in the dataset were diagnosed with both high precision and recall, as can be seen in Table 5. The lowest values of precision and recall are related to the fault *outdoor air damper stuck at 45%* (OAS45), for which part of the records have been mislabelled as “Normal” (i.e., 17 out of 44 objects, that correspond to the 39% of data labelled as OAS45 and to the 89% of the total amount of False Negatives). This condition is due to the fact that the outdoor air damper stuck open at 45% does not invalidate the operation of the system which is similar to the fault-free one during the non-transient period. It is worth nothing that all the assumptions taken, and results obtained are related to a specific operative condition of the system (i.e., cooling mode). The set of rules extracted can be then considered a valid FDD solution if only applied on data consistent with the initial hypotheses. Despite this, even though the analysis is related to a portion of the possible operative conditions of an AHU, the performance achieved suggests good perspectives in applicability and generalizability of the proposed methodology.

7 Discussion and concluding remarks

The paper introduces a data-driven based methodology to perform an AFDD in AHUs. Two different analytics modules were proposed for transient and non-transient conditions of the AHU operation and consequently to enhance the energy performance of the ventilation and air-conditioning process. The dataset used for testing the methodology includes several faulty and fault-free running conditions related to the cooling operative mode of AHUs. Data were gathered from monitoring campaign on two identical AHUs in the framework of the Research Project ASHRAE RP-1312.

The fault detection during the start-up period was performed with an innovative approach by searching frequent and non-anomalous relationships between events in a temporal transaction set using temporal association rules. A temporal association rule is expressed as a logical IF-THEN implication where the presence of an event (i.e., antecedent) implies the occurrence of another event (i.e., consequent) within a certain time lag. According to this approach, in the analysed case study the violation of a rule or group of rules may suggest the occurrence of abnormal conditions during system operation. Three potential rule violations have been considered for detecting faults during the start-up period: i) absence of the antecedent, ii) absence of consequent, iii) absence of antecedent and consequent.

The used rules are extracted by expert knowledge from a large set of possible rules and are representative of the normal operation of the AHU and are characterised by high physical interpretability. The introduction of innovative parameters (e.g. SUPP.DAY in faulty and normal conditions, support and confidence in the ACTUAL TIME LAG) allowed a robust selection of the most interesting association rules, minimising the effort required in the post-processing stage. Furthermore, an effective visualization of the temporal association rules was introduced with the aim of supporting energy managers in the interpretation of the temporal associations between operational variables in real-time.

The AFDD during non-transient period was performed by training and testing 21 CT models for providing a robust benchmark of the fault-free operation. The CT models are able to predict the discrete values of a target operational variable considering the values of the input variables both in the same and previous aggregation interval. The CTs showed high performance (i.e., high accuracy, precision and recall) in modeling all the variable relations that are characteristic of the operative condition of interest (i.e., 20 days of AHU operated in cooling mode).

Eventually, an additional CT was developed in order to perform fault diagnosis. The model showed an overall accuracy of 90% and consists of a set of intuitive rules easy to be implemented for detecting up to 11 typical faults in AHUs. However, the set of rules extracted can be then considered as a valid FDD solution only if applied on data consistent with the initial hypotheses (i.e., AHU operated in cooling mode).

Overall, the results obtained are characterised by robustness and high interpretability proving the effectiveness of the proposed methodology for ensuring a correct energy and operational management of the ventilation and air-conditioning process.

