Automation in Construction 49 (2015) 1–17

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Automated daily pattern filtering of measured building performance data

Clayton Miller *, Zoltán Nagy, Arno Schlueter

Architecture & Building Systems, Institute of Technology in Architecture (ITA), ETH Zürich, Switzerland

ARTICLE INFO

Article history:
Received 2 April 2014
Received in revised form 6 August 2014
Accepted 11 September 2014
Available online 7 October 2014

Keywords:
Building efficiency
Building performance analysis
Knowledge discovery
Clustering
Temporal data mining
Occupancy patterns
Diversity factors

ABSTRACT

The amount of sensor data generated by modern building systems is growing rapidly. Automatically discovering the structure of diurnal patterns in this data supports implementation of building commissioning, fault detection and retrofit analysis techniques. Additionally, these data are crucial to informing design professionals about the efficacy of their assumptions and strategies used in performance prediction simulation models. In this paper, we introduce *DayFilter*, a day-typing process that uses Symbolic Aggregate approXimation (SAX), motif and discord extraction, and clustering to detect the underlying structure of building performance data. Discords, or infrequent daily patterns, are filtered and tagged for deeper, detailed analysis of potential energy savings opportunities. Motifs, or the most frequent patterns, are detected and further aggregated using k-means clustering. This procedure is designed for application on whole building and sub-system metrics from hierarchical building and energy management systems (BMS/EMS). The process transforms quantitative raw data into qualitative subgroups based on daily performance similarity and visualizes them using expressive techniques. We apply *DayFilter* on 474 days of example data from an international school campus in a tropical climate and 407 days of data from an office building from a temperate European climate. Discords are filtered resulting in 17 and 22 patterns found. Selected discords are investigated and many correlate with specific failures and energy savings detected by the on-site operations staff. Six and ten motif candidates are detected in the two case studies. These motifs are then further aggregated to five and six performance clusters that reflect the typical operational behavior of those projects. We discuss the influence of the parameter choices and provide initial parameter settings for the *DayFilter* process.

1. Introduction

Performance and energy data generation in the built environment is rapidly growing [1]. Modern building controls and management systems are improving in their ability to acquire and store measured data as the technology improves. This phenomenon results in vast portfolios of collected data from heterogeneous buildings. Fig. 1 illustrates a general example of various types of measured data from a conventional commercial building. Whole building performance is influenced by layers of complex measurement systems. Aggregated performance metrics or sensors are often measured or calculated at each level of this hierarchy in order to condense the exponential detailed sensor data downstream.

In addition to the increase in building performance data, there is a growing awareness of the gap in performance between building design and operations [2–6]. Multiple studies have documented and validated this phenomenon, with the most extreme mismatch finding measured energy consumption at 5 times predicted consumption for a commercial building [2]. A framework for investigating this gap emphasizes more robustly leveraging measurement data and ensuring that research in this field aligns with actual building engineering practices [3].

From the conventional operations and management side, this performance gap is generally addressed through the use of various performance analysis techniques. The literature describes two major categories of building analysis: top-down, whole building techniques and bottom-up, device-focused diagnostics [7,8].

Top-down approaches such as Energy Information Systems (EIS) are designed to qualify the building's overall performance health. They leverage the whole building and sub-systems level data to show how well a building performs compared to its peers (benchmarking) or simple tracking metrics. Despite their high-level usefulness, these techniques have a limited amount of insight and ignore much of the detailed digital data created in recently built or renovated high performance buildings [9]. In addition, they often aren't able to leverage higher frequency, sub-hourly measurements.

Bottom-up, component level approaches such as commissioning and automated fault detection and diagnostics (AFDD) are more effective at detecting the root cause of performance problems. A review of AFDD approaches for building systems diagnostics describes three general categories: Qualitative Model-based, Quantitative Model-based, and Process History Based [10]. The first two categories often require an understanding of the impact of each detailed data stream in order

* Corresponding author. Tel.: +41 446332991.
E-mail address: miller.clayton@arch.ethz.ch (C. Miller).

http://dx.doi.org/10.1016/j.autcon.2014.09.004
0926-5805/

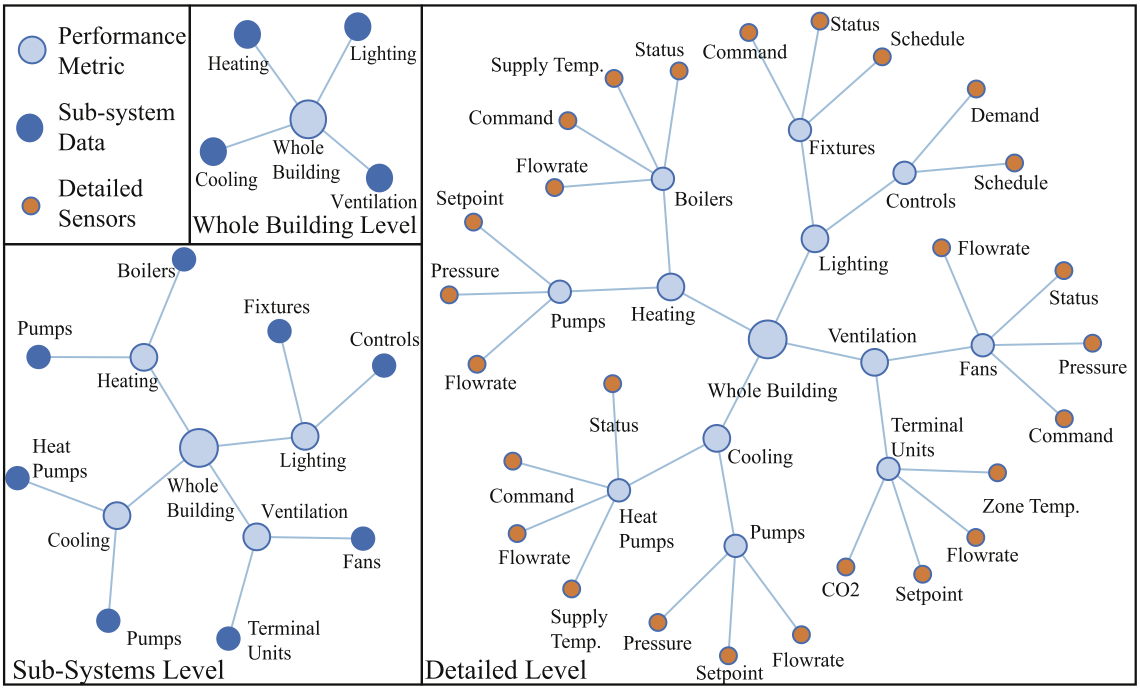

Fig. 1. Example of levels of building performance data complexity.

to set thresholds or parameters for detection of anomalies. Process history based methods rely on large amounts of empirical, measured data to create statistical models or use pattern recognition to find operational anomalies. Only process history based approaches are identified as useful with little a priori knowledge. However, they are implied to be inferior due to weaknesses such as the inability to extrapolate beyond a range of training data, amount of data necessary, and being specific to a particular dataset.

Beyond AFDD and EIS, another very active research topic is the process of calibrating the building simulation model developed in the design phase with measured performance data from the operations phase [11]. The benefits of such a process have long been lauded as key in understanding of the performance gap; first, in identifying the deficiencies in modeling engines and assumptions and, second, in investigating potential performance deviation in operations. This field was one of the first to investigate the use of *day-typing* as a means of parameter reduction of measured data for simulation feedback. Much of the literature in simulation model calibration treats measured raw data preparation and day-typing in a manual way, often ignoring the shape and magnitude patterns and relying on rules-of-thumb regarding schedule creation. These approaches add to the cost, time and lack of automation burden that calibration suffers from in real implementation.

A comprehensive study of building performance tracking was completed by the California Commissioning Collaborative (CACx) and funded by the California Energy Commission (CEC) to characterize the technology, market, and research landscape in the United States. Three of the key tasks in this project focused on establishing the state of the art [12], characterizing available tools and barriers to adoption [8], and establishing standard performance metrics [13]. These reports were accomplished through investigation of the available tools and technologies on the market as well as discussions and surveys with building operators and engineers. The common theme amongst the interviews and case studies was the *lack of time and expertise* on the part of the involved operations professionals. The findings showed that installation time and cost were driven by the need for a controls engineer to develop a full understanding of the building and systems. We interpret these results as a latent need for techniques that take into consideration the people, process, and philosophy aspects of the performance analysis equation [14]. The effort described in this paper addresses this challenge by focusing on automatically finding insight in large, unstructured building performance datasets *as part of an analysis process*.

1.1. Parameter-light exploratory analysis for building performance data

We draw inspiration from other time-series analysis and visualization applications in order to address the progression of data mining in the building industry. One emerging trend is that "data mining algorithms should have as few parameters as possible, ideally none. A parameter-free algorithm prevents us from imposing our prejudices and presumptions on the problem at hand and let the data itself speak to us [15]." This approach is known as *parameter-free* or *parameter-light* data mining. The efficacy of these algorithms has been proven comparable or better than many more complex, traditional time-series data mining approaches [15].

An emerging circumstance in the building industry is the consolidated analysis of multiple buildings or portfolios by third-party experts [16, 17]. The responsibility of managing and mining performance data is shifted from operations staff to data and building science experts who develop specific skills and efficiencies of scale. This scaled analysis and intervention addresses the previously-mentioned time and expertise deficiency and the cost effectiveness of building performance investigations. This scenario requires computation techniques which, on one hand, condense information more effectively than conventional top-down techniques, and on the other hand, requires less a priori knowledge than bottom-up, component-level approaches. Therefore, exploratory visualization and data mining techniques could be designed as part of a process to bridge these gaps. Our research combines traditional AFDD with exploration techniques such as time-series pattern recognition and visualization.

We propose a new context for the process history based methods found in the literature by testing their usefulness not as a full-scale automated fault detection and diagnostics (AFDD) approach, but as an exploratory step between the top-down and bottom-up paradigms within a process of analysis. The goal is to reduce the expert intervention needed to utilize measured raw data, most importantly in the initial analysis stages. The question addressed is whether parameter-light techniques can provide meaningful insight for bridging the performance gap and implementing other techniques in a more automated way. Additionally, there is a goal of automated filtering of the typical performance behavior of a building in order to better understand whether the building is performing as designed. These objectives are useful in the post-occupancy phase for stakeholders such as designers and architects and non-technical occupants and managers.

This paper introduces *DayFilter*, a process with three key contributions related to post-occupancy exploratory analysis:

- Automated tagging of the *most common daily profiles* whose patterns occur most frequently. These subsequences are defined as *motif candidates* and they can be used to characterize general daily performance profiles.
- Automated tagging of *potentially anomalous individual daily profiles* whose patterns occur least frequently in the dataset. These subsequences are defined as *discord candidates* and are further investigated in the development of rule-based diagnostics.
- Both types of analysis are presented using combinations of visualizations that are expressive and interpretable by analysts as part of a larger process.

These efforts are meant to assist in transforming high frequency collected data into more qualitative or simplified means so as to be compared to design phase assumptions or to be used to implement more sophisticated bottom-up techniques such as AFDD. This target assists in bridging the previously mentioned performance gap between design and operations.

The paper is organized as follows. In Section 2, we discuss the background of whole building performance analysis and give an overview of the time-series data mining techniques investigated with application to the buildings context. Section 3 explains *DayFilter*, a process of analysis used to filter information from raw measured building data. Then, Section 4 outlines two real-world case studies as applications of the process. Section 5 discusses the achievement of the parameter-light goals by investigating the influence of the input parameters and statistical analysis of the tightness of fit of the clustering process. Finally, Section 6 discusses insight gained from the application process with respect to the achievement of the objectives.

2. Background and related work

Whole building performance diagnostic approaches have been investigated for many years, especially since digital data collection systems became common in commercial buildings. This section reviews previous efforts in conventional building performance analysis approaches such as AFDD and building simulation model calibration. The intent of these reviews is to show the context in which the proposed exploratory analysis process is meant to supplement. We provide a review of data-driven approaches that specifically utilize historic process data. We also discuss the background of temporal data symbolic aggregation and the mining techniques utilized in this paper.

2.1. Automated fault detection and diagnostics

Automated fault detection and diagnostics (AFDD) is a process in which abnormal conditions in specific equipment are found and the root cause is identified automatically, i.e. without human intervention [10]. Quantitative methods are based on the development of detailed physical, white-box models that are analytically compared to measured building data. Recent quantitative methods developed still require an extensive amount of intimate knowledge in order to construct simulation models in programs such as EnergyPlus [18]. Qualitative methods rely on expert analysis of specific building systems and the setting of thresholds or alarms, often on much of the detailed sensor data [19]. Other methods have been developed that do not rely on simulation models and use decoupling-based techniques and virtual sensors to find faults in specific systems [20]. Many of the AFDD approaches in the literature focus on the efficacy of detecting and diagnosing faults and neglect the people and process issues related to implementation and utilization of the systems. This gap can be addressed through investigation of data mining and exploratory methods.

