Anomalous Thermal Behavior Detection in Data Centers using Hierarchical PCA

Manish Marwah, Ratnesh Sharma, Wilfredo Lugo, Lola Bautista
Hewlett-Packard Co. Laboratories
1501 Page Mill Road, Palo Alto, CA
{manish.marwah, ratnesh.sharma, lola.bautista, wilfredo.lugo}@hp.com

ABSTRACT

In recent years, there has been a significant growth in number, size and power densities of data centers. A significant part of a data center power consumption is attributed to the cooling infrastructure, such as air handling units and chillers. For energy efficient operation and management of the cooling infrastructure, data centers are beginning to be extensively instrumented with temperature sensors. However, it is virtually impossible to manually inspect and analyze the large volumes of dynamic data generated by these sensors for presence of anomalous behavior. Furthermore, threshold-based methods are useful but limited in the kind of anomalies that can be detected. Thus, in order to improve energy efficiency of data centers, there is a need for real-time detection of thermal anomalies such that corrective measures can be promptly taken to remove the inefficiencies and save power.

In this paper, we propose a hierarchical principal component analysis (PCA) based methodology for detection of anomalous thermal behavior, and demonstrate it on a large temperature sensor network in a production data center. Specifically, the technique is applied to thermal anomalies that result from inefficiencies in the airflow pattern in a part of a data center and normally go undetected since no thresholds are violated. The hierarchical analysis performed on the temperature sensor data streams also identifies the location and scope of such anomalous behavior. A prototype of this technique has been implemented and applied to a temperature sensor network spanning 75 racks with 10 sensors each for over a period of 30 days. The results – accuracy: 98.0%, sensitivity: 87.1%, and specificity: 98.8% – demonstrate the effectiveness of our methodology in real-time detection of anomalous thermal behavior in data centers.

1. INTRODUCTION

In recent years, the demand for data centers has seen tremendous growth. Many of the largest data centers in the US are experiencing a growth of 20% per year and over 40% of enterprises are refurbishing or building new data centers to support ongoing business operations and future demand [12]. However, energy consumption of data centers is a concern. The Environmental Protection Agency (EPA) calculates that in 2006, 61 billion kilowatt-hour (kWh) was consumed by data centers in the US. This amount accounts for 1.5% of the total electricity consumed costing $4.5 billion [1]. Moreover, the cooling infrastructure can be responsible for up to 50% of that consumption [7]. It is estimated that data center power consumption will increase 4% to 8% annually and is expected to reach 100 billion kWh by 2011 [13].

Given these trends, monitoring thermal conditions in data centers and responding rapidly to anomalies assumes great significance and can help save energy and operational costs. Until recently data centers were a black box with minimal instrumentation in the way of thermal sensing. After their initial design (where cooling infrastructure was typically overprovisioned, thus leading to higher operational energy costs), there was not much monitoring, only required maintenance after a failure occurred. However, the state-of-the-art data centers today are extensively instrumented and closely monitored. Indeed, a large data center can easily contain tens of thousands of sensors which produce a continuous stream of data. Although these sensors produce a wealth of information on the state of a data center, using this information effectively is a challenge. To detect an anomaly, an administrator must correlate observed measurements to anomalous behavior based on past experience. In addition to very specific domain knowledge required, just the volume of data can be prohibitive to examine manually. The current industry trend is towards a lights out data center that is managed remotely with no manual intervention required.

The monitoring techniques currently deployed in data centers are typically threshold based, that is, they alarm when an administrator configured threshold is crossed. These, however, do not always work well and important anomalies are missed since many do not manifest as threshold violations. Also, early detection of anomalies, which allow preemptive measures to be taken, is difficult using only threshold techniques.

Furthermore, when an anomalous sensor reading is observed, current monitoring systems raise alarms requiring investigation by an administrator. It is nontrivial to determine if the cause of the anomaly is local or related to a larger, facility wide outage. For example, a high temperature sensor reading could be caused by any of the following: 1) a faulty sensor; 2) a rack level anomaly e.g. obstruction of a cool air vent near a rack; or, 3) a failed computer room air-conditioning (CRAC) unit affecting a significant portion of a data center. Automated mechanisms to determine which of the above has occurred is challenging.

The observations made above necessitate automated, timely and specific anomaly detection using the available sensor data streams. In this paper, we propose a hierarchical, principal component analysis (PCA) based technique for automated monitoring of correlations between sensor measurements within a data center. Any change in the correlations signals anomalous behavior. Correlations across several hierarchical groupings are analyzed to determine the extent of an anomaly. Furthermore, the sensors most likely responsible for the anomalous behavior are identified.

We conducted performance evaluation of our methodology at a large, heterogeneous, state-of-the-art production data center. For efficient monitoring and control, this facility has an extensive infrastructure of sensors. The results show that we can detect anomalies at rack, zone and data center region levels. For rack level analysis, the results show an accuracy, sensitivity and specificity of 97.96%, 87.1% and 98.76%, respectively. Threshold based methods are unable to detect most of these anomalies.

Specifically, in this paper, we make three key contributions.

1. We present a scalable, hierarchical PCA-based data mining methodology that can be applied to a large data center sensor network.
2. We introduce a mechanism that allows PCA hidden variables associated with short-lived and insignificant trends to be ignored.
3. We demonstrate the effectiveness of our technique by analyzing sensor data from around 375 temperature sensors for a period of 30 days in a real life production data center.