Even though the rule set and the classification models are tailored for the case study analysed, the outcomes of the process can be considered flexible and generalizable. The methodologies were conceived for being automatic and for effectively managing the redundancy, interpretability and physical meaningfulness of the association and classification rules. Moreover, the proposed AFDD process is conceived for quasi real-time implementation, also minimising the user contribution and paying attention to the optimisation of computational cost. To this purpose, the preliminary discretisation of the variables, performed trough the aSAX algorithm, proved to be particularly effective in extracting the crucial operational conditions of the AHU reaching the optimal trade-off between data reduction and information loss. Moreover, the association rules were extracted from an event-based dataset (i.e., database of transactions) where only information about the discrete-value changes of the operational variables is stored. As a consequence, the computational cost related to the mining of rules is strongly reduced, increasing the feasibility of such approach in real case studies. As a reference the whole analytics process takes about 45 min in terms of computational time on a computer equipped with quad-core processor Intel i7-3632QM CPU (2.20GHz) and 8GB RAM DDR4. In more detail the rule extraction phase takes more or less 10 minutes. It means that the most onerous parts of the analysis are represented by the pre-processing and post-mining phases. In the pre-processing phase the assessment of the optimal quantization of the time series through aSAX is validated by using more than 20 metrics (cluster validity indices included in the R Nbclust package [68]). Such calculation takes more than 10 minutes. In the post mining phase, the recalculation of support and confidence of each rule within the evaluated ACTUAL TIME LAG (instead of the window of 60-min), and the violation analysis performed on the testing dataset take about 20 minutes. For what concern the analysis of non-transient data, the development of each classification tree takes few seconds of computation and can be considered a task easily parallelizable. As a result, the impact of the analysis of non-transient data can be considered negligible in terms of computational cost compared to the pre-processing, rule extraction and rule post-mining. Indeed, in the perspective of a real-time implementation of the whole AFDD process, the update of the discretization intervals, set of association rules and estimation models can be accomplished during night-time while the fault detection and diagnosis tool can be run online during operation. For what concern the pre-processing stage, during the real-time operation the Hampel filter can still be used but considering that its intrinsic latency equal to *(Len-1)/2* should be added to the latency of the FDD process in detecting faults (in this case study the latency of the FDD process is equal to the length of the aggregation interval i.e., 15 min). For avoiding high latency in the analysis, *Len* can be reduced. As an alternative, other pre-processing algorithms, particularly suitable for the analysis of data streams, can be employed for detecting statistical outliers in real-time (i.e., before time t + 1 and without any look ahead) [71]. Further research will be then conducted to assess the scalability of the methodology to other operation modes and systems and to integrate it with knowledge driven-based analysis for better addressing the implementation issues characteristic of data-driven tools. Indeed, data-driven based FDD tools need a proper amount of data for the development of diagnosis models and cannot extrapolate beyond the range of training data [10]. It means that their capability in automatically extracting pattern from actual performance data is strictly related to the availability of pre-labelled monitored data (typically derived from AHU recommissioning or simulated data). On the contrary, knowledge driven-based approach can introduce domain knowledge and user experience into the FDD process [10], especially in the case initial information is not enough for deploying a data-driven process. In this perspective, a perfect integration of both approaches represents the main opportunity for significantly improve robustness, accuracy and generalizability of FDD tools conceived for application in building energy systems.

8 Acknowledgements

Particular thanks to ASHRAE for the permission given for the use of the data and documents from ASHRAE RP-1312.

References

[1] A. Capozzoli, T. Cerquitelli, M.S. Piscitelli, Chapter 11 – Enhancing energy efficiency in buildings through innovative data analytics technologies, in: D. Ciprian, F. Xhafa (Eds.), Pervasive Comput., 2016: pp. 353–389. https://doi.org/10.1016/B978-0-12-803663-1.00011-5.

[2] L. Pérez-Lombard, J. Ortiz, C. Pout, A review on buildings energy consumption information, Energy Build. 40 (2008) 394–398. https://doi.org/10.1016/j.enbuild.2007.03.007.

[3] Office of Energy Efficiency & Renewable Energy (EERE) U.S. Department of Energy, DOE Office of Energy Efficiency and Renewable Energy, Buildings energy databook, 2012. http://buildingsdatabook.eren.doe.gov/DataBooks.aspx.

[4] J. Proctor, Residential and Small Commercial Central air Conditioning; Rated Efficiency isn’t Automatic, Present. Public Sess. ASHRAE Winter Meet. (2004).

[5] K. Yan, C. Zhong, Z. Ji, J. Huang, Semi-supervised learning for early detection and diagnosis

of various air handling unit faults, Energy Build. 181 (2018) 75–83. https://doi.org/10.1016/j.enbuild.2018.10.016.