2.2. Building simulation model calibration

Building simulation model calibration is a promising approach for bridging the gap between design and operational performance [11,21]. A building data model hierarchy matching approach was created to analyze the relationship between simulated and measured datasets [22]. Calibration is utilized in AFDD quantitative methods outlined previously, to validate model assumptions, as a tool for various measurement and verification procedures (M&V), and in performing investment-grade retrofit analysis. Previous *day-typing* processes have been developed to reduce measurement data complexity for simulation calibration [23,24]. There have been several more recent attempts to convert measured data into occupancy patterns and diversity factors [25–27]. These studies require manual intervention to describe the basic occupancy schedule partitions and removal of anomalous days such as holidays and abnormal performance. Overall, reviews of the literature in this area have found lack of standards, methods for identification of discrepancies, and automation, which all contribute to the process suffering from under-utilization in the industry. More work in exploratory and automated parameter reduction is necessary to further enhance automation of the calibration process.

2.3. Data-driven building performance analysis

Building process history-based, or data-driven, techniques are the third major category of diagnostics in buildings. The key feature uniting many of these studies is their focus on extracting knowledge from measured datasets without detailed intervention from an expert. The concept of whole building diagnostics was established in the 1990s as a means of finding major disruptions in performance from high-level system metrics and probabilistic regression models [28]. Various unsupervised learning or exploratory methods have become more popular in extracting information. For example, clustering has been used as a means of finding similar daily performance [29–31], detecting deviant performance [7], enhancing analyst productivity [32], and supplementing controls optimization algorithms [33,34]. More generalized building data mining methodologies have been developed using outliers detection [35], multiple linear regression models [36], fuzzy behavior modeling [37], generalize additive models [38], and association rule mining [39]. Wavelet transformations and clustering have been used in large scale classification of electrical demand profiles of hundreds of buildings [40]. The newest advanced methods combine a number of different techniques in an analysis framework [41,42]. Semi-supervised methods have been introduced which allow expert intervention in the process to leverage both unsupervised and supervised machine learning [43]. All of these approaches focus on the efficacy of detecting and diagnosing various potential issues in buildings, but most lack discussion or design dedicated to the implementation within the context or interpretability of the results.

2.4. Temporal data mining

Temporal data mining for performance monitoring focuses generally on the extraction of patterns and model building of time series data. These techniques are, in some ways, similar to many existing building performance analysis approaches, however different concepts and terminology are used. Two key concepts to understand when applying data mining to buildings are that of *motifs* and *discords*. A motif is a common subsequence pattern that has the highest number of non-trivial matches [44], thus, a pattern that is found frequently in the dataset. A discord, on the other hand, is defined as a subsequence of a time series that has the largest distance to its nearest non-self match [45]. It is a subsequence of a univariate data stream that is least like all other non-overlapping subsequences and is, therefore, a rare pattern that diverges from the rest of the dataset. These definitions are more general than that of a *fault* and therefore more appropriate for our goal of higher level information extraction with less parameter setting. In short, we want to efficiently find *interesting or infrequent* behavior and not create a detailed list of specific problems that could be occurring in individual systems.

In order to work with common temporal mining approaches, we utilize the extensive work in the development of the Symbolic Aggregate approXimation (SAX) representation of time-series data [46]. SAX allows discretization of time series data which facilitates the use of various motif and discord detection algorithms. The process breaks time series data into subsequences which are converted into an alphabetic symbol. These symbols are combined to form strings to represent the original time series enabling various mining and visualization techniques. In terms of application, an example of a process using SAX-based techniques is the VizTree tool that uses augmented suffix tree visualizations designed for usability by an analyst [47,48]. A specific application of VizTree is the analysis of collected sensor data from an impending space craft launch in which thousands of telemetry sensors are feeding data back to a command center where experts are required to interpret the data. Visualization and filtering tools are needed that allow a natural and intuitive transfer of mined knowledge to the monitoring task. Human perception of visualizations and the algorithms behind them must work in unison to achieve understanding of large amounts of novel data streams.

SAX has been used on building performance data before in a few studies focused on data center chilled water plants and it was found effective in detecting the most efficient control strategies [49]. The same research was used to create a visual exploration tool of high frequency time series data [50]. Despite these efforts, our review of the literature found a lack of tools or processes similar to the VizTree tool for day-types that fit in our targeted context of bridging the performance gap. We will introduce a new process focused on combining temporal approximation, filtering, and visualization.

3. Our approach

We present a process of analysis, *DayFilter*, as an application of temporal data mining to building performance data. It includes five steps designed to incrementally filter structure from daily raw measured performance data. These steps, as seen in Fig. 2, are intended to bridge the gap between contemporary top-down and bottom-up techniques. The arrows in the diagram denote the execution sequence of the steps. Note that steps 3, 4, and 5 produce results applicable to the implementation of bottom-up techniques.

Whole building and subsystem metrics are targeted for analysis in order to determine high level insight. The process begins with a data preprocessing step which removes obvious point-based outliers and accommodates for gaps in a univariate dataset of variable length. Next, the raw data is transformed into the SAX time-series representation for dimensionality reduction by creating groups of SAX words from daily windows. This step enables the quick detection of *discords*, or daily patterns of performance that fall outside what is considered normal in the dataset according to the frequency of patterns. The discords are filtered out for future investigation while the remaining set of SAX words is clustered to create performance *motifs*, or the most common daily profiles. The additional clustering step beyond the SAX transformation and filtering adds the ability to further aggregate daily profiles beyond the SAX motif candidates. These clusters are useful in characterizing what can be considered as *standard* performance. Finally, these data are presented using visualization techniques as an aid to identify and interpret the questionable discords and the common clusters. In the following simplified example, we detail each of these steps. The input parameter selections in this section are based on suggestions from other studies using SAX aggregation and clustering approaches. Additional discussion of parameter selection is presented in Section 5.

3.1. Data preprocessing

As in any data mining approach, data preprocessing is an important step to clean and standardize the data. In our approach, we first remove extreme point measurements that fall outside of three standard

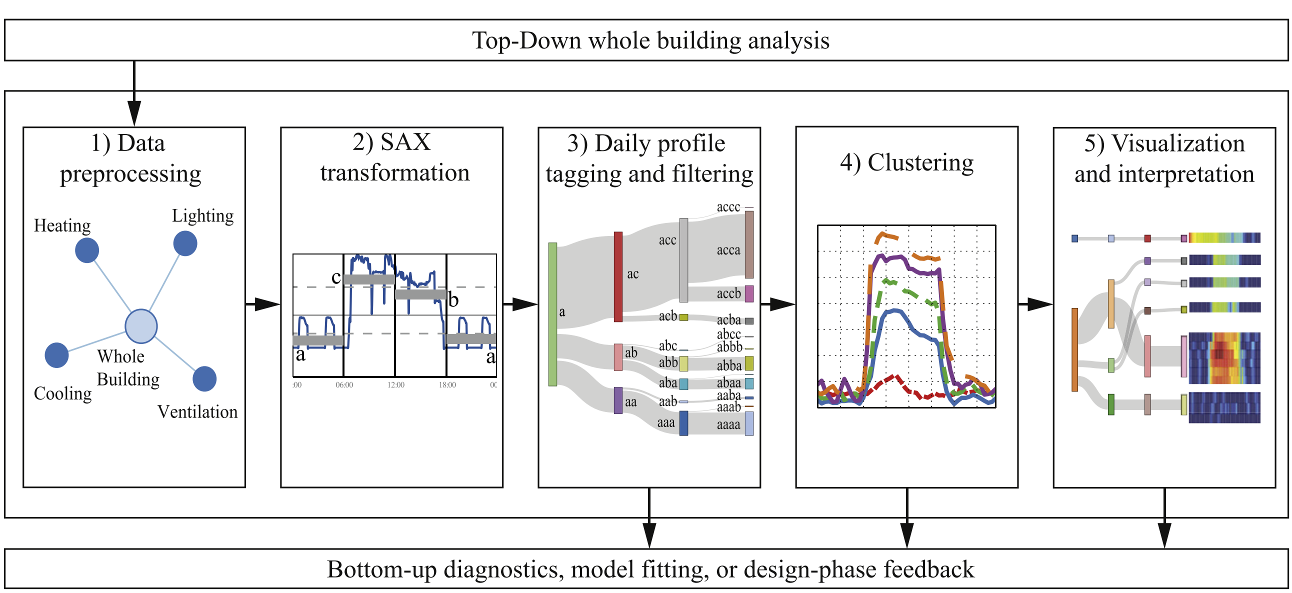

Fig. 2. Diagram of the five steps in the *DayFilter*.

Table 1
Example breakpoint lookup table from Keogh et al. [45] for $A = 3, 4$, and 5 calculated from a Gaussian distribution.

βi \ A	3	4	5
β_1	-0.43	-0.67	-0.84
β_2	0.43	0	-0.25
β_3		0.67	0.25
β_4			0.84

deviations, 3σ, of the mean, μ, of the selected univariate data stream $x(t)$. The data are then normalized in order to create a dataset, $Z(t)$ with an approximate 0 mean and a standard deviation of close to 1 [51]:

$$Z(t) = \frac{x(t) - \mu}{\sigma} \tag{1}$$

3.2. Symbolic Aggregate approXimation (SAX) transformation

In the second step, we transform $Z(t)$ into a symbolic representation using SAX. It is one of the many means of representing time-series data to enhance the speed and usability of various analysis techniques. SAX is a type of Piecewise Aggregate Approximation (PAA) representation developed by Keogh et al. and it has been used extensively in numerous applications [52].

In brief, the SAX transformation is as follows. The normalized time-series, $Z(t)$, is first broken down into N individual non-overlapping subsequences. This step is known as *chunking*, and the period length N is based on a context-logical specific period [48]. In our situation N is chosen as 24 h due to the focus on daily performance characterization. Each chunk is then further divided into W equal sized segments. The mean of the data across each of these segments is calculated and an alphabetic character is assigned according to where the mean lies within a set of vertical breakpoints, $B = \beta_1, \ldots, \beta_{a-1}$. These breakpoints are calculated according to a chosen alphabet size, A, to create equiprobable regions based on a Gaussian distribution, as seen in Table 1.

Based on a chosen value of W segments and alphabet size A, each N size window is transformed into a SAX word. An example of this process is seen in Fig. 3. This example shows two daily profiles which are converted to the SAX words, *acba* and *abba*. The SAX word is useful from an interpretation point of view in that each letter corresponds consistently to a subsequence of data from the daily profile. For example, the first letter explains the relative performance for the hours of midnight to 6:00 AM. Therefore if the size of A is set to 3, a SAX word whose first letter is a would have low, b would indicate average, and c would correspond to high consumption. Larger sizes of A would create SAX words with a more diverse range of characters and would capture more resolution magnitude-wise.

It should be noted that we do not normalize the individual subsequences, N, independently. This particular decision is divergent from the generalized shape-based discord approaches and is due to the fact that, in this level of analysis and the context of building performance data, we are interested in discovering interesting subsections based on both magnitude and shape.

The targeted benefits of using SAX in this scenario are that discretization uniformly reduces the dimensionality and creates sets of words from the daily data windows. This transformation allows the use of hashing, filtering, and clustering techniques that are commonly used to manipulate strings [52].

3.3. Daily profile tagging and filtering

Once the SAX words are created, we are interested in visualizing each pattern and tagging each type as either a motif or discord. The results of applying the SAX process to a two-week sample power dataset are shown in Fig. 4. The diagram shows how each daily chunk of high frequency data is transformed into a set of SAX characters. In this example we used an alphabet size, A, of 3 and a subsequence period count, W, of 4 with each character aggregating the data from 6 h of each profile. These parameters are the same as used in the more simplified two day example from Fig. 3.

Fig. 5 visualizes the frequency of the SAX strings and substrings in the form of an augmented suffix tree. Suffix trees have been an integral part of string manipulation and mining for decades [53]. Augmented suffix trees enable a means of visualizing the substring patterns to show frequency at each level. This figure incorporates the use of a sankey diagram to visualize the tree with each substring bar height representing the number of substring patterns existing through each window of the day-types. The more frequent patterns are categorized as *motifs*, or patterns which best describe the average behavior of the system. One can see the patterns with the lower frequencies and their indication as *discords*, or subsequences that are least common in the stream.

Heuristically, we set a decision threshold to distinguish between motifs and discords. This threshold can be based on the word frequency count for each pattern as a percentage of the count of all observations. This threshold can be tuned to result in a manageable number of discord candidates to be further analyzed. More details pertaining to setting this threshold will be discussed the applied case studies.