The rest of the paper is organized as follows. In the next section, we discuss related work. In section 3, we discuss the hierarchical anomaly detection methodology. The layout and architecture of the data center where we demonstrate our techniques is described in section 4. The results are presented in section 5. Finally, we conclude in section 6.

2. RELATED WORK

Considering the huge potential for cost and energy savings, mining of streams of environmental data in data centers has recently received attention. Additionally, local temperature sensing within a data center for better thermal management is becoming important [6]. In the past, exploratory data analysis techniques have been used to evaluate data center environmental data [19]. While statistical and Fourier analysis of air temperature data from rack inlet sensors was performed, the study did not identify events or anomalies within a data center.

SPIRIT [17] performs on-line PCA on n data streams by incrementally updating the principal components as each data point arrives. As long as correlations between these streams continue to hold, the number of hidden variables remains constant. Change in the number of hidden variables indicates anomalous behavior. While our methodology is based on SPIRIT, we make it scalable by using hierarchical groupings, and add a mechanism to filter out hidden variables associated with short-lived trends.

InteMon [14] provides a prototype for monitoring data center information through use of SPIRIT [17]. It analyzes correlations in real-time and alarms on detecting an anomaly. While our work is related to InteMon, there are clear differences. InteMon uses only four temperature sensors, while we analyze a large sensor network consisting of 375 temperature sensors. Using a hierarchical approach makes our technique inherently more scalable. Furthermore, it is only through rich instrumentation that anomalies that we are interested in surface.

In recent years, data streams have been the focus of extensive research. The availability of continuous, real time, dynamic data in systems such as sensor networks and web servers, and, the need for real-time monitoring and analysis have been the prime motivations. Traditional database management systems (DBMS) [11] are not suited to store or process such high volume data streams due to performance and storage limitations. However, data stream management systems (DSMS) [5] have emerged to address this need. They aim to provide DBMS like functionalities for data streams [5, 9, 16].

In addition to DSMS, the other major area of research in data streams – and the one that is the focus of this paper – is mining of data streams for discovering patterns and correlations. Numerous research efforts have focused on clustering and classifying data streams into groups including CluStream [2] and HPStream [3]. Each data stream is passed through an evaluation function – typically based on distance measures – which determines the membership of the stream. Data mining of time series data has been investigated in many research projects including SPIRIT [17], StatStream [21]. StatStream uses discrete Fourier transform for computing statistical measures over time series data streams.

3. STREAM MINING OF SENSOR DATA

Mining of sensor data in a data center can provide important information for its management including control, optimization and fault-tolerance of various devices and processes. Early detection of abnormal events such as failure of a computer room air conditioning (CRAC) unit can be used to redeploy resources [6] and minimize any potential user impact. Cooling resources from other CRAC units can be provided to the affected region. Additionally, if server virtualization techniques are in use, workload can be preemptively migrated to other racks not affected by the failure. Similarly, identification of an errand temperature sensor by observing the correlations between the sensors in the same rack provides valuable information to the data center management system, allowing it to ignore measurements from such sensors instead of taking remedial actions.

3.1 Hierarchical Methodology

The goal of our methodology is to analyze sensor data streams to detect anomalous behavior in a data center. Anomalies that manifest as broken correlations between sensors are detected. In addition to detecting an anomaly, the level or scope of the anomaly is also inferred. In the data center context, this refers to whether the anomalous behavior occurred at a sensor level, a rack level, or in an entire zone of a data center. An advantage of our approach is that no prior learning or training is required. Furthermore, by virtue of being hierarchical, it is very scalable.

The core component of the technique consists of analyzing sensor measurements organized in hierarchical groupings. The analysis comprises performing streaming principal component analysis (PCA) on each grouping and at each level. Considering hierarchical groupings provides a deeper insight into the location and nature of an anomaly. The groupings exploit correlations that exist between sensors during normal operation. In this paper, a simple mechanism of grouping the sensors, based on their physical locality, is used since the expectation is that closely located temperature sensors receive similar air flow and hence are likely to show correlated behavior. We verified this by using historic data to compute correlation coefficients between pairs of sensors in the same group. Note that in the absence of any domain knowledge, these correlation coefficients computed over historic data could be used to group the sensors. Further, our technique is generic and does not depend on the criterion used for the grouping.

We consider three groupings: 1) Rack level, 2) Zone level, and 3) Region level. As shown in Figure 1, PCA is conducted in a bottom up fashion starting at the rack level. At each level, trends in sensor measurements are evaluated to identify anomalous behavior of the entities comprising that level. Analysis at a particular level requires trends from the level below. For example, zone level analysis requires rack trends and allows rack anomalies to be discovered. Trends (hidden variables) identified as anomalous at a particular level are removed from analysis at higher levels. The three hierarchical levels considered are described below.

Rack Level. This is the lowest level consisting of sensors located in a rack. The objective of rack level analysis is to identify anomalous behavior at the scale of a sensor. Incremental PCA is separately performed on groups of sensor data streams from each rack. The expectation is that during normal operation sensors in the same rack are correlated. Ceasing of this correlation indicates an anomaly. In PCA, this is reflected by a change in the number of hidden variables. Furthermore, the sensor(s) associated with the anomalous behavior is (are) also identified as discussed in the next section.