[6] R. Isermann, Fault-Diagnosis Systems: an introduction from fault detection to fault tolerance, Springer Science & Business Media, 2006. https://doi.org/10.1007/3-540-30368-5.

[7] J. Granderson, G. Lin, R. Singla, E. Mayhorn, P. Ehrlich, D. Vrabie, Commercial Fault Detection and Diagnostics Tools: What They Offer, How They Differ, and What’s Still Needed, 2018. https://doi.org/10.20357/B7V88H.

[8] H. Kramer, G. Lin, J. Granderson, C. Curtin, E. Crowe, Synthesis of Year One Outcomes in the Smart Energy Analytics Campaign Building Technology and Urban Systems Division, (2017).

[9] W. Kim, S. Katipamula, A review of fault detection and diagnostics methods for building systems, Sci. Technol. Built Environ. 24 (2018) 3–21. https://doi.org/10.1080/23744731.2017.1318008.

[10] Y. Zhao, T. Li, X. Zhang, C. Zhang, Artificial intelligence-based fault detection and diagnosis methods for building energy systems: Advantages, challenges and the future, Renew. Sustain. Energy Rev. 109 (2019) 85–101. https://doi.org/10.1016/j.rser.2019.04.021.

[11] C. Fan, F. Xiao, Z. Li, J. Wang, Unsupervised data analytics in mining big building operational data for energy efficiency enhancement: A review, Energy Build. 159 (2018) 296–308. https://doi.org/10.1016/j.enbuild.2017.11.008.

[12] Y. Yu, D. Woradechjumroen, D. Yu, A review of fault detection and diagnosis methodologies on air-handling units, Energy Build. 82 (2014) 550–562. https://doi.org/10.1016/j.enbuild.2014.06.042.

[13] A. Beghi, R. Brignoli, L. Cecchinato, G. Menegazzo, M. Rampazzo, F. Simmini, Data-driven Fault Detection and Diagnosis for HVAC water chillers, Control Eng. Pract. 53 (2016) 79–91. https://doi.org/10.1016/j.conengprac.2016.04.018.

[14] S. Wen, Jin; Li, ASHRAE 1312-RP Tools for Evaluating Fault Detection and Diagnostic

Methods for Air-Handling Unit [FINAL REPORT], (2011) 13. http://marketingdatabase.tat.or.th/download/article/research/1201finalreport.pdf.

[15] D. Dehestani, F. Eftekhari, Y. Guo, S. Ling, S. Su, H. Nguyen, Online Support Vector Machine Applicationfor Model Based Fault Detection and Isolationof HVAC System, Int. J. Mach. Learn. Comput. 1 (2011) 66–72. https://doi.org/10.7763/ijmlc.2011.v1.10.

[16] Y. Zhao, J. Wen, F. Xiao, X. Yang, S. Wang, Diagnostic Bayesian networks for diagnosing air handling units faults – part I: Faults in dampers, fans, filters and sensors, Appl. Therm. Eng. 111 (2017) 1272–1286. https://doi.org/10.1016/j.applthermaleng.2015.09.121.

[17] Y. Zhao, J. Wen, S. Wang, Diagnostic Bayesian networks for diagnosing air handling units faults - Part II: Faults in coils and sensors, Appl. Therm. Eng. 90 (2015) 145–157. https://doi.org/10.1016/j.applthermaleng.2015.07.001.

[18] T. Mulumba, A. Afshari, K. Yan, W. Shen, L.K. Norford, Robust model-based fault diagnosis for air handling units, Energy Build. 86 (2015) 698–707. https://doi.org/10.1016/j.enbuild.2014.10.069.

[19] R. Yan, Z. Ma, Y. Zhao, G. Kokogiannakis, A decision tree based data-driven diagnostic strategy for air handling units, Energy Build. 133 (2016) 37–45. https://doi.org/10.1016/j.enbuild.2016.09.039.

[20] M.K. Mchugh, Data-Driven Leakage Detection in Air-Handling Units on a University Campus, ASHRAE Annu. Conf. (2019).