In the two week example, this process yields two patterns which have a frequency greater than one and thus are the motif candidates. A manual review of the data confirms that those patterns match with

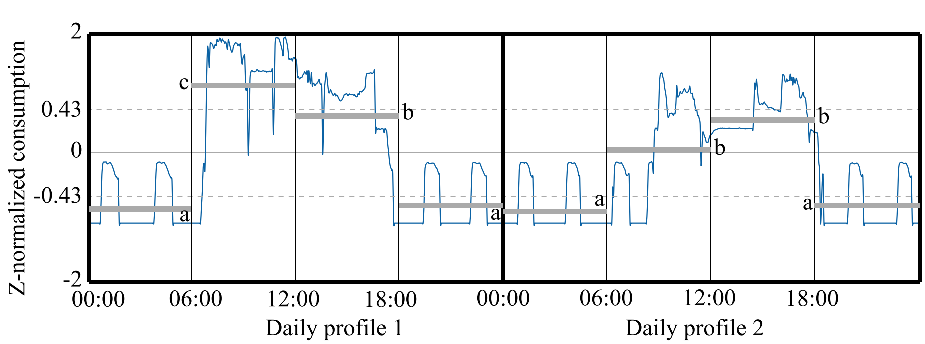

Fig. 3. SAX word creation example (based on figure from Keogh et al. [45]) of two days of 3 minute frequency data, parameters are $N = 480$, $W = 4$, and $A = 3$ and the generated representative word for daily profile 1 is *acba* and daily profile 2 is *abba*.

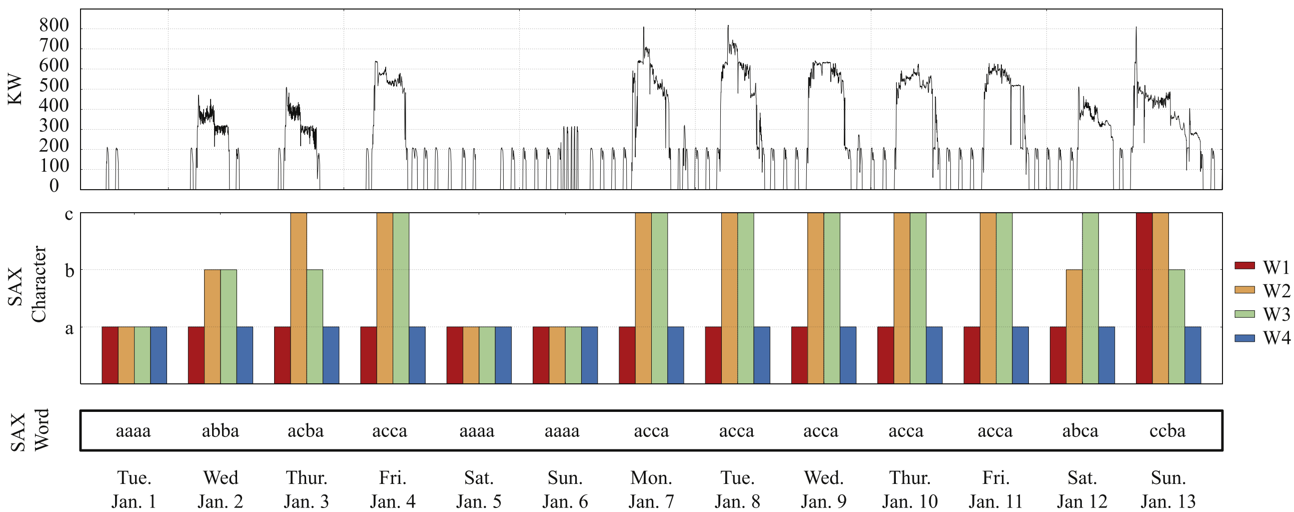

Fig. 4. Creation of SAX words from daily non-overlapping windows: *W*1: 00:00–06:00, *W*2: 06:00–12:00, *W*3: 12:00–18:00, *W*4: 18:00–24:00. Time series data is transformed according to a SAX character creation and then as a string, or SAX word.

a normally expected profile for a typical weekday (*acca*) and weekend (*aaaa*). The less frequent patterns are tagged as discords and can be analyzed in more detail. In this case it can be determined that the patterns *abba*, *abca*, and *acba*, despite being infrequent, are not abnormal due to the occupancy schedule for those particular days. Pattern *ccba*, however, is not explainable within the scheduling and is due to a fault causing excessive consumption in the early morning hours.

This step leads into the next phase of the process focused on further aggregating the motif candidates of the dataset. The size and number of potential motif filtered in this step will give an indication of the number of clusters that will likely pickup meaningful structure from the dataset.

3.4. Clustering

After dividing the profiles into motif and discord candidates, we go on to cluster the motif candidates to create general daily performance phenotypes of the targeted data stream. This step is supplementary if the SAX transformation process produces too many motif candidates based on the input parameter settings. Clustering would be useful, for example, if 15 motif candidates are created and the user wants to further aggregate those candidates into 4 or 5 more *typical* profiles for simulation calibration purposes. This feature gives the user additional control to further aggregate the performance characterization, which can be useful when choosing large values of A or W in the SAX process. It should be noted that in some simplified cases or small datasets, this step may be redundant with SAX aggregation.

We use the k-means algorithm to cluster the daily profiles after removing the discord candidate day-types. This ensures load profile patterns that are not influenced by the less frequent discords. Time series clustering can be approached as a raw-data-based, feature-based, or model-based solution [54]. Numerous clustering techniques have been developed and evaluated for various contexts and optimization goals. The most common implementation is the raw-data-based k-means clustering algorithm and we chose to use it with the Euclidean distance measure due to its simplicity and demonstrated appropriateness for this application [55,56]. The algorithm in our application takes our daily chunks ($N_1, N_2, \ldots, N_n$) and partitions these observations into k sets, $S = \{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squares [57]:

$$argmin \sum_{i=1}^{k} \sum_{N_j \in S_i} \left\| N_j - \mu_i \right\|^2 \tag{2}$$

where μ_i is the mean of the points in S_i.

The disadvantages of the k-means are related to the need to specify the number of clusters and the selection of the initial partition. Both of these faults are not obviously detrimental to this particular application and it is outside the scope of this work to test various clustering algorithms.

Each clustering step includes the calculation of two internal validation metrics which statistically evaluate how well the k-means algorithm was able to create distinct groups of daily profiles, the silhouette coefficient and the sum of square error. The silhouette coefficient score is a measurement of the inter-cluster cohesiveness and intra-cluster separation; a score of 1 is best and −1 is the worst. The coefficient is calculated with the following equation [58]:

$$s = \frac{b - a}{max(a, b)} \tag{3}$$

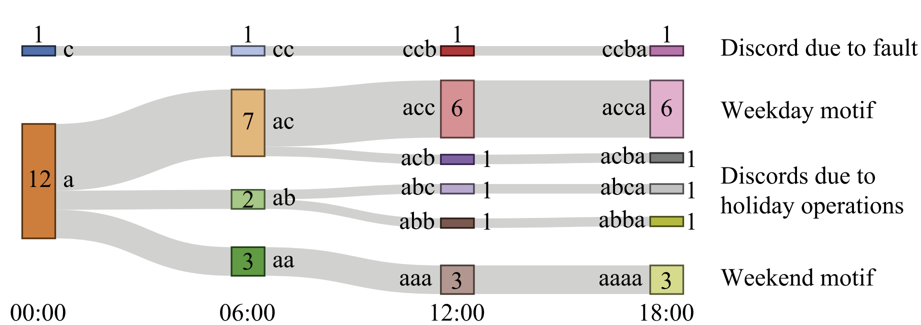

Fig. 5. Augmented suffix tree of SAX words. Each level from left to right represents the *W*1–*W*4, the substrings are noted adjacent to each bar, and the bar thickness is proportional to the number of days within each pattern type. The pattern frequency in number of days is noted in this graphic within or just adjacent to each bar.

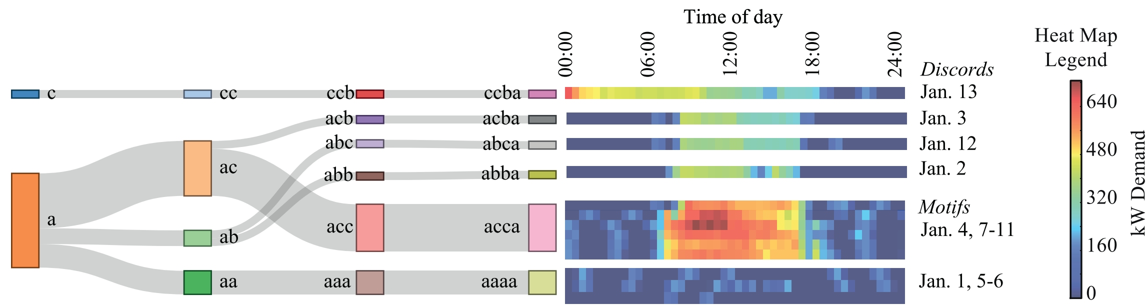

Fig. 6. Example suffix tree with heatmap from the two week dataset. The sankey diagram illustrates the divisions according to pattern and the general categories of motif vs. discord candidates. Each horizontal line in the heatmap represents a single daily profile to illustrate consumption magnitude of each SAX word.

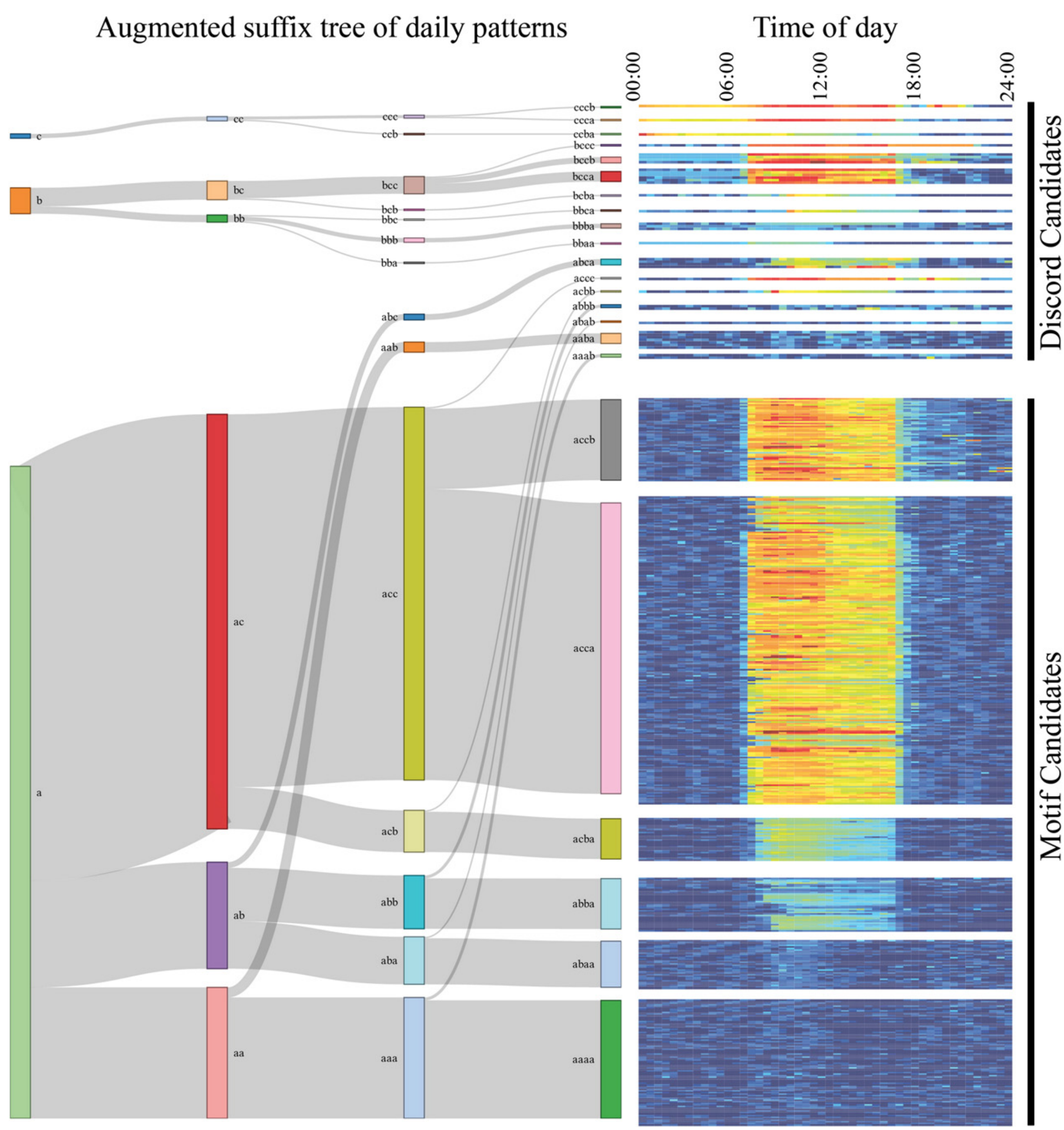

Fig. 7. Case Study 1 cooling electricity consumption representation of the day-types from the DayFilter process.

where a is the mean distance between a sample and all other points in the same cluster and b is the mean distance between a sample and all other points in the next nearest cluster. Minimum sum of square error is another metric that indicates the tightness of clusters with a smaller error being more desirable. Sum of square errors is calculated using the following equation [57]:

$$\mathrm{SSE} = \sum_{i=1}^{k} \sum_{N_j \in S_i} \left\| N_j - \mu_i \right\|^2 \tag{4}$$

where the variables are the same as those from Eq. (2).