Zone Level. A zone consists of a group of racks. Zones are demarcated based on commonality of an attribute related to the sensor measurements. For example, for temperature sensors, racks in a row are considered one zone. The objective of zone level analysis is to identify entire racks within a zone that show anomalous behavior. Analysis at this level utilizes the results of the rack level analysis. The trends (hidden variables) – discovered in the rack level analysis – of each rack in a zone are analyzed together to determine if the number of trends remain preserved. An additional trend indicates anomalous behavior. An example of a rack level anomaly is an obstruction blocking a cool air vent next to a rack. This causes the rack sensors to exhibit deviant behavior. Note that rack level analysis is unlikely to uncover this problem, since the blocked vent affects all sensors in the rack, which are likely to remain correlated.

Region Level. This level consists of a number of related zones. For example, all zones under the influence of a particular CRAC unit can be considered together. The objective of analysis at this level is to discovery aberrant zones. The main trends from each zone – computed in the zone level analysis – are used to perform incremental PCA. The emergence of additional trends in the results indicates anomalous behavior.

Although, in this paper, we only use temperature sensors, sensors measuring other characteristics such as humidity can also be simultaneously analyzed to detect related anomalies [20]. Furthermore, even with one kind of sensor, different criteria for association of sensors can be used. In addition to physical locality, locality based on data center cooling infrastructure or workload/power consumption of servers can be exploited.

3.1.1 Pre-processing

Before being passed through the PCA algorithm, a data stream is pre-processed to make it more amenable to PCA analysis. This consists of two main components. First, high frequency noise is removed through use of a moving average filter. Then, the data is normalized such that it has zero mean and a standard deviation of one, that is,

$$T' = (T - \mu)/\sigma \tag{1}$$

Although this is trivial to do for historical data, efficiently computing mean and standard deviation over a sliding window for streaming data is challenging. While several research efforts [21, 10, 4] have focused on computing statistics for streaming data without having to store an entire window's worth of data, we use a simple solution.

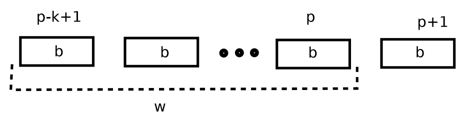

Figure 2: Computing sliding window mean for streaming data.

The basic idea is to divide the sliding window into blocks and statistics related to these blocks are preserved and used to update statistics over the sliding window. Assume a window size of w is divided into k blocks of size b, as shown in Figure 2. At the beginning/end of each block, the statistics can be accurately computed. The sliding window mean at the end of block $p+1$ is given by

$$\mu_{w,p+1} = \mu_{w,p} - \frac{\mu_{b,p-k+1} \cdot b}{w} + \frac{\mu_{b,p+1} \cdot b}{w} \tag{2}$$

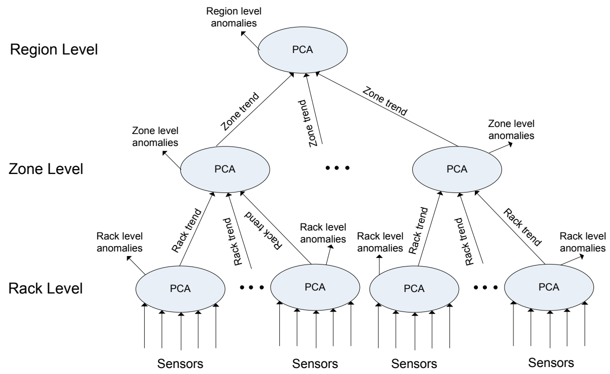

Figure 1: Our hierarchical PCA methodology applied to data centers with groupings of temperature sensors at three levels, namely, rack, zone and region.

where $\mu_{w,p}$ is the sliding window mean at the end of block p. While in a block, the mean can be estimated by assuming that the data point moving out of the sliding window is equal to the average of its block and updating the mean with the newly arrived point. Standard deviation of streaming data can be similarly computed. It requires that sum of squares for each block be also saved.

Since temperatures depend on server workloads, the appropriate choice of the window size is governed by the workload patterns. In our analysis, based on the observed diurnal workload pattern in the data center, we use a window size of 24 hours with block size of 15 minutes.

3.1.2 PCA of streaming data

Our methodology to discover trends and anomalous behavior in data center sensor streams involves using principal component analysis (PCA) [15]. PCA is a generic technique to reduce the dimensionality of correlated variables by introducing a new orthogonal basis. These are called the principal components (PCs). Each PC is successively chosen such that it captures the maximum variance remaining in the data. Thus, usually the first few PCs are sufficient for reconstructing the data to a good approximation. Since the PC directions are orthogonal, they are uncorrelated. Note that PCA only considers linear dependence; non-linear interdependence between variables is not captured by PCA. Another assumption is that the data has a normal distribution, a property satisfied by the temperature sensor data considered in this paper.

At each time tick, a data point (vector containing a measurement from each sensor) is received and transformed from the original n-dimensional space to the new m-dimensional space by taking its projection onto the PCs.

$$\mathbf{y}_{m\times 1} = \mathbf{W}_{m\times n} \cdot \mathbf{x}_{n\times 1} \tag{3}$$

where $\mathbf{x}$ is the input vector; $\mathbf{y}$ is the output vector in the PC space (the components of $\mathbf{y}$ are also called hidden variables); $\mathbf{W}$ is the projection matrix with its ith row containing a unit vector along the ith PC. A row vector of $\mathbf{W}$ is also called the participation weight vector since its elements determine the contribution of an input (x_i) to a hidden variable (y_i). This is very useful information since it can be used to rank the contributions of input variables to a particular hidden variable. The original variables can be reconstructed as follows:

$$\tilde{\mathbf{x}}_{n\times 1} = \mathbf{W}^T_{n\times m} \cdot \mathbf{y}_{m\times 1} \tag{4}$$

The reconstruction error is given by $||\mathbf{x} - \tilde{\mathbf{x}}||^2$.