[21] Z. Yu, F. Haghighat, B.C.M. Fung, L. Zhou, A novel methodology for knowledge discovery through mining associations between building operational data, Energy Build. 47 (2012) 430–440. https://doi.org/10.1016/j.enbuild.2011.12.018.

[22] P. Xue, Z. Zhou, X. Fang, X. Chen, L. Liu, Y. Liu, J. Liu, Fault detection and operation optimization in district heating substations based on data mining techniques, Appl. Energy. 205 (2017) 926–940. https://doi.org/10.1016/j.apenergy.2017.08.035.

[23] C. Zhang, X. Xue, Y. Zhao, X. Zhang, T. Li, An improved association rule mining-based

method for revealing operational problems of building heating, ventilation and air conditioning (HVAC) systems, Appl. Energy. 253 (2019) 113492. https://doi.org/10.1016/j.apenergy.2019.113492.

[24] C. Fan, F. Xiao, H. Madsen, D. Wang, Temporal knowledge discovery in big BAS data for building energy management, Energy Build. 109 (2015) 75–89. https://doi.org/10.1016/j.enbuild.2015.09.060.

[25] C. Fan, Y. Sun, K. Shan, F. Xiao, J. Wang, Discovering gradual patterns in building operations for improving building energy efficiency, Appl. Energy. 224 (2018) 116–123. https://doi.org/10.1016/j.apenergy.2018.04.118.

[26] A. Capozzoli, F. Lauro, I. Khan, Fault detection analysis using data mining techniques for a cluster of smart office buildings, Expert Syst. Appl. 42 (2015) 4324–4338. https://doi.org/10.1016/j.eswa.2015.01.010.

[27] I. Khan, A. Capozzoli, S.P. Corgnati, T. Cerquitelli, Fault detection analysis of building energy consumption using data mining techniques, Energy Procedia. 42 (2013) 557–566. https://doi.org/10.1016/j.egypro.2013.11.057.

[28] M. Dey, S.P. Rana, S. Dudley, Smart building creation in large scale HVAC environments through automated fault detection and diagnosis, Futur. Gener. Comput. Syst. (2018). https://doi.org/10.1016/j.future.2018.02.019.

[29] Z. Du, X. Jin, Y. Yang, Fault diagnosis for temperature, flow rate and pressure sensors in VAV systems using wavelet neural network, Appl. Energy. 86 (2009) 1624–1631. https://doi.org/10.1016/j.apenergy.2009.01.015.

[30] Z. Du, B. Fan, X. Jin, J. Chi, Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis, Build. Environ. 73 (2014) 1–11. https://doi.org/10.1016/j.buildenv.2013.11.021.

[31] Y. Guo, J. Wall, J. Li, S. West, Intelligent Model Based Fault Detection and Diagnosis for HVAC System Using Statistical Machine Learning Methods, in: ASHRAE 2013 Winter Conf.,

2013: pp. 1–8.

[32] S. Li, J. Wen, A model-based fault detection and diagnostic methodology based on PCA method and wavelet transform, Energy Build. 68 (2014) 63–71. https://doi.org/10.1016/j.enbuild.2013.08.044.

[33] X. Jin, Z. Du, Fault tolerant control of outdoor air and AHU supply air temperature in VAV air conditioning systems using PCA method, Appl. Therm. Eng. 26 (2006) 1226–1237. https://doi.org/10.1016/j.applthermaleng.2005.10.039.

[34] J. Liang, R. Du, Model-based Fault Detection and Diagnosis of HVAC systems using Support Vector Machine method, Int. J. Refrig. 30 (2007) 1104–1114. https://doi.org/10.1016/j.ijrefrig.2006.12.012.

[35] S. Wu, J.Q. Sun, Cross-level fault detection and diagnosis of building HVAC systems, Build. Environ. 46 (2011) 1558–1566. https://doi.org/10.1016/j.buildenv.2011.01.017.

[36] W.Y. Lee, J.M. House, N.H. Kyong, Subsystem level fault diagnosis of a building's air-handling unit using general regression neural networks, Appl. Energy. 77 (2004) 153–170. https://doi.org/10.1016/S0306-2619(03)00107-7.