3.5. Expressive visualization for interpretation

As the final step, interpretation and visualization are important for *DayFilter* in order for a human analyst to visually extract knowledge from the results, and to make decisions regarding further analysis. We utilize insight from the Overview, zoom and filter, details-on-demand approach [59] and the previously mentioned VizTree tool [47]. The hidden structures of building performance data are revealed through the SAX process and we use visualization to communicate this structure to an analyst. The process uses a modified sankey diagram to visualize the augmented suffix tree in a way which the count frequency of each SAX word can be distinguished. Fig. 6 shows how this visualization is combined with a heatmap of the daily profiles associated with each of the SAX words using the same two-week example data from Figs. 4 and 5. The sankey diagram is rearranged according to the frequency threshold set to distinguish between the motif and discord candidates.

In Fig. 6, the discords are shown as the top four days, Jan. 2, 3, 12, and 13 and the remaining days shown as more frequent potential motifs below. Each daily profile is shown adjacent to the right of the sankey diagram and is expressed as a color-based heatmap. Each horizontal bar of

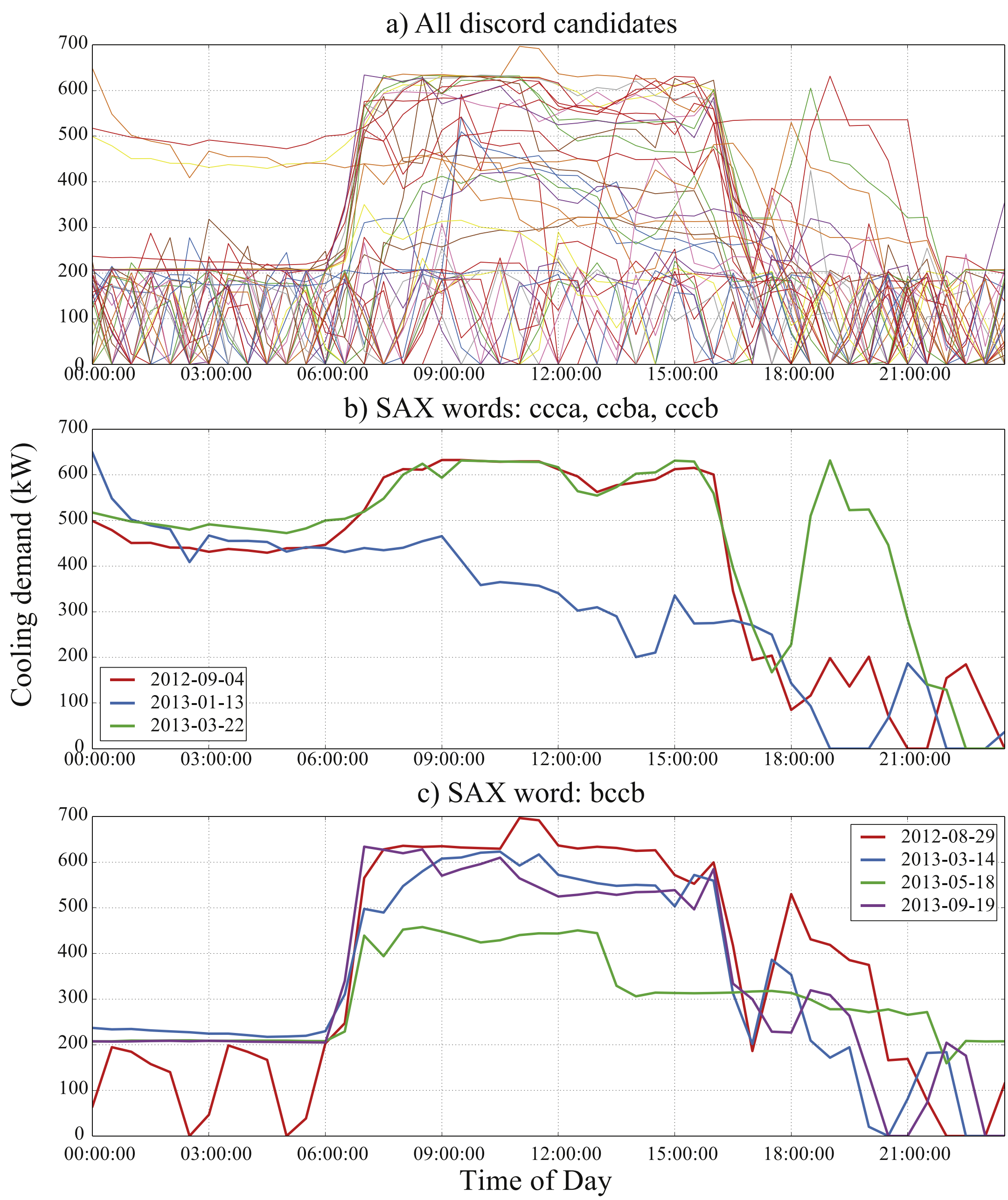

Fig. 8. Daily profiles from the discord filtering process of Case Study 1: a) all of the 17 discord patterns, b) patterns starting with the letter 'c' showing the detected consumption deviation due to air-handling unit fault, and c) 'bccb' pattern showing examples of early evening scheduled activities.

the heatmap is an individual day and they are grouped according to pattern with the associated legend informing the viewer the magnitude of energy consumption across the day. This visualization is designed to quickly present the patterns grouped according to a sort of hierarchy provided by the suffix tree. One can more easily distinguish seemingly *normal* versus *abnormal* behavior with this combination of visualizations.

In the two week example, we have not visualized the clustering step due to the simplicity of such a small dataset. The following case studies illustrate the need for this step through further aggregation of much larger datasets.

4. Application

We now apply *DayFilter* on two large energy performance datasets to demonstrate the usability and results in real-life scenarios. The first application is to the cooling energy consumption of a large campus in which a significant amount of time was spent interpreting and evaluating the motif and discord candidate profiles created. Case Study 1 is completed in cooperation with the facilities and operations staff on the campus and the results are correlated with the manual processes they utilize to find operational issues. The second application is an abbreviated study of a single building in a continental, European climate. Case Study 2 is provided as a contrast of a different climate, building use type, and data stream and less emphasis is placed on investigation of results.

4.1. Case Study 1: cooling energy from tropical climate international school campus

We apply the process to a 70,000 square meter international school campus in the humid, tropical climate of Singapore. It was built in 2010 and includes a building management system (BMS) with over 4000 measured data points taken at 5 min intervals from the years of 2011–2013 — resulting in close to 800 million records of raw data. This collection includes 120 power meters and 100 water meters in the energy and water management system. The data from this study are a seed dataset in an open repository of detailed commercial building datasets [60]. The

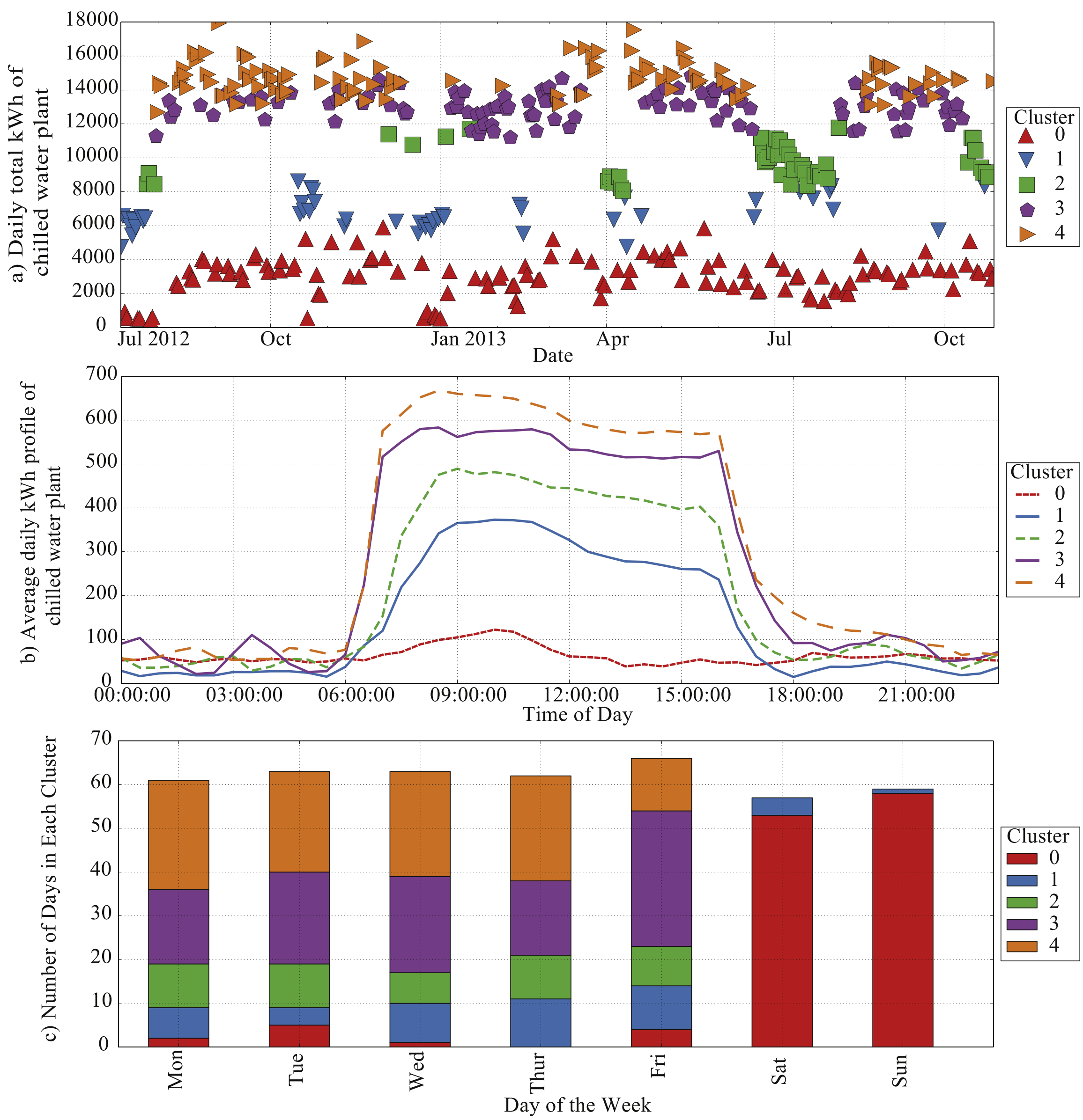

Fig. 9. Case Study 1 cooling electricity consumption results from the automated clustering process: a) Daily clusters, b) average daily profiles, and c) cluster breakdown by day of the week.

primary energy load for this campus is the chilled water plant which consumes almost €40,000/month in electricity.

We focus on the chilled water plant electricity consumption due to its importance in this climate and the potential savings opportunities available through chilled water plant optimization. Measured kilowatt-hour (kWh) and kilowatt (kW) readings were taken from July 12, 2012 to October 29, 2013 with 474 total daily profiles analyzed. Fig. 7 illustrates a sankey diagram with heatmap of the output of the *DayFilter* process with parameters set to $A = 3$ and $W = 4$. The discord and motif candidates are separated in this case according to a decision threshold which quantifies a discord as a day-type with a frequency count less than 2% of total days available. This distinction results in 39 days with patterns tagged as discord candidates, which is 8.2% of the total days in the dataset.

In general, there are six main motif candidates with two candidates appearing to be typical weekday types, two holiday or half-capacity types, and two weekend unoccupied types. Pattern *aaaa* and *abaa* are predominantly flat profiles common to non-occupied cooling consumption. Patterns *abba* and *acba* are representative of days in which school is out of session but the office spaces are still occupied by staff. Pattern *acca* represents a standard full-occupied school day and it is by far the most common with 202 days tagged out of 474. Pattern *accb* is similar to *acca* with slightly more use in the late afternoon and early evening. This phenomenon is due to extracurricular activities planned outside the normal operating schedule of the facility.