The basic assumption in using PCA for anomalous behavior detection is that during normal operation the number of PCs remains constant. An increase or decrease in the number of hidden variables indicates an underlying change in the number of correlations of the original data and hence considered anomalous behavior. While our application of PCA to streaming data is based on SPIRIT [17], we improve scalability by hierarchical processing. We also make one other enhancement (described further in the next section): the criterion for determining the number of hidden variables is modified such that short-lived and insignificant trends are ignored. The algorithm incrementally updates the PCs (matrix W) as each data point arrives. It is efficient with $O(mn)$ complexity in both time and space and is independent of the total number of data points seen. In our analysis each sensor measurement is considered a separate dimension. Thus, n is equal to the number of sensors being analyzed.

3.1.3 Number of hidden variables

The number of hidden variables depends on the degree of reconstruction accuracy desired. A common technique to estimate this number is energy thresholding [17, 15]. Energy of a variable is defined as the average sum of squares of all its past values.

$$E(i) \leftarrow E(i) + {y_i}^2 \qquad i \in [1, m] \tag{5}$$

Energy thresholding operates as follows. The energies of the reconstructed and original variables are compared. As long as this ratio is within threshold limits (e.g. 0.95 and 0.98), the number of hidden variables is kept constant. If it falls below the lower threshold (indicating unacceptably high reconstruction error), the number of hidden variables is increased. On the other hand, if it rises above the upper threshold (indicating unacceptably low reconstruction error), the number is decreased.

An issue with energy thresholding is that small changes in the value of the energy ratio around the threshold values increases or decreases the number of hidden variables, signaling anomalous behavior. However, these new trends created may be short lived and insignificant, likely related to a transient phenomenon in the original data. In order to filter out such trends, energy thresholding is enhanced to also consider the energy contribution of a new hidden variable in conjunction with the thresholds. A new hidden variable, i, is considered viable only if it has made a significant contribution to the total energy since it appeared, i.e.,

$$E(i)_a \geq \alpha \cdot E_a \tag{6}$$

continues to hold for b time ticks. Here, $E(i)_a$ is the contribution of the ith hidden variable since time a; E_a is the total energy since time a; and, α is the contribution threshold. The parameters α and b can be adjusted based on the degree of sensitivity desired. For the results described in section 5, the values of α and b were set at 0.4 and 6, respectively. These values worked well for the temperature data analyzed.

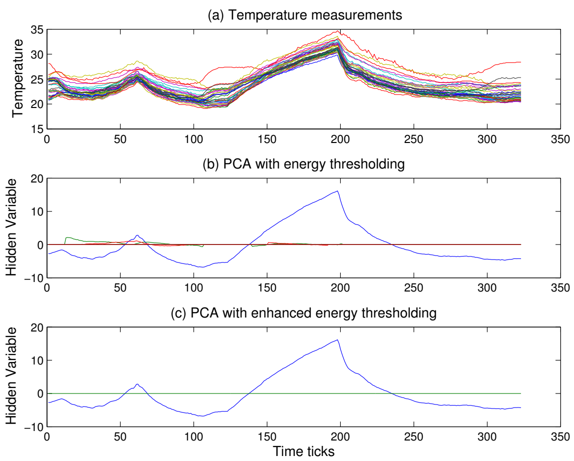

Figure 3: Use of enhanced energy thresholding with PCA analysis removes insignificant trends.

Figure 3 (a) shows temperature measurements from seven racks, that is, 35 sensors in all. The hidden variables that result from conducting incremental PCA are shown in Figure 3 (b). In addition to the main trend, four short-lived trends (appearing at time 12, 23, 139 and 150) are also seen. These are caused by transitory disturbances and are not significant trends. Although uninteresting, these events are not distinguished from cases where a major new trend appears since both are signaled by the appearance of a hidden variable. However, using the mechanism described above, these insignificant trends are filtered out (shown in Figure 3 (c)). In all the results described in section 5, this hidden variable filtering algorithm was used.

3.1.4 Summary of Methodology

The following summarizes the steps involved in analyzing data at each time tick.

- Pre-process incoming data points: synchronize using linear interpolation, if needed; normalize and smoothen using moving averages.
- Perform streaming PCA for each rack; identify sensor level anomalous behavior, if any, and the associated sensors.
- Use rack level trends to perform streaming PCA at zone level; identify racks with anomalous behavior, if any.
- Use zone level trends to perform streaming PCA at the region level; identify zones with anomalous behavior, if any.

4. EXPERIMENTAL TEST BED

We apply our analysis and data stream mining methodology to a real life, state-of-the-art data center. In this study, temperature sensor data from a production data center is considered. Power consumption on a per rack basis in this data center ranges from 5 to 20kW. Racks comprise of off-the-shelf standard or blade servers, storage arrays and network switches. Note that our methodology is generic and not limited to the data center architecture presented here.