[37] D. Dey, B. Dong, A probabilistic approach to diagnose faults of air handling units in buildings, Energy Build. 130 (2016) 177–187. https://doi.org/10.1016/j.enbuild.2016.08.017.

[38] D. Li, Y. Zhou, G. Hu, C.J. Spanos, Optimal Sensor Configuration and Feature Selection for AHU Fault Detection and Diagnosis, IEEE Trans. Ind. Informatics. 13 (2017) 1369–1380. https://doi.org/10.1109/TII.2016.2644669.

[39] S. Li, J. Wen, Application of pattern matching method for detecting faults in air handling unit system, Autom. Constr. 43 (2014) 49–58. https://doi.org/10.1016/j.autcon.2014.03.002.

[40] K. Yan, J. Huang, W. Shen, Z. Ji, Unsupervised learning for fault detection and diagnosis of air handling units, Energy Build. 210 (2020) 109689. https://doi.org/10.1016/j.enbuild.2019.109689.

[41] Y. Yan, P.B. Luh, K.R. Pattipati, Fault diagnosis of HVAC: Air delivery and terminal systems,

IEEE Int. Conf. Autom. Sci. Eng. 2017-Augus (2017) 882–887. https://doi.org/10.1109/COASE.2017.8256214.

[42] C. Zhong, K. Yan, Y. Dai, N. Jin, B. Lou, Energy efficiency solutions for buildings: Automated fault diagnosis of air handling units using generative adversarial networks, Energies. 12 (2019) 1–11. https://doi.org/10.3390/en12030527.

[43] T. Gao, B. Boguslawski, S. Marié, P. Béguery, S. Thebault, S. Lecoeuche, Data mining and data-driven modelling for air handling unit fault detection, E3S Web Conf. 111 (2019). https://doi.org/10.1051/e3sconf/201911105009.

[44] G. Li, Y. Hu, H. Chen, H. Li, M. Hu, Y. Guo, J. Liu, S. Sun, M. Sun, Data partitioning and association mining for identifying VRF energy consumption patterns under various part loads and refrigerant charge conditions, Appl. Energy. 185 (2017) 846–861. https://doi.org/10.1016/j.apenergy.2016.10.091.

[45] N.D. Pham, Q.L. Le, T.K. Dang, HOT aSAX: A novel adaptive symbolic representation for time series discords discovery, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 5990 LNAI (2010) 113–121. https://doi.org/10.1007/978-3-642-12145-6_12.

[46] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A Symbolic Representation of Time Series with Implication for Streaming Algorithms-SAX.pdf, in: DMKD ’03 Proc. 8th ACM SIGMOD Work. Res. Issues Data Min. Knowl. Discov., 2003: pp. 2–11.

[47] P.-N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Pearson, Boston, 2005.

[48] R. Agrawal, R. Srikant, Mining Sequential patterns: Generalizations and performance improvements, in: Springer Verlag, 1996: pp. 3–15.

[49] J. Pei, Mining sequential patterns efficiently by prefix-projected pattern growth, in: Int. Conf. Data Eng. (ICDE2001), April, 2001.

[50] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, M.C. Hsu, Mining sequential patterns by pattern-growth: The prefixspan approach, IEEE Trans. Knowl. Data

Eng. 16 (2004) 1424–1440. https://doi.org/10.1109/TKDE.2004.77.

[51] M.J. Zaki, SPADE: An efficient algorithm for mining frequent sequences, Mach. Learn. 42 (2001) 31–60. https://doi.org/10.1023/A:1007652502315.

[52] J. Ayres, J. Flannick, J. Gehrke, T. Yiu, Sequential pattern mining using A bitmap representation, Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (2002) 429–435. https://doi.org/10.1145/775107.775109.

[53] J. Pei, J. Han, W. Wang, Mining sequential patterns with constraints in large databases, Int. Conf. Inf. Knowl. Manag. Proc. (2002) 18–25. https://doi.org/10.1145/584796.584799.