There are 17 tagged discord types containing anywhere from one to eight daily profiles. These discord candidate profiles are seen separated in Figs. 7 and 8a. They are filtered out of the dataset and investigated in more detail with the assistance of the facilities maintenance personnel on-site. The three days with patterns starting with the letter *c* are most prominently divergent from the rest of the dataset. These patterns are *cccb*, *ccca*, and *ccba* and they can be seen in Fig. 8b. They represent days in which a significant amount of consumption was measured in the very early morning hours; this situation should not occur according to any schedule. These days corresponded to behavior experienced by the facilities team in which they noticed numerous air handling units (AHUs) would turn on spontaneously in the middle of the night despite no signal from the BMS. The discord analysis was useful for the maintenance staff to observe as they were unsure of just how often the AHUs in the building had been "running wild" at night. This issue was remedied by replacement of certain power meters on the BMS network that were suspected of introducing significant signal noise that caused the AHU problems. The seven patterns starting with the letter *b* are also suspected to have this root cause albeit with less AHU units malfunctioning in that way. An example of this type of pattern is the *bccb*, seen in Fig. 8c, which also shows early evening consumption spikes that are correlated with planned extracurricular activities. The discord patterns starting with *a* have normal early morning cooling energy use but have usage profiles slightly abnormal as compared to the common patterns. Most of these were determined to be scheduling-related with extracurricular activities that are not part of the conventional school schedule.

The next step in the process is to use k-means clustering to further aggregate the motif candidates. We choose to group the day-types into five clusters based on the six motif candidates and the similarity between the *aaaa* and *aaba* patterns. The results of this clustering are seen in Fig. 9. Cluster 0 is strongly prevalent on the weekends, while clusters 2, 3, and 4 are weekday dominant. Cluster 1 correlates with days in which the facility is partially open due to a teacher working day without students or extracurricular activities.

The *DayFilter* process is univariate in this analysis, however the clusters can be visualized as compared to distributions of potential influencing external variables using box plots. Fig. 10 illustrates this visualization. Fig. 10a shows the distributions of daily cooling energy according to cluster, producing intuitive results of consecutively increasing, tight clusters due to utilization of this variable as the clustering target. The discord candidates span most of the cooling consumption range and include quite a few high consumption outliers as compared to the characteristic clusters. Fig. 10b illustrates the chilled water plant efficiency distributions which vary slightly in mean but are quite different in variance and outliers. It is interesting to observe the difference in variance between the clusters, with some exhibiting very small ranges (Cluster 3) and others showing much larger ranges (Cluster 1). This insight could be further investigated with respect to the cooling system control. The discord candidates actually exhibit a tight range of cooling system efficiency. Fig. 10c shows the cluster distribution according to outside ambient dry bulb temperature. This result shows how consistent the chiller plant output is over the operating range with respect to weather conditions.

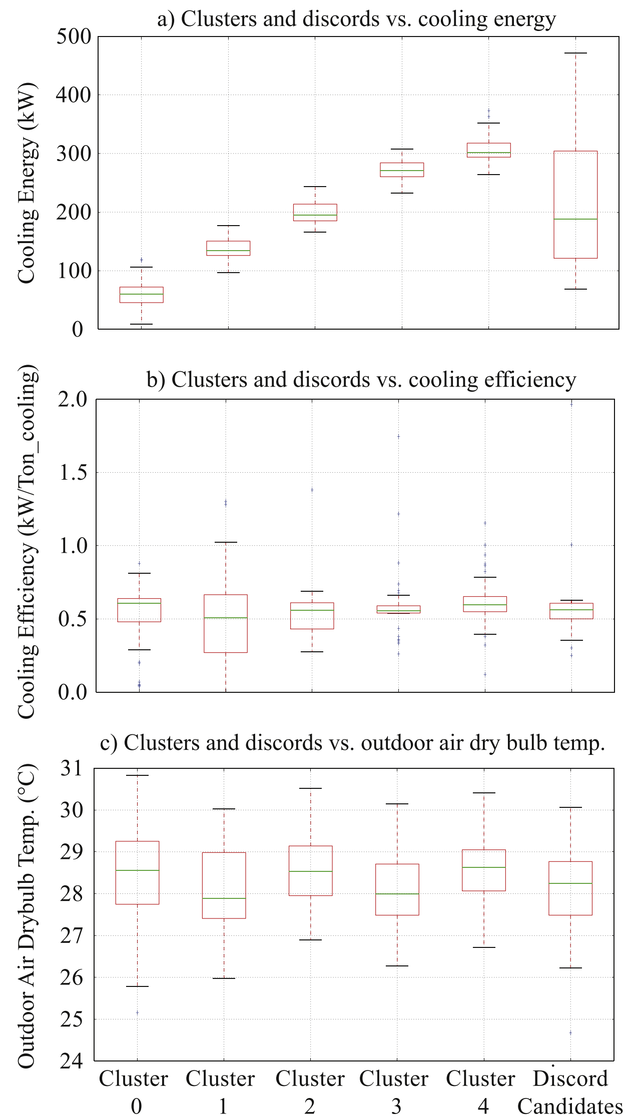

Fig. 10. Box plot distribution comparison of generated clusters: a) clusters as compared to chilled water plant kW, b) comparison to cooling plant efficiency, and c) comparison to outdoor air dry bulb temperature.

4.2. Case Study 2: whole building electricity from European temperate climate office building

The second case study is an office building in a temperate, continental climate in Switzerland. It is a facility with gas-heated hot water and electric chilled water cooling systems. The building is used as a case study facility for innovative building systems. Measurement system data has been collected for over three years from various building submetering systems. This analysis is abbreviated as compared to Case Study 1 as less time was spent on the investigation of the motifs and discords created in this example; it is included to contrast the basic capabilities of *DayFilter* for a dataset in a continental climate with a different type of univariate data stream. For this application, we investigate the electricity profiles for the whole building. Whole building electricity should be more diverse in this example since more subsystems are contributing to the profiles, as compared to only cooling energy in the previous example. A dataset of 9769 hourly data points spanning from July 1, 2012 to August, 12, 2013 are implemented using the *DayFilter* technique on 407 daily profiles. Fig. 11 illustrates the process as applied to this case study with parameters $A = 3$ and $W = 4$. The discord and motif candidates are separated in this case according to a decision threshold which quantifies a discord as a day-type with a frequency count less than 2% of total days available. This distinction results in 53 days with patterns tagged as discord candidates, which is 13% of the total days in the dataset.

Ten motif candidates are created in the SAX process due to the larger diversity of operating profiles because of the distinct heating and cooling seasons that exist in a temperate climate as well as the different operating schedules and system types. Additionally, this analysis is using whole building electricity consumption as opposed to only cooling electricity, thus increasing the potential diversity of load patterns. The most notable difference in the cooling season patterns compared to Case Study 1 is in the presence of many more daily SAX words starting with *b* or *c*. It was discovered that this facility has an ice storage system that is utilizing electricity in the early morning hours to shift the demand during the cooling season. Motif candidate patterns fitting in this category include *cccb*, *caaa*, *bccb*, *bbcb*, *bbbb*, and *baaa*. The remaining motif candidates are more common during the winter with *accb* and *abcb* exemplifying weekday profiles and *abbb* and *aaaa* representing weekends and holidays.

There are 22 discord candidates in this case study with anywhere between 1 and 8 days in each discord. The most obvious anomaly in the discord dataset are the top consuming *cccc*, *ccbb*, *cbcb*, and *cbcc* patterns.

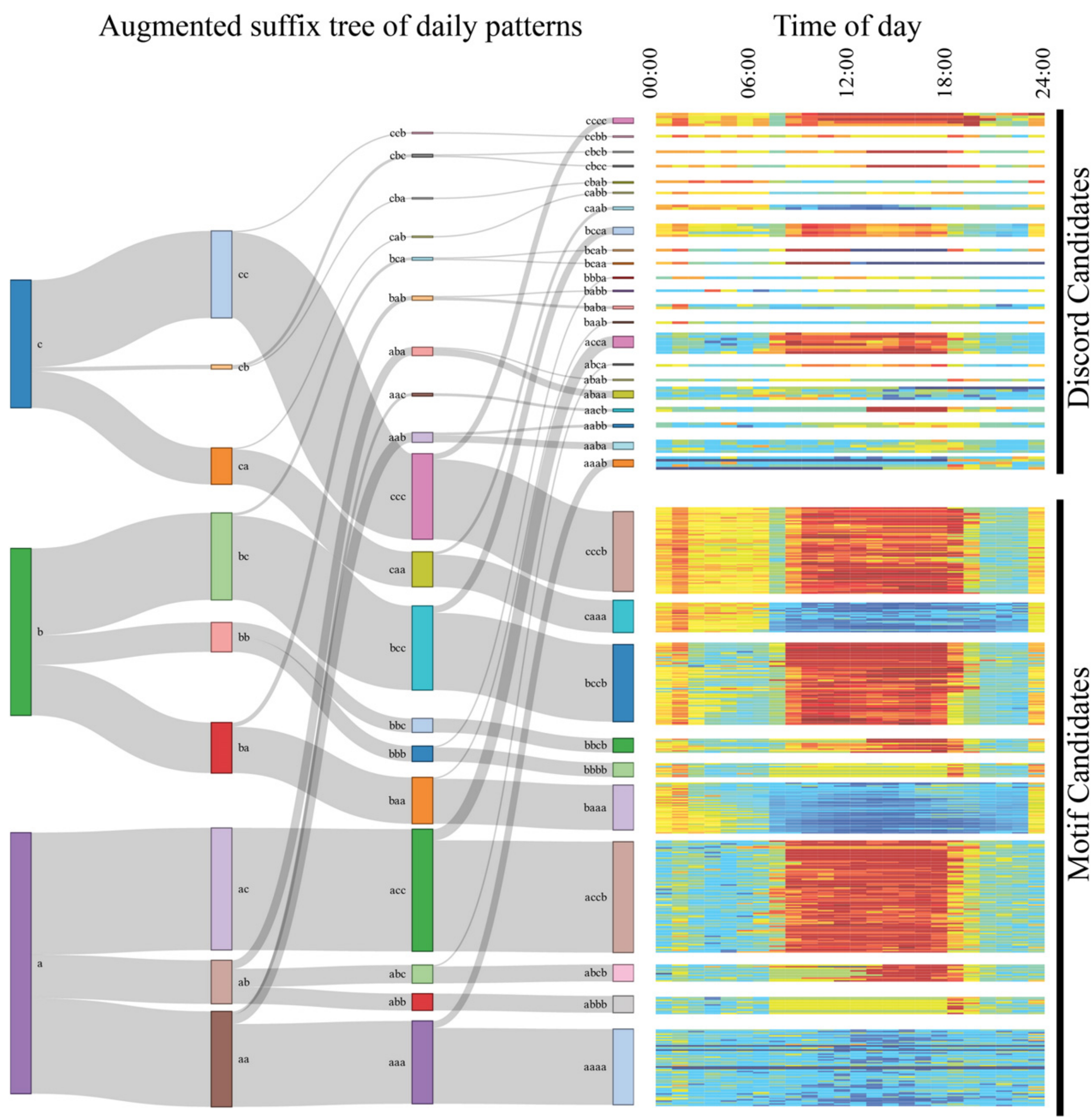

Fig. 11. Case Study 2 whole building electricity consumption representation of the day-types from the DayFilter process.

These discords represent days in the dataset in which the whole building consumption remained high for most hours of the day. Unlike Case Study 1, these discord patterns are not investigated in more detail for this paper. However, the wider range of discords is not surprising due to the increased complexity of the whole building electricity data stream as compared to a subsystem like cooling.

Due to the larger number of SAX word generated motif candidates, this analysis makes better use of the clustering step in order to further aggregate the profiles into a smaller set of performance profiles that can be used in design phase feedback. The clustering step for this project seen in Fig. 12 also reflects the larger diversity of profiles through the choice of using six clusters instead of five. The choice of six clusters was a result of the analysis of the motif patterns, which resulted in the decision to create three heating season clusters and three cooling season clusters.

The *DayFilter* implementation process shows how clustering and visualization techniques can be used to find the structure in building performance data. Results from the two case studies give an indication of daily performance motifs and discords of whole building and cooling energy consumption across the various control-type phases of the building.

5. Parameter selection analysis

An overview of the case studies provides insight into the type of results that *DayFilter* creates. We describe the process as parameter-light as there are only a few input variables that need to be set by the user. The key parameters include the SAX word creation parameters of alphabet size, *A*, number of windows per day, *W*, and the number of clusters chosen in the clustering step, *k*. This section discusses these input parameters related to their influence on the results of the process.