4.1 Data Center Infrastructure

These data centers are air-cooled with a raised floor plenum to distribute cool air, power and networking. Figure 4 depicts a typical state-of-the-art data center air-conditioning environment with under-floor cool air distribution [18]. Computer room air conditioning (CRAC) units cool the exhaust hot air from the computer racks. Energy consumption in data center cooling comprises work done to distribute the cool air to the racks and to extract heat from the hot exhaust air. The air movers in the CRAC units pressurize the plenum with cool air which enters the data center through vented tiles located on the raised floor close to the inlet of the racks. Typically the racks are laid out in rows separated by hot and cold aisles as shown in Figure 4. This separation is done for thermal efficiency considerations. Air inlets for all racks face cold aisles while hot air is expelled to hot aisles.

4.2 Sensor Network and Data Aggregation

Temperature data is collected from sensors, mounted at the inlet and outlet of racks (see Figure 5). A data center wide distribution of such temperature sensor networks are deployed on rack-to-rack basis. The placement density of the sensors is based on the power dissipated per unit area of a data center. The temperature sensors are mounted on racks as shown in the figure and provide temperature data at both air inlet and outlet of the racks. The digital output from each sensor is accurate to 0.5 C in the range of interest.

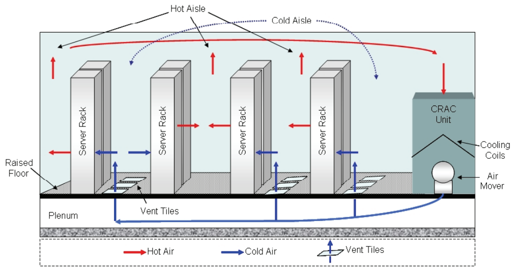

Figure 4: A typical raised-floor data center.

Since the sensor is primarily a transistor with compensation for leakage, no calibration is needed. Ten temperature sensors are attached to each rack, with five at the inlet and the other five at the outlet. Each rack also contains a base station to which all sensors on a rack are connected. The base station has an Ethernet interface and multicasts the temperature data collected on the data center LAN. In addition to temperature sensors, data is collected from CRAC units, Variable fan drive (VFD) units and power distribution units (PDUs). However, in this paper, only rack inlet temperature data is considered since it is more critical (as compared to outlet temperature) in determining the thermal well-being of the entities in a data center.

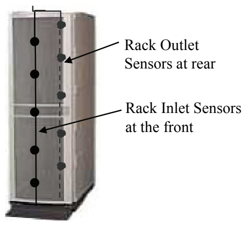

Figure 5: Sensors mounted at the rack inlet and exhaust.

An underlying assumption in the use of PCA on a data set is that it is normally distributed. Figure 6 shows the cumulative frequency distribution (CFD) of typical instances of the temperature sensor data taken from the test bed data center. The standard normal curve is also shown for comparison. In the anomaly-free case, the close agreement between the normal CFD and the temperature data CFD indicates that the temperature data is normally distributed with random variations. The anomalous data deviates slightly from the normal curve due to the systemic variation in the temperature values because of the anomaly.

5. RESULTS AND DISCUSSION

As a proof of concept, we have implemented a prototype of our methodology and applied it to sensor data streams obtained from a real-life production data center located in Palo Alto, CA. The servers in this 3000 sq. ft. facility dissipate 350 KW of power. Its architecture is similar to that described in Section 4. There are 75 racks of computing equipment, each with 10 temperature sensors, arranged in rows as shown in Figure 7. Each rack can contain up to 64 blade servers or 42 1U servers. Six CRAC units provide

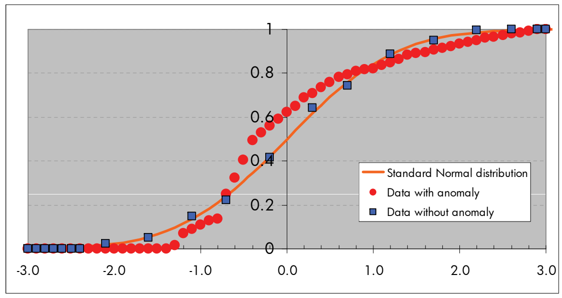

Figure 6: Distribution of temperature sensor data.

cooling. Temperature data streams from five sensors located at the air inlet of each rack, resulting in 375 data streams in all, are analyzed.

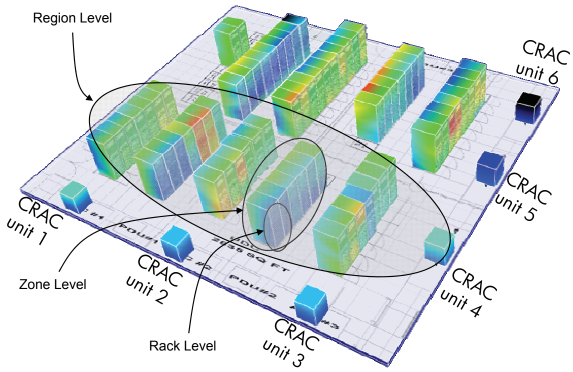

Figure 7: Layout of the test bed data center.

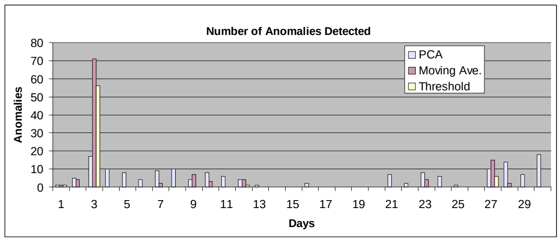

Figure 8: Number of anomalies detected over a period of 30 days.