[54] Y.-L. Chen, T.-K. Huang, Discovering fuzzy time-interval sequential patterns in sequence databases, IEEE Trans. Syst. Man, Cybern. Part B. 35 (2005) 959–972.

[55] Y. Hirate, Generalized Sequential Pattern Mining with Item Intervals, 1 (2006) 51–60.

[56] Y.-L. Chen, M.-C. Chiang, M.-T. Ko, Discovering time-interval sequential patterns in sequence databases, Expert Syst. Appl. 25 (2003) 343–354. https://doi.org/https://doi.org/10.1016/S0957-4174(03)00075-7.

[57] F.J. Martínez-De-Pisón, A. Sanz, E. Martínez-De-Pisón, E. Jiménez, D. Conti, Mining association rules from time series to explain failures in a hot-dip galvanizing steel line, Comput. Ind. Eng. 63 (2012) 22–36. https://doi.org/10.1016/j.cie.2012.01.013.

[58] F.J. Martínez-de-Pisón Ascacíbar, A. Pernía Espinoza, F. Martínez, Roberto, R. Escribano García, P. Guillén Rondón, D. Conti Guillén, System for uncovering hidden knowledge in real time for the analysis of environmental and agricultural processes, XIII Int. Conf. Proj. Eng. (2009).

[59] G. Kaur, Association Rule Mining: A survey, Int. J. Comput. Sci. Inf. Technol. 5 (2014) 2320–2324.

[60] T.M. Therneau, E.J. Atkinson, An introduction to recursive partitioning using the rpart routines, 1997.

[61] A. Capozzoli, M.S. Piscitelli, S. Brandi, D. Grassi, G. Chicco, Automated load pattern learning

and anomaly detection for enhancing energy management in smart buildings, Energy. 157 (2018) 336–352. https://doi.org/10.1016/j.energy.2018.05.127.

[62] R.K. Pearson, Data cleaning for dynamic modeling and control, Eur. Control Conf. ECC 1999 - Conf. Proc. (2015) 2584–2589. https://doi.org/10.23919/ecc.1999.7099714.

[63] S. Li, A Model-Based Fault Detection and Diagnostic Methodology for Secondary HVAC Systems, Drexel Univ. (2009).

[64] M. Kim, S.H. Yoon, P.A. Domanski, W. Vance Payne, Design of a steady-state detector for fault detection and diagnosis of a residential air conditioner, Int. J. Refrig. 31 (2008) 790–799. https://doi.org/10.1016/j.ijrefrig.2007.11.008.

[65] C.W. Roh, M. Kim, H.S. Kim, M.S. Kim, Design Method Of Steady State Detector For Multi-Evaporator Heat Pump System With Decomposition Analysis Technique, (2010).

[66] C. Miller, Z. Nagy, A. Schlueter, A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings, Renew. Sustain. Energy Rev. 81 (2018) 1365–1377. https://doi.org/10.1016/j.rser.2017.05.124.

[67] A. Dexter, J. Pakanen, Demonstrating Automated Fault Detection and Diagnosis Methods in Real Buildings, in: Proc. VTT Symp. 217, 2001: p. 381. https://doi.org/951-38-5725-5.

[68] M. Charrad, N. Ghazzali, V. Boiteau, A. Niknafs, NbClust: An R Package for Determining the Relevant Number of Clusters in a Data Set Malika, J. Stat. Softw. 61 (2014). https://doi.org/10.18637/jss.v061.i06.

[69] R Core Team, R: A Language and Environment for Statistical Computing, (2017). http://www.r-project.org/.

[70] M. Hahsler, B. Grun, K. Hornik, arules – A Computational Environment for Mining Association Rules and Frequent Item Sets, J. Stat. Softw. 14 (2005) 1–6. papers2://publication/uuid/388D2132-AF39-463F-8D87-91A45FA1E26D.

[71] S. Ahmad, S. Purdy, Real-Time Anomaly Detection for Streaming Analytics, (2016). http://arxiv.org/abs/1607.02480.

Appendix A

Table A. Discretization intervals for all the analysed variables.