5.1. SAX parameters

The input parameters of *A* and *W* allow the user to tune the granularity of the SAX word creation process. The *DayFilter* process was executed for six total scenarios to test the general impact of these two input parameters. This process was completed on the cooling energy data from

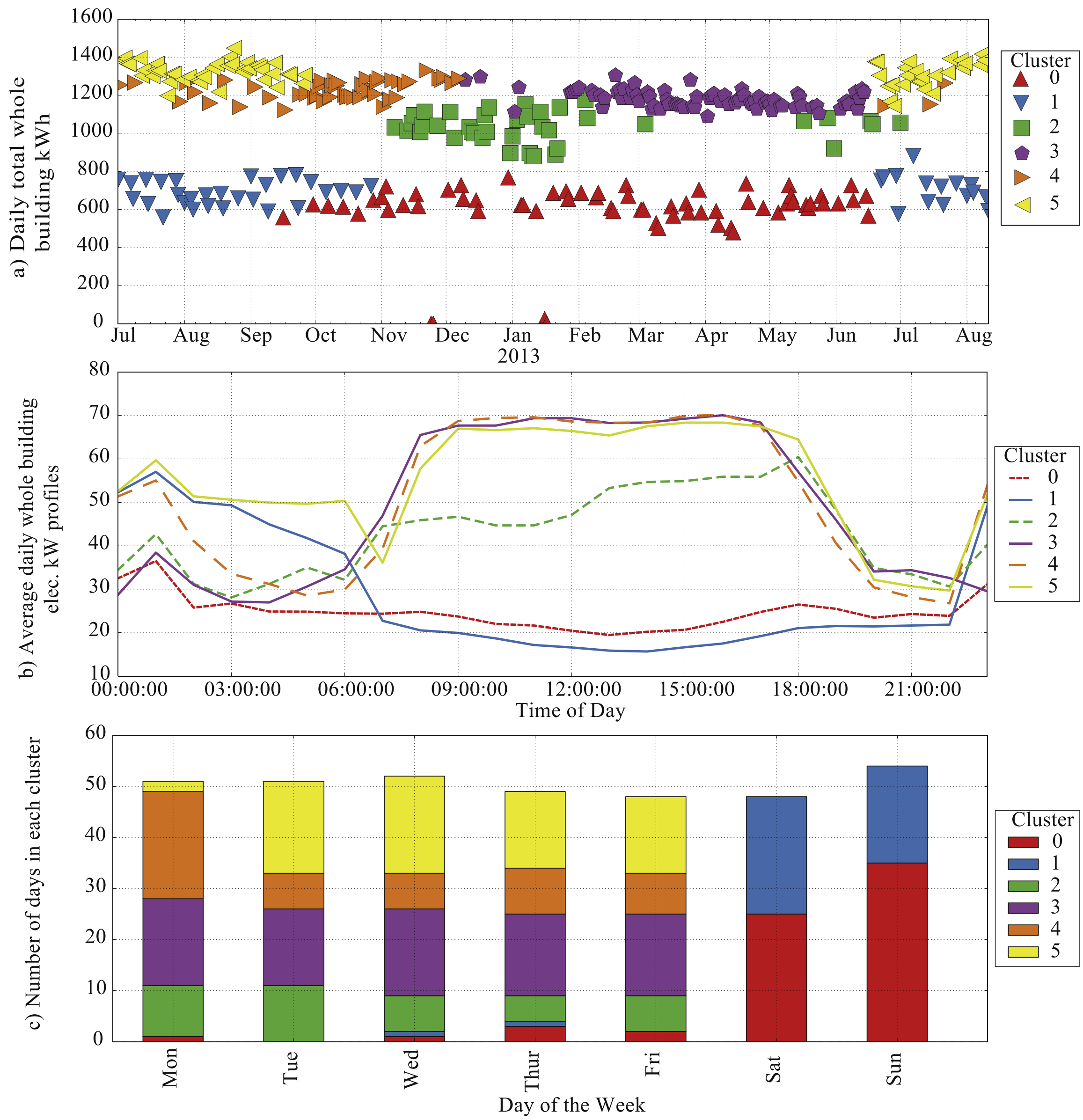

Fig. 12. Case Study 2 whole building electricity consumption results from the automated clustering process: a) Daily clusters, b) average daily profiles, and c) cluster breakdown by day of the week.

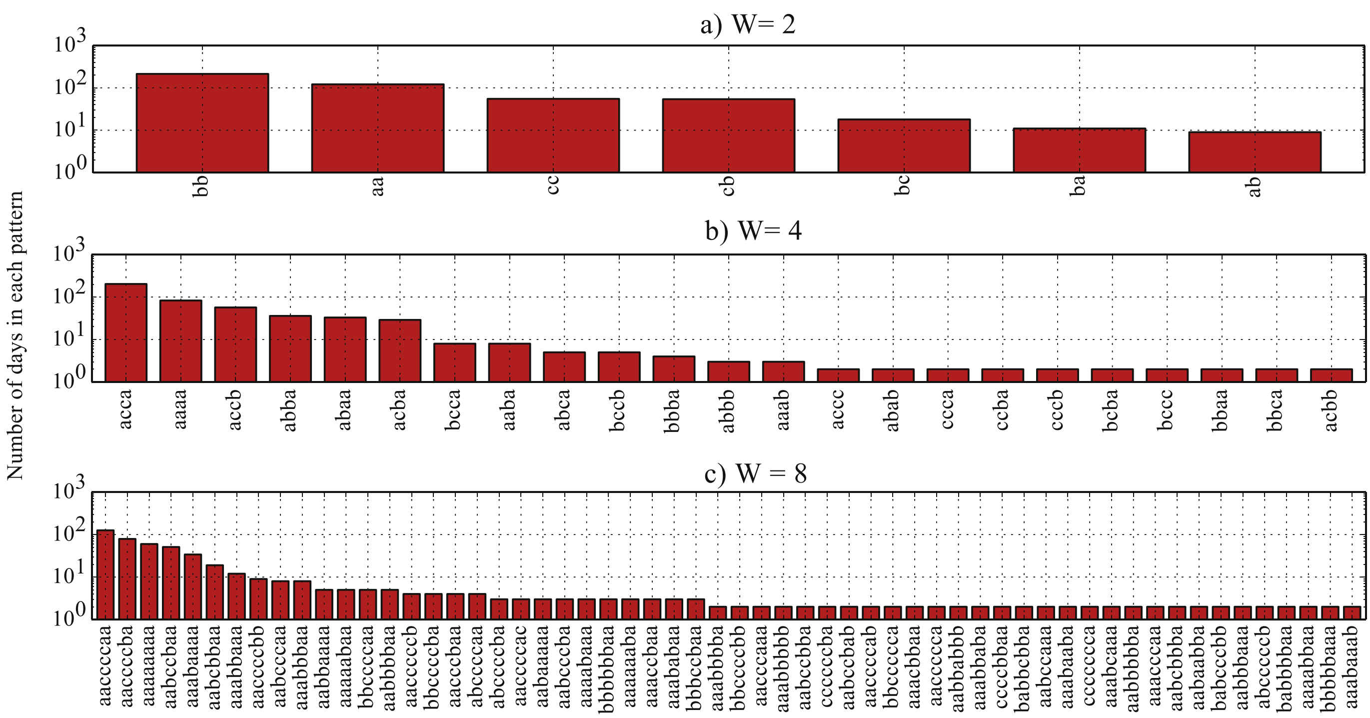

Fig. 13. Comparison of size and number of patterns with different SAX window sizes, a) $W = 2$, b) $W = 4$, c) $W = 8$. Alphabet size, A, is set to 3 in all cases.

Case Study 1. These examples are generated and visualized in this section in order to illustrate the impact on the size and number of generated patterns.

Fig. 13 illustrates the first set of experiments in modulating the window size, W, within the process. Three scenarios are presented in which W is set to 2, 4, and 8 resulting in 12, 6, and 3 hour windows, while all have set the alphabet size, A, to 3. This set of experiments created 7, 23, and 58 patterns respectively. Increasing the number of windows increases the temporal granularity and thus the length of the SAX words and the number of patterns detected.

Fig. 14 shows the SAX words generated by modulating the size of the SAX alphabet, A. Three scenarios are presented in which A is chosen as 2, 3, and 4 letters. This set of experiments created 8, 23, and 39 patterns respectively. Increasing the number of letters increases the granularity of capturing the magnitude and thus the number of patterns detected.

For both the $W = 2$ and $A = 2$ scenarios, the number of patterns is relatively low at 7 and 8 SAX words created. This low number creates a more simplified situation that may be less overwhelming for an analyst to interpret, however, the downside is the low level of detail. The finer granularity scenarios of $W = 8$ and $A = 4$ create the most detailed

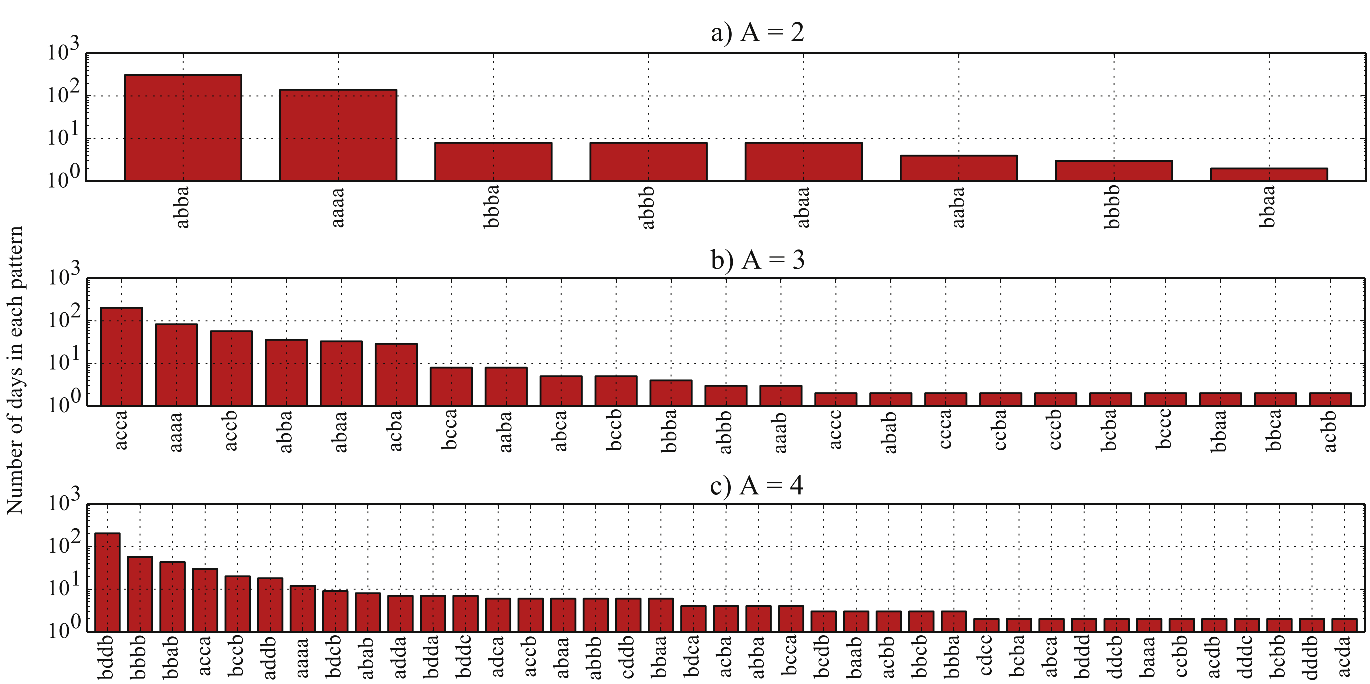

Fig. 14. Comparison of size and number of patterns with different SAX alphabet sizes, a) $A = 2$, b) $A = 3$, and c) $A = 4$. Window count, W, is set to 4 in all cases.

and highest number of patterns, which may be overwhelming in a manual analysis situation.

We investigate these scenarios further by taking the most frequent pattern found for each set of parameter inputs and using a box plot to visualize the statistical distribution of the days contained in each pattern according to different windows within the daily profile. Fig. 15 shows this analysis for the W window count input. A strong difference exists between the variability and, thus, the detail captured by the selection of this input. A relatively high variance is present for $W = 2$ during occupied hours between 06:00 and 18:00. Values of W of 8 and 4 created profiles of more similar distribution with slightly higher variance for the latter. These results reinforce the observation that smaller window sizes create more tightly grouped patterns, albeit with many more variations.