Figure 8 shows the number of rack anomalies detected on each day for a 30 day period, from January 1, 2008, to January 30, 2008. In addition to our methodology (labeled PCA), also shown are the number of rack anomalies detected through (1) a threshold method, where an anomaly is flagged if any temperature sensor in a rack exceeds 30°C, and (2) a moving average method, where an anomaly is flagged if a rack temperature is greater than 5°C from the moving average of the previous 24 hours. During this period a major failure occurred on day 3 when all CRAC units failed due to unavailability of chilled water, and a similar minor failure occurred on day 27. These are captured well by the moving average and threshold methods. The PCA method does not appear to do well for such large scale anomalies where temperature of sensors remain correlated while increasing. However, many anomalies manifest with no violation of temperature thresholds and are thus particularly hard to detect. Several of these can be detected through the PCA method since they result in uncorrelated behavior between sensors. These anomalies indicate inefficient airflow in the data center and result in higher power consumption. The cause of airflow inefficiencies could be related to misconfiguration of equipment, or increased recirculation of hot air. However, automatic determination of the cause of a particular anomaly is beyond the scope of the current work. Anomaly detection allows an operator to investigate further and if required take corrective measures to fix airflow inefficiencies, thus, saving power. Note that the threshold-based and PCA-based methods compliment each other.

In order to validate the performance of the PCA method, a thermal sciences expert visually inspected the daily rack temperature plots for the 30 days and identified racks that seemed to exhibit abnormal behavior. Each of the 75 racks, for each of the 30 days, were marked as anomalous or normal. These labeled samples were then compared with the results obtained using PCA. The resulting confusion matrix is shown in Table 1. In all, there are 2250 day-long samples (75 racks over 30 days). In the table, *Positive* indicates presence of an anomaly while *Negative* indicates its absence. 135 anomalous and 2069 normal samples are correctly classified. There are 26 false positive samples while 20 are false negatives. There are 155 anomalies in all (about 7%). Since the anomaly rate is relatively low, the total *accuracy*, that is, proportion of correctly classified samples, of 97.96%, although high, is not very significant. The *sensitivity*, which measures the true positive rate, and the *specificity*, which measures the true negative rate, are better indicators of the performance. As shown in Table 2, these are 87.1% and 98.76%, respectively. The precision of the PCA method, that is, the proportion of true positives out of the total number of positives, is 83.85%.

		PCA Method		
		Positive	Negative	Total
Actual	Positive	135	20	155
	Negative	26	2069	2095
	Total	161	2089	2250

Table 1: Results from the PCA method as compared to the actual positive (anomalous) and negative (normal) results, as provided by the domain (thermal sciences) expert.

Measure	Value(%)
Accuracy	97.96
Sensitivity	87.1
Specificity	98.76
Precision	83.85

Table 2: Summary of the performance of the PCA method.

In the following sections, we present some qualitative results from the 30 day run and show how the hierarchical analysis allows the source and scope of an anomaly to be identified.

5.1 Rack Level Analysis

Figure 9 (a) shows the temperature measurements from five sensors located on the same rack (A1). Each time tick corresponds to about 1 minute. The key point to note is that although the temperature varies in the rack, the five sensors follow the same trend. This trend is captured by a single hidden variable obtained by conducting PCA (shown in Figure 9(b)). This also shows the usefulness of hidden variables in summarizing trends.

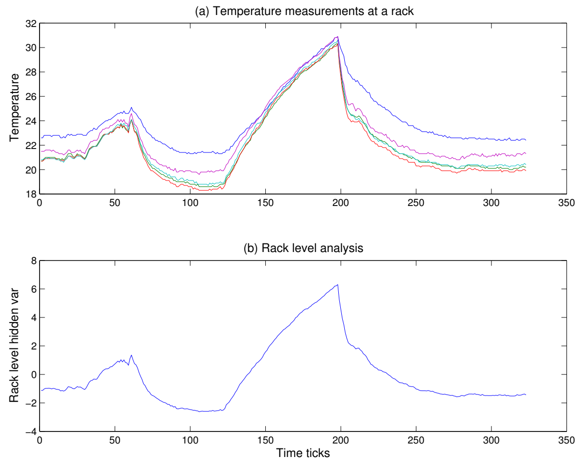

Figure 9: (a) Rack temperature data; (b) One hidden variable is able to capture all five temperature sensors.

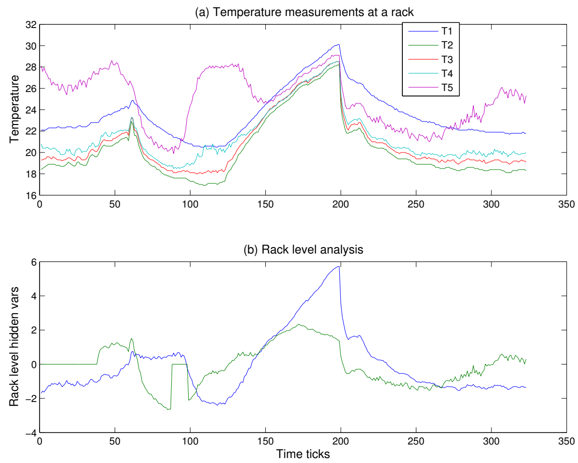

Figure 10: Analysis of Uncorrelated Rack Temperature Data.