Variable	ID	Unit	Sym. A	Sym. B	Sym. C	Sym. D	Sym. E
SF_WAT	1	[W]	< 522 **OFF**	522 – 1265 **ON**	1265 – 2440 **ON**	> 2440 **ON**	-
RF_WAT	2	[W]	< 181 **ON**	181 - 337 **ON**	336 – 502 **ON**	> 502 **ON**	-
SA_CFM	3	[m³/h]	< 591 **OFF**	591 – 2276 **ON**	2276 – 3414 **ON**	3414 – 4706 **ON**	> 4706 **ON**
RA_CFM	4	[m³/h]	< 838 **OFF**	838 – 2712 **ON**	2712 – 3527 **ON**	> 3527 **ON**	-
OA_CFM	5	[m³/h]	< 477 **ON**	477 – 1146 **ON**	> 1146 **ON**	-	-
SA_TEMP	6	[°C]	< 15,9 **ON**	15,9 – 23,5 **ON**	23,5 – 32,4 **ON**	> 32,4 **ON**	-
MA_TEMP	7	[°C]	< 20,3 **ON**	20,3 – 30,8 **ON**	> 30,8 **ON**	-	-
RA_TEMP	8	[°C]	< 25,7 **ON**	25, 7 – 31,5 **ON**	>31,5 **ON**	-	-
HWC_DAT	9	[°C]	< 20,2 **ON**	20,2 – 26 **ON**	26 – 35,7 **ON**	> 35,7 **ON**	-
CHWC_DAT	10	[°C]	< 14,4 **ON**	14,4 – 22 **ON**	22 – 30,7 **ON**	> 30,7 **ON**	-
SF_DP	11	[Pa]	< 324 **OFF**	324 – 562 **ON**	562 – 770 **ON**	> 770 **ON**	-
RF_DP	12	[Pa]	< 46 **ON**	46 – 114 **ON**	> 114 **ON**	-	-
SF_SPD	13	[%]	< 50 **OFF**	50 – 67 **ON**	67 – 75 **ON**	75 – 87 **ON**	> 87 **ON**
RF_SPD	14	[%]	< 30 **OFF**	30 – 43 **ON**	43 – 57 **ON**	57- 69 **ON**	> 69 **ON**
OA_TEMP	15	[°C]	< 18,3 **ON**	> 18,3 **ON**	-	-	-
CHWC_EWT	16	[°C]	< 1,3 **ON**	1,3 – 2,8 **ON**	2,8 – 6,7 **ON**	> 6,7 **OFF**	-
CHWC_LWT	17	[°C]	< 13,3 **ON**	13,3 – 19,9 **ON**	19,9 – 21,2 **ON**	> 21,2 **OFF**	-
CHWC_GPM	18	[m³/h]	< 0,9 **ON**	0,9 – 1,7 **ON**	> 1,7 **ON**	-	-
E_ccoil	19	[kW]	< 11,7 **ON**	> 11,7 **ON**	-	-	-
CHWC_VLV	20	[%]	< 41 **ON**	41 – 75 **ON**	> 75 **ON**	-	-
EA_DMPR	21	[%]	< 20 **ON**	20 – 70 **ON**	> 70 **ON**	-	-
OA_DMPR	22	[%]	< 20 **ON**	20 – 47 **ON**	47 – 77 **ON**	> 77 **ON**	-
RF_SST	23	[-]	< 0,5 **OFF**	> 0.5 **ON**	-	-	-

Table A summarizes the transformation results obtained, with the specification of the numerical range corresponding to each symbol for all the analysed operational variables.

Table B. Most representative extracted temporal association rules.