Fig. 16 illustrates the same process with the most frequent patterns from the SAX alphabet size, A, experiments. The alphabet size decision modulates the magnitude resolution available to the algorithm. Thus, it is intuitive that the pattern has less variance as the value of A increases. This observation is also apparent in this analysis where an alphabet size of 4 produces tighter distributions than a value of 2, in this case primarily during occupied hours as well.

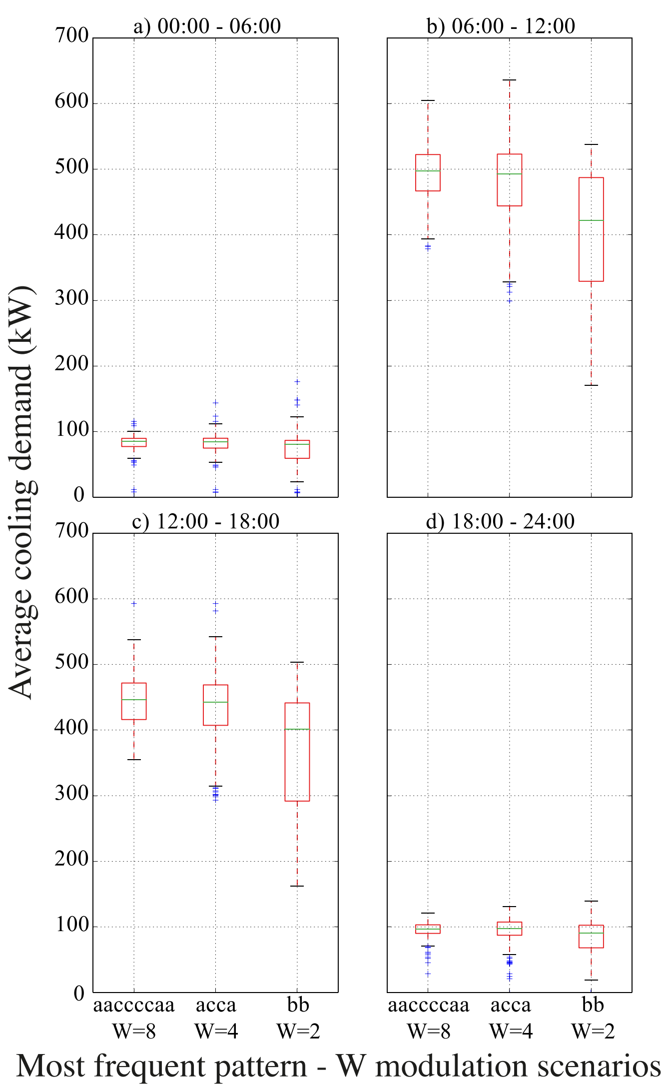

Fig. 15. Comparison of statistical variance of window count, W, parameter choice for average daily time spans of a) morning unoccupied, b) morning occupied, c) afternoon occupied, and d) night unoccupied.

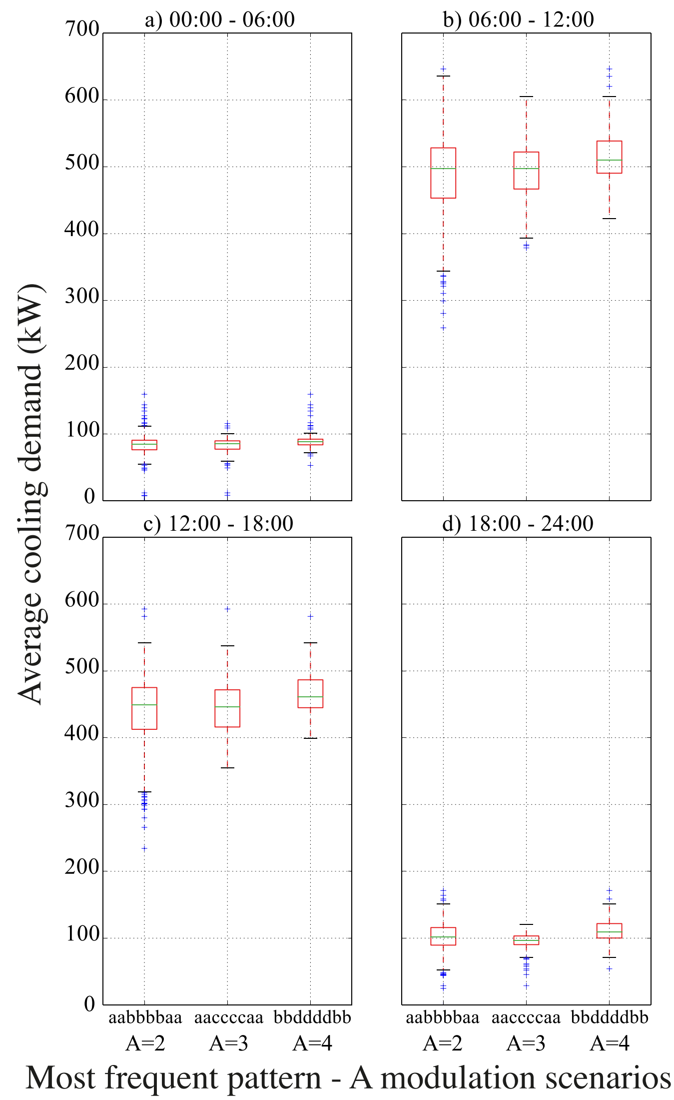

Fig. 16. Comparison of statistical variance of SAX alphabet size, A, parameter choice for average daily time spans of a) morning unoccupied, b) morning occupied, c) afternoon occupied, and d) night unoccupied.

Based on these observations of the A and W parameters, we found that setting $A = 3$ and $W = 4$ resulted in the best balance between number of patterns generated and resolution of detail needed to adequately filter discords in a 24 hour period. While these findings are specific to our case studies, we hypothesize that similar settings will be useful when analyzing other building performance data due to the generally reoccurring daily patterns. These initial parameter settings may be used as a default when implementing the *DayFilter* process and adjusted accordingly based on visualizations similar to those developed in this section.

5.2. Clustering parameters

The number of clusters to be created in the clustering step is based primarily on the interpretation of the motif candidates by an analyst. However, to understand the quality of clustering from a statistical standpoint, we executed the clustering step on both case studies with quantities, k, between 2 and 11 clusters. Eqs. (3) and (4) are used to calculate the silhouette score and sum of square error for each scenario. Fig. 17 illustrates the two quantitative clustering metrics calculated for a range of cluster number options for the two case studies. The results

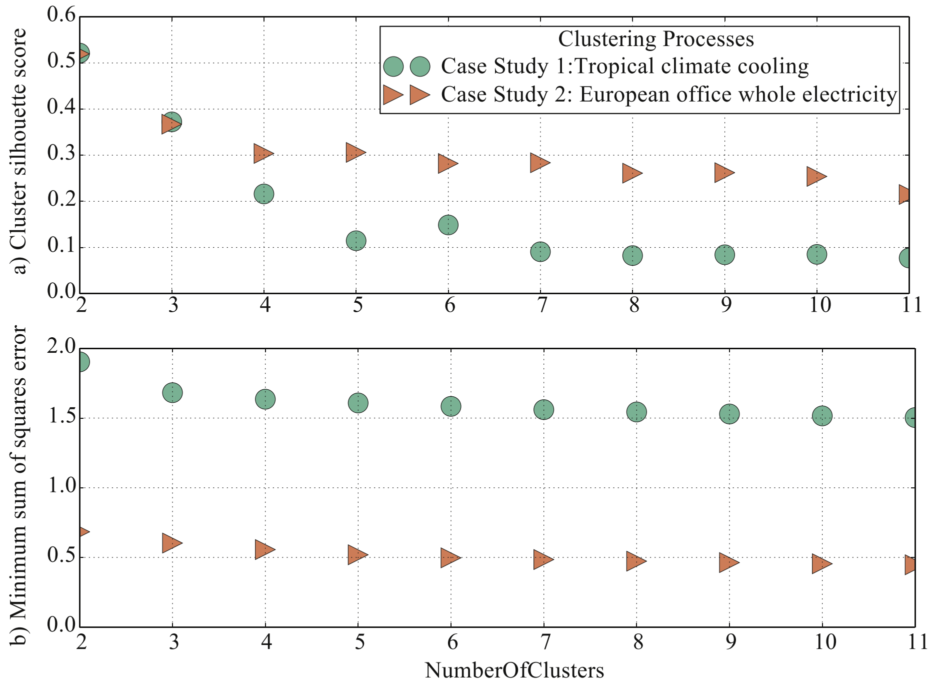

Fig. 17. Clustering validation metrics for the case studies: a) Silhouette score and b) minimum sum of square error.

of these quantitative metrics are relatively flat after 4 clusters indicating that there isn't a significant difference between 4 and 11 clusters based on the metrics. Cluster sizes of 2 and 3 performed better in the Silhouette score, however these number of clusters would usually aggregate the data too much to properly capture the important structure. The minimum sum of square metric remains consistent through the experiments.

6. Discussion

The intent of the *DayFilter* process is to provide building analysts a means of quickly filtering diurnal patterns in time series building performance data. The objective is to provide a link between top-down and bottom-up analysis approaches, specifically AFDD and model calibration. We discuss the results of the application with consideration to these two contexts including limitations encountered. Additionally, we address the topic of data quality and its importance in this process.

For the implementation of AFDD, *DayFilter* provides a means of investigating a dataset at the early state of implementation. For example, qualitative, rule-based approaches benefit specifically as they require the setting of threshold parameters based on expert analysis. Despite their lack of sophistication, this category of AFDD is quite popular in implementation and many of the tools in the literature are based on this paradigm [8]. This fact is due to the capability of many building management systems to incorporate these techniques easily. In Case Study 1, the building operations staff was able to do a deeper analysis of the AHU tripping fault discords to set an alarm which would notify staff if the phenomenon occurred again. This type of rule-creation could occur automatically based on the outputs from the process.

One limitation in using this process for AFDD is the observation that anomalous behavior won't always manifest itself as a discord. More systematic or gradual failures could blend into the frequent motifs or eventually create patterns frequent enough to form their own motif. We found that a key feature in this consideration was the ability to modify temporal granularity (through number of windows per day, W) or the magnitude or shape difference granularity (through alphabet size, A) to increase or decrease the proportion days tagged as discords as compared to motifs. The number of patterns created when selecting smaller windows or larger alphabets greatly increases, which can be seen as a disadvantage in terms of interpretability. The trade-off is that these settings give more resolution to the process, thus creating more tightly grouped clusters and more effectively detecting discords. Modulating this feature enables focus on coarse high-level patterns or more sub-hourly phenomenon. We have tested these parameters according to the two case studies and provided suggestions for initial settings as applied to energy data in these contexts. However, a user of the process may decide to tune these parameters to further investigate a dataset for the implementation of certain AFDD approaches.

For simulation model calibration, the *DayFilter* approach provides a few improvements in occupancy pattern and diversity factor detection as compared to conventional day-typing techniques [27,25]. It was observed in the case studies that the process was able to effectively distinguish patterns between cooling and heating seasons, occupied and unoccupied phenomenon according to operation schedules and other such profiles. This differentiation was done in an automated way and is an improvement as compared to the day-typing techniques reviewed in the literature. The influence of anomalous days is not averaged into the patterns and thus a better representation of typical performance can be achieved. The output of the clustering step within the process can be used as an input to diversity factor calculations outlined in the literature.

A limitation with respect to day-typing for calibration is that we don't fully extract the diversity profiles as part of the process. Further refinement of the outputs using conventional techniques from the literature needs to be implemented to achieve typical profiles for model calibration purposes.

As with any data-driven approach, the usefulness of the process is only as good as the quality and amount of data available. Statistical approaches do not rely on physical knowledge of the building systems and are thus prone to error with inaccurate data. We prequalified the case studies in this paper based on how rigorous the data quality process for the sensor networks already is. The data from Case Study 1 was from a centralized chilled water plant which had undergone extensive accuracy measures including a heat balance test and third party sensor accuracy verification. These checks are newly mandated by the local jurisdiction and are part of a steadily growing improvement in the quality and cost of sensor networks. Case Study 2 also had a modern, well-calibrated sensor network and data acquisition system.

7. Conclusion and outlook

We have described and shown applications of the *DayFilter* process, a set of temporal data-mining techniques that can find characteristic performance and can help pinpoint infrequent behavior for further evaluation. The process is designed as a forensic filter to focus the effort of an analyst on particular data subsets. The process is applied to two case studies and the results confirm the ability of the process to find various types of diurnal patterns amongst large univariate datasets. Discord filtering for Case Study 1 found anomalous daily profiles that are consistent with cooling system faults observed on site. The motif and clustering aggregation process for both case studies produce profiles that may be used to calibrate a simulation model from the design phase or to inform future designs using empirical data. This process benefits creation of occupancy, lighting, and plug load schedules for whole building simulation input.