Five temperature measurements from a different rack (Bext4), during the same period of time, are shown in Figure 10(a). After conducting PCA, we discover two trends (Figure 10(b)). The larger trend is similar to the one seen for the previous rack (A1); however, an additional trend starting at time tick 38 is also seen. The largest contributor to this new trend – as determined from the weight vector – is sensor 5

(T5). Although the fact – that T5 shows deviant behavior – is quite apparent from the temperature plot, the ability to identify this behavior and the particular sensor involved autonomously in a data center with thousands to tens of thousands of sensors is a big advantage. Furthermore, in this case, the new trend is detected before (at time tick 38) it becomes visually obvious from the temperature plot (between time ticks 50 and 100). This is an extremely useful input to a data center monitoring and control system which can perform further analysis to determine the root cause, or take other preemptive actions. Note that since the deviant sensor shows temperatures that are within the normal operation range, a threshold based mechanism will be unable to detect this behavior.

5.2 Zone Level Analysis

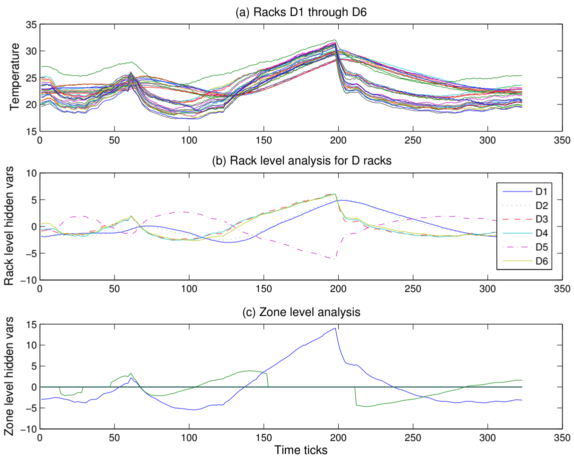

Figure 11: Zonal Analysis of Rack Temperature Data.

At this level, a group of racks, organized as a zone, is analyzed with the objective of detecting anomalous behavior in an entire rack. Figure 11(a) shows the raw temperatures of six racks (D1 through D6). These racks, comprising a zone, are located in the same aisle. The main trends (hidden variables) for the six racks, computed during rack level analysis, is shown in Figure 11(b). These six variables, each representing one rack, are passed through another round of PCA. The results, shown in Figure 11(c), indicate two hidden variables. The smaller trend can be traced to racks D1 and D2. The larger one represents the other racks. Note that in Figure 11(b) trend D5 is essentially the same as D3, D4 and D6 (inverting D5 will result in a close approximation of the others). The results indicate that racks D1 and D2 show anomalous behavior as compared to the other racks in the zone. Another observation (from Figure 11 (c)) is that the anomalous behavior is intermittent as on two occasions the second hidden variable disappears. Although deviant behavior can be identified, the cause of the deviance cannot be inferred though this analysis.

5.3 Region Level Analysis

At the region level, trends from multiple zones are analyzed together to detect the existence of zone-wide anomalous behavior. Note that an anomaly impacting an entire zone may not be detected at the zone level analysis, since the zone may continue to show correlated behavior. However, conducting PCA on multiple zones, that show correlated behavior during normal operation, can facilitate identification of entire zones that exhibit anomalous behavior.

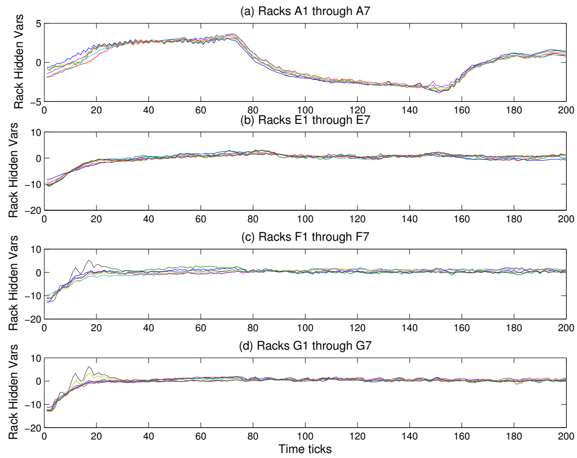

Figure 12: Hidden Variable plots from rack level analysis for zones A, E, F and G.

Figure 12 shows the hidden variables of racks, obtained from rack level analysis, for four different zones (Zones A, E, F and G). Each zone consists of seven racks in a single aisle and each rack is summarized by one hidden variable. Zone level trends for the four zones are plotted in Figure 13 (a). Note that each zone can be represented by one hidden variable implying that within each of these zones the temperature behavior is highly correlated.

PCA is performed on the four zone level hidden variables and the results are plotted in Figure 13 (b). Two distinct trends can be seen. Trend T1 strongly corresponds to Zone A (as determined from the participation weight vector) while trend T2 is associated with the remaining zones, namely, E, F and G. The implication is that while the behavior of zones E, F and G remains correlated, zone A shows anomalous behavior. Note that this is obvious from the rack level hidden variables (Figure 12) where racks A1 through A7 show markedly different behavior than the other racks. The key advantage is that this distinction can be autonomously deduced without human involvement. From knowledge of the data center during this time period, it is known that the settings at a CRAC unit next to zone A racks were being manually changed. Due to their location, Racks E, F and G were not impacted by this event. The region level analysis is aimed at detection of such larger scale anomalies.