ID N°	Antecedent	Consequent	Supp.	Conf.	ACTUAL TIME LAG	SUPP. DAY FAULTY
1077	SF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A]	0.70	0.8	15	0.27
1078	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A]	0.70	0.8	15	0.27
1526	SF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], CHWC_LWT [B-A]	0.70	0.8	15	0.27
1527	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], CHWC_LWT [B-A]	0.70	0.8	15	0.27
1864	SF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], CHWC_LWT [B-A], SA_TEMP [B-A]	0.75	0.8	15	0.27
1865	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], CHWC_LWT [B-A], SA_TEMP [B-A]	0.75	0.8	15	0.27
8661	RF_SPD [A-B], CHWC_LWT [C-B], EA_DMPR [A-B]	CHWC_DAT [B-A], RF_SPD [B-C]	0.9	1	30	0.09
8750	RF_SPD [A-B], EA_DMPR [A-B], RF_SST [A-B]	CHWC_DAT [B-A], RF_SPD [B-C]	0.8	0.89	30	0
6255	RF_SPD [A-B], CHWC_LWT [C-B], RA_CFM [A-B]	CHWC_DAT [B-A], RF_SPD [B-C], CHWC_LWT [B-A]	0.89	0.8	15	0.18
6256	RF_SPD [A-B], CHWC_LWT [C-B], RF_SST [A-B]	CHWC_DAT [B-A], RF_SPD [B-C], CHWC_LWT [B-A]	0.89	0.8	15	0.09
6226	RF_SPD [A-B], CHWC_LWT [C-B], RF_SST [A-B]	CHWC_DAT [B-A], RF_SPD [B-C], SA_TEMP [B-A]	0.89	0.8	15	0.09
6936	RF_SPD [A-B], CHWC_LWT [C-B]	CHWC_DA T [B-A], RF_SPD [B-C], SA_TEMP [B-A]	0.889	0.8	15	0.18
1933	SF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_LWT [B-A], SA_TEMP [B-A]	0.75	0.8	15	0.27
1934	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_LWT [B-A], SA_TEMP [B-A]	0.75	0.8	15	0.27
5257	RF_SPD [A-B], EA_DMPR [A-B], RA_CFM [A-B]	RF_SPD [B-C]	0.82	1	30	0.18
5259	RF_SPD [A-B], EA_DMPR [A-B], RF_SST [A-B]	RF_SPD [B-C]	0.82	1	30	0.09
6415	RF_SPD [A-B], CHWC_LWT [C-B], RF_WAT [A-B]	RF_SPD [B-C], CHWC_LWT [B-A]	0.8	0.8	15	0.09
6416	RF_SPD [A-B], CHWC_LWT [C-B], RF_DP [A-B]	RF_SPD [B-C], CHWC_LWT [B-A]	0.8	0.8	15	0.09
6126	RF_SPD [A-B], CHWC_LWT [C-B], RF_WAT [A-B]	RF_SPD [B-C], CHWC_LWT [B-A], SA_TEMP [B-A]	0.89	0.8	15	0.09
8309	RF_SPD [A-B], CHWC_LWT [C-B], EA_DMPR [A-B]	RF_SPD [B-C], CHWC_LWT [B-A], SA_TEMP [B-A]	0.78	0.78	15	0.09
15268	RF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]	RF_SPD [B-C], SA_TEMP [B-A]	0.8	0.89	30	0
15269	RF_SPD [A-B], EA_DMPR [A-B], RF_DP [A-B]	RF_SPD [B-C], SA_TEMP [B-A]	0.8	0.89	30	0
1253	**SF_SPD [A-B], EA_DMPR [A-B], RF_WAT [A-B]**	**SA_TEMP [B-A]**	**0.70**	**0.8**	**15**	**0.27**
1254	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	SA_TEMP [B-A]	0.70	0.8	15	0.27
1406	SF_WAT [A-B], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], SA_TEMP [B-A]	0.70	0.8	15	0.27
5240	CHWC_EWT [D-C], EA_DMPR [A-B], RF_WAT [A-B]	CHWC_DAT [B-A], SA_TEMP [B-A]	0.70	0.8	30	0.27

Table B reports 26 rules (two for each unique consequent transaction) extracted from the transient dataset with the specification of the event chains of antecedent and consequent, the value of support and confidence within the ACTUAL TIME LAG and its duration (evaluated on the training dataset), and the $\text{SUPP.DAY}_{\text{FAULTY}}$ (evaluated on the testing dataset).