The process as defined in this work is univariate. The next question is whether *DayFilter* can be utilized to extract patterns across multiple data streams. This challenge is approached by performing the univariate process for all available data points, including the detailed data downstream from the metrics used in this study. A multivariate analysis of the overlapping discords could be used to further automate the process of AFDD implementation. Another enhancement could be the use of higher granularity parameter settings for finding patterns in sub-hourly measured data, as opposed to the daily profiles used here.

Acknowledgments

The authors would like to acknowledge United World College of South East Asia (UWCSEA) and Siemens Building Technologies for providing the data used in this paper. This research was funded through an Institute of Technology in Architecture (ITA) Fellowship at the ETH Zürich.

References

[1] A. Khan, K. Hornbæk, Big data from the built environment, Proceedings of the 2nd International Workshop on Research in the Large (LARGE '11), ACM, Beijing, China, 2011, pp. 29–32.
[2] Carbon Trust, Closing the Gap: Lessons Learned on Realising the Potential of Low Carbon Building Design, Carbon Trust, London, 2011.
[3] P. de Wilde, The gap between predicted and measured energy performance in buildings: a framework for investigation, Autom. Constr. 41 (C) (2013) 1–10.
[4] A.C. Menezes, A. Cripps, D. Bouchlaghem, R.A. Buswell, Predicted vs. actual energy performance of non-domestic buildings: using post-occupancy evaluation data to reduce the performance gap, Appl. Energy 97 (2012) 355–364.
[5] C. Demanuele, T. Tweddell, M. Davies, Bridging the gap between predicted and actual energy performance in schools, World Renewable Energy Congress XI, UAE, Abu Dhabi, 2010, pp. 25–30.
[6] C. Turner, M. Frankel, Energy performance of LEED for new construction buildings, Tech. rep., 2008.
[7] J.E. Seem, Using intelligent data analysis to detect abnormal energy consumption in buildings, Energy Build. 39 (1) (2006) 52–58.
[8] J. Ulickey, T. Fackler, E. Koeppel, J. Soper, Building performance tracking in large commercial buildings: tools and strategies — subtast 4.3 characterization of fault detection and diagnostic (FDD) and advanced energy information system (EIS) tools, Tech. rep., Sep. 2010.
[9] H. Friedman, E. Crowe, E. Sibley, M. Effinger, The Building Performance Tracking Handbook, California Energy Commission, Portland Energy Conservation, Inc., 2011.
[10] S. Katipamula, M. Brambley, Methods for fault detection, diagnostics, and prognostics for building systems—a review, part I, HVAC&R Res. 11 (1) (2005) 3–25.
[11] D. Coakley, P. Raftery, M.M. Keane, A review of methods to match building energy simulation models to measured data, Renew. Sust. Energ. Rev. 37 (c) (2014) 123–141.
[12] M. Effinger, H. Friedman, D. Moser, Building performance tracking in large commercial buildings: tools and strategies — subtask 4.2 research report: investigate energy performance tracking strategies in the market, Tech. rep., Sep. 2010.
[13] E. Greensfelder, H. Friedman, E. Crowe, Building performance tracking in large commercial buildings: tools and strategies — subtask 4.4 research report: characterization of building performance metrics tracking methodologies, Tech. rep., Nov. 2010.
[14] C. Miller, A. Schlueter, Applicability of lean production principles to performance analysis across the life cycle phases of buildings, Proceedings of the 8th International Conference on Indoor Air Quality, Ventilation and Energy Conservation in Buildings (CLIMA 2013), Prague, Czech Republic, 2013, pp. 1–10.
[15] E.J. Keogh, S. Lonardi, C.A. Ratanamahatana, Towards parameter-free data mining, Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04), ACM, Seattle, WA, USA, 2004, pp. 206–215.
[16] J. Granderson, M.A. Piette, G. Ghatikar, Building energy information systems: user case studies, Energy Efficiency 4 (1) (2010) 17–30.
[17] T.M. Lawrence, R.T. Watson, M.-C. Boudreau, K. Johnsen, J. Perry, L. Ding, A new paradigm for the design and management of building systems, Energy Build. 51 (2012) 56–63.
[18] Z. O'Neill, X. Pang, M. Shashanka, P. Haves, T. Bailey, Model-based real-time whole building energy performance monitoring and diagnostics, J. Build. Perform. Simul. 7 (2) (2014) 83–99.
[19] A. Costa, M.M. Keane, J.I. Torrens, E. Corry, Building operation and energy performance: monitoring, analysis and optimisation toolkit, Appl. Energy 101 (C) (2013) 310–316.
[20] X. Zhao, M. Yang, H. Li, Field implementation and evaluation of a decoupling-based fault detection and diagnostic method for chillers, Energy Build. 72 (2014) 419–430.
[21] T.A. Reddy, I. Maor, S. Jian, C. Panjapornpon, Procedures for reconciling computer-calculated results with measured energy data, Tech. Rep. Research Project 1051-RP, Jan. 2006.
[22] T. Maile, V. Bazjanac, M. Fischer, A method to compare simulated and measured data to assess building energy performance, Build. Environ. 56 (2012) 241–251.
[23] D.L. Hadley, Daily variations in HVAC system electrical energy consumption in response to different weather conditions, Energy Build. 19 (3) (1993) 235–247.
[24] M. Kaplan, J. McFerran, J. Jansen, R. Pratt, Reconciliation of a DOE2.1C model with monitored end-use data for a small office building, ASHRAE Trans. 93 (1) (1990) 981–993.
[25] C. Duarte, K. Van Den Wymelenberg, C. Rieger, Revealing occupancy patterns in an office building through the use of occupancy sensor data, Energy Build. 67 (2013) 587–595.
[26] J.A. Davis III, D.W. Nutter, Occupancy diversity factors for common university building types, Energy Build. 42 (9) (2010) 1543–1551.
[27] B. Abushakra, A. Sreshthaputra, J.S. Haberl, D. Claridge, Compilation of diversity factors and schedules for energy and cooling load calculations, ASHRAE Research Project 1093-RP, Final Report, Tech. Rep., 2001.
[28] R.H. Dodier, J.F. Kreider, Detecting whole building energy problems, ASHRAE Trans. (1999).
[29] J.E. Seem, Pattern recognition algorithm for determining days of the week with similar energy consumption profiles, Energy Build. 37 (2) (2005) 127–139.
[30] I.P. Panapakidis, T.A. Papadopoulos, G.C. Christoforidis, G.K. Papagiannis, Pattern recognition algorithms for electricity load curve analysis of buildings, Energy Build. 73 (2014) 137–145.
[31] F. Xiao, C. Fan, Data mining in building automation system for improving building operational performance, Energy Build. 75 (2014) 109–118.
[32] W. Foslien, V. Guralnik, S. Harp, W. Koran, Application of clustering techniques to energy data to enhance analysts' productivity, American Council for an Energy Efficient Economy, American Council for an Energy-Efficient Economy, 2002, p. 97.
[33] A. Kusiak, Z. Song, Clustering-based performance optimization of the boiler turbine system, IEEE Trans. Energy Convers. 23 (2) (2008) 651–658.
[34] A.C. Bogen, M. Rashid, E.W. East, J. Ross, Evaluating a data clustering approach for life-cycle facility control, J. Inf. Technol. Constr. (2013).
[35] X. Li, C.P. Bowers, T. Schnier, Classification of energy consumption in buildings with outlier detection, IEEE Trans. Ind. Electron. 57 (11) (2010) 3639–3644.
[36] D. Jacob, S. Dietz, S. Komhard, C. Neumann, S. Herkel, Black-box models for fault detection and performance monitoring of buildings, J. Build. Perform. Simul. 3 (1) (2010) 53–62.
[37] O. Linda, D. Wijayasekara, M. Manic, C. Rieger, Computational intelligence based anomaly detection for building energy management systems, Proceedings of the 5th International Symposium on Resilient Control Systems (ISRCS), Salt Lake City, UT, USA, 2012, pp. 77–82.
[38] J. Ploennigs, B. Chen, A. Schumann, N. Brady, Exploiting generalized additive models for diagnosing abnormal energy use in buildings, Proceedings of the 5th ACM Workshop on Embedded Systems for Energy-Efficient Buildings (Buildsys '13), Roma, Italy, 2013.
[39] Z.J. 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.
[40] A.R. Florita, L.J. Brackney, T.P. Otanicar, J. Robertson, Classification of commercial building electrical demand profiles for energy storage applications, J. Solar Energy Eng. Trans. ASME 135 (3) (2013) 031020.
[41] Z.J. Yu, B.C.M. Fung, F. Haghighat, Extracting knowledge from building-related data — a data mining framework, Build. Simul. 6 (2) (2013) 207–222.
[42] R. Fontugne, J. Ortiz, N. Tremblay, P. Borgnat, P. Flandrin, K. Fukuda, D. Culler, H. Esaki, Strip, bind, and search: a method for identifying abnormal energy consumption in buildings, Proceedings of the 12th ACM/IEEE Int Conference on Information Processing in Sensor Network (ISPN 2013), ACM, Philadelphia, PA, USA, 2013, pp. 129–140.
[43] K. Yoshida, M. Inui, T. Yairi, K. Machida, M. Shioya, Y. Masukawa, Identification of causal variables for building energy fault detection by semi-supervised LDA and decision boundary analysis, Data Mining Workshops, 2008. ICDMW '08. IEEE International Conference on, Pisa, Italy, 2008, pp. 164–173.

[44] P. Patel, E. Keogh, J. Lin, S. Lonardi, Mining motifs in massive time series databases, Proceedings of 2002 IEEE International Conference on Data Mining (ICDM 2002), Maebashi City, Japan, 2002, pp. 370–377.

[45] E.J. Keogh, J. Lin, A. Fu, Hot sax: efficiently finding the most unusual time series subsequence, Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM'05), IEEE, Houston, TX, USA, 2005.

[46] J. Lin, E.J. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD '03), ACM, San Diego, CA, USA, 2003, pp. 2–11.

[47] J. Lin, E.J. Keogh, S. Lonardi, J.P. Lankford, D.M. Nystrom, Visually mining and monitoring massive time series, Proceedings of the 10th ACM SIGKDD International conference on Knowledge Discovery and Data Mining (KDD '04), ACM, Seattle, WA, USA, 2004, pp. 460–469.

[48] J. Lin, E.J. Keogh, S. Lonardi, Visualizing and discovering non-trivial patterns in large time series databases, Inf. Vis. 4 (2) (2005) 61–82.

[49] D. Patnaik, M. Marwah, R. Sharma, N. Ramakrishnan, Sustainable operation and management of data center chillers using temporal data mining, Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09), ACM, Paris, France, 2009, pp. 1305–1314.

[50] M.C. Hao, M. Marwah, H. Janetzko, U. Dayal, D.A. Keim, D. Patnaik, N. Ramakrishnan, R.K. Sharma, Visual exploration of frequent patterns in multivariate time series, Inf. Vis. 11 (1) (2012) 71–83.

[51] D.Q. Goldin, P.C. Kanellakis, On similarity queries for time-series data: constraint specification and implementation, Principles and Practice of Constraint Programming (CP '95), Springer, 1995, pp. 137–153.

[52] J. Lin, E.J. Keogh, L. Wei, S. Lonardi, Experiencing SAX: a novel symbolic representation of time series, Data Min. Knowl. Disc. 15 (2) (2007) 107–144.

[53] P. Weiner, Linear pattern matching algorithms, Switching and Automata Theory, 1973. SWAT '08. IEEE Conference Record of 14th Annual Symposium on, Iowa, USA, 1973, pp. 1–11.

[54] T. Warren Liao, Clustering of time series data—a survey, Pattern Recogn. 38 (11) (2005) 1857–1874.

[55] F. Iglesias, W. Kastner, Analysis of similarity measures in times series clustering for the discovery of building energy patterns, Energies 6 (2) (2013) 579–597.

[56] J. MacQueen, Some methods for classification and analysis of multivariate observations, Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, California, USA, Berkeley, CA, 1967, p. 14.

[57] L. Rokach, O. Maimon, Clustering methods, Data mining and Knowledge Discovery Handbook, Springer, 2005.

[58] P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.

[59] B. Shneiderman, The eyes have it: a task by data type taxonomy for information visualizations, Proceedings of the IEEE Symposium on Visual Languages, Boulder, CO, USA, 1996, pp. 336–343.

[60] C. Miller, Z. Nagy, A. Schlueter, A seed dataset for a public, temporal data repository for energy informatics research on commercial building performance, In proceeding of: 3rd Conf. on Future Energy Business & Energy Informatics, Rotterdam, Netherlands, 2014, pp. 1–2.