6. CONCLUSIONS

Timely and specific discovery of anomalous behavior is vital for efficient and cost-effective management of state-of-the-art data centers hosting tens of thousands of servers. Considering the large volumes of sensor data continuously being produced, automated mechanisms for discovering anomalies and trends are indispensable. In this paper, we used incremental PCA on data streams generated by a large temperature sensor network in a data center. This allowed hard-

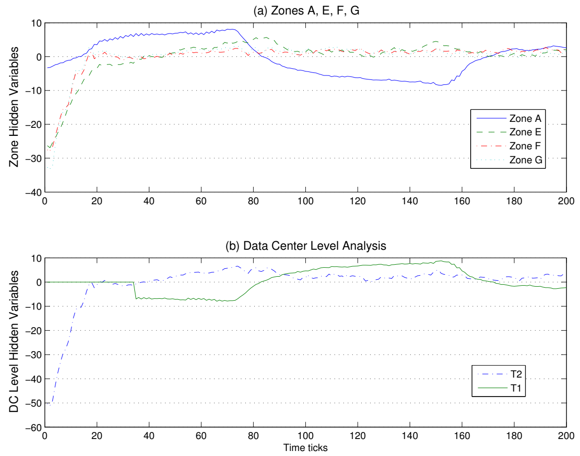

Figure 13: Zonal and Region level Analysis.

to-detect anomalous behavior resulting from airflow inefficiencies in a data center to be detected and then potentially fixed to save energy. A hierarchical methodology, that is inherently scalable and allows the scope of an anomaly to be determined, was proposed. Furthermore, an enhanced mechanism to detect new hidden variables — that filters short-lived and insignificant trends — was presented. Our methodology was deployed in a production data center and we presented results from 30 continuous days of operation involving 75 racks and 375 sensors. The results validate the performance of our methodology, with the accuracy, sensitivity, and specificity being 97.96%, 87.1% and 98.76%, respectively.

7. REFERENCES

[1] U. E. P. Agency. Report to congress on server and data center energy efficiency public law. pages 109–431, 2007.

[2] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for clustering evolving data streams. In *vldb'2003: Proceedings of the 29th international conference on Very large data bases*, pages 81–92. VLDB Endowment, 2003.

[3] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for projected clustering of high dimensional data streams. In *VLDB '04: Proceedings of the Thirtieth international conference on Very large data bases*, pages 852–863. VLDB Endowment, 2004.

[4] A. Arasu and G. S. Manku. Approximate counts and quantiles over sliding windows. In *PODS '04: Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 286–296, New York, NY, USA, 2004. ACM.

[5] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. In *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 1–16, New York, NY, USA, 2002. ACM.

[6] C. Bash and et. al. Dynamic thermal management of aircooled datacenter. In *ITherm2006Intersoc. Conf. Thermo-mech. Phenomena in Elec. Sys.,*, 2006.

[7] C. L. Belady. In the data center, power and cooling costs more than the it equipment it supports. *Electronics Cooling*, 13(1), Feb 2007.

[8] C. E. Board. Data center consolidation. *Infrastructure Executive Council*, 2005.

[9] S. Chandrasekaran, O. Cooper, A. Deshpande, M. J. Franklin, J. M. Hellerstein, W. Hong, S. Krishnamurthy, S. R. Madden, F. Reiss, and M. A. Shah. Telegraphcq: continuous dataflow processing. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 668–668, New York, NY, USA, 2003. ACM.

[10] M. Datar, A. Gionis, P. Indyk, and R. Motwani. Maintaining stream statistics over sliding windows: (extended abstract). In *SODA '02: Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 635–644, Philadelphia, PA, USA, 2002. Society for Industrial and Applied Mathematics.

[11] L. Golab and M. T. Özsu. Issues in data stream management. *SIGMOD Rec.*, 32(2):5–14, 2003.

[12] C. Greiner. Considerations for a 'green' energy-efficient data center. *Ovum*, 2008.

[13] T. C. Group. Smart 2020: Enabling the low carbon economy in the information age, 2008. http://www.theclimategroup.org.

[14] E. Hoke, J. Sun, J. D. Strunk, G. R. Ganger, and C. Faloutsos. Intemon: continuous mining of sensor data in large-scale self-infrastructures. *SIGOPS Oper. Syst. Rev.*, 40(3):38–44, 2006.

[15] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002.

[16] A. Lerner and D. Shasha. Aquery: query language for ordered data, optimization techniques, and experiments. In *vldb'2003: Proceedings of the 29th international conference on Very large data bases*, pages 345–356. VLDB Endowment, 2003.

[17] S. Papadimitriou, J. Sun, and C. Faloutsos. Streaming pattern discovery in multiple time-series. In *VLDB '05: Proceedings of the 31st international conference on Very large data bases*, pages 697–708. VLDB Endowment, 2005.

[18] R. Schmidt. Computer and telecommunications equipment room cooling: A review of literature. In *Proceedings of Eighth Intersociety Conference on Thermal and Thermomecanical Phenomena in Electronic Systems*, May 2002.

[19] R. Sharma and et. al. Application of exploratory data analysis to temperature data in conventional data centers. In *Proc, IPACK*, 2007.

[20] J. Sun, D. Tao, and C. Faloutsos. Beyond streams and graphs: dynamic tensor analysis. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 374–383, New York, NY, USA, 2006. ACM.

[21] Y. Zhu and D. Shasha. Statstream: statistical monitoring of thousands of data streams in real time. In *VLDB '02: Proceedings of the 28th international conference on Very Large Data Bases*, pages 358–369. VLDB Endowment, 2002.