

Data Mining for Big Dataset-related Thermal Analysis of High Performance Computing (HPC) Data Center

Davide De Chiara¹, Marta Chinnici², Ah-Lian Kor³

¹ ENEA-ICT Division, C.R. Portici Piazzale Enrico Fermi 1, Napoli 80055, Italy
davide.dechiara@enea.it

² ENEA-ICT Division, C.R. Casaccia Via Anguillarese 301, ROMA 00123, Italy
marta.chinnici@enea.it

³ Leeds Beckett University, Leeds, UK
a.kor@leedsbeckett.ac.uk

Abstract. Greening of Data Centers could be achieved through energy savings in two significant areas, namely: compute systems and cooling systems. A reliable cooling system is necessary to produce a persistent flow of cold air to cool the servers due to increasing computational load demand. Servers' dissipated heat effects a strain on the cooling systems. Consequently, it is necessary to identify hotspots that frequently occur in the server zones. This is facilitated through the application of data mining techniques to an available big dataset for thermal characteristics of High-Performance Computing ENEA Data Center, namely Cresco 6. This work presents an algorithm that clusters hotspots with the goal of reducing a data centre's large thermal-gradient due to uneven distribution of server dissipated waste heat followed by increasing cooling effectiveness.

Keywords: Data Center, HPC, Data Mining, Big Data, Thermal, Hotspot, Cooling, Thermal management.

1 Introduction

A large proportion of worldwide generated electricity is through hydrocarbon combustion. Consequently, this causes a rise in carbon emission and other Green House Gasses (GHG) in the environment, contributing to global warming. Data Center (DC) worldwide were estimated to have consumed between 203 to 271 billion kWh of electricity in the year 2010 [1] and in 2017, US based DCs alone used up more than 90 billion kilowatt-hours of electricity [14]. According to [2], unless appropriate steps are taken to reduce energy consumption and go-green, global DC share of carbon emission is estimated to rise from 307 million tons in 2007 to 358 million tons in 2020. Servers in DCs consume energy that is proportional to allocated computing loads, and unfortunately, approximately 98% of the energy input is being dissipated as waste heat energy. Cooling systems are deployed to maintain the temperature of the computing servers at the vendor specified temperature for consistent and reliable performance. Koomey [1] emphasises that a DC energy input is primarily consumed by cooling and compute systems (comprising servers in chassis

and racks). Thus, these two systems have been critical targets for energy savings. Computing-load processing entails jobs and tasks management. On the other hand, DC cooling encompasses the installation of cooling systems and effective hot/cold aisle configurations. Thermal mismanagement in a DC could be the primary contributor to IT infrastructure inefficiency due to thermal degradation. Server microprocessors are the primary energy consumers and waste heat dissipators [4]. Generally, existing DC air-cooling systems are not sufficiently efficient to cope with the vast amount of waste heat generated by high performance-oriented microprocessors. Thus, it is necessary to disperse dissipated waste heat so that there will be an even distribution of waste heat within a premise to avoid overheating. Undeniably, a more effective energy savings strategy is necessary to reduce energy consumed by a cooling system and yet efficient in cooling the servers (in the compute system). One known technique is thermal-aware scheduling where a computational workload scheduling is based on waste heat. Thermal-aware schedulers adopt different thermal-aware approaches (e.g. system-level for work placements [16]; execute 'hot' jobs on 'cold' compute nodes; predictive model for job schedule selection [17]; ranked node queue based on thermal characteristics of rack layouts and optimisation (e.g. optimal setpoints for workload distribution and supply temperature of the cooling system). Heat modelling provides a model that links server energy consumption and their associated waste heat. Thermal-aware monitoring acts as a thermal-eye for the scheduling process and entails recording and evaluation of heat distribution within DCs. Thermal profiling is based on useful monitoring information on workload-related heat emission and is useful to predict the DC heat distribution. In this paper, our analysis explores the relationship between thermal-aware scheduling and computer workload scheduling. This is followed by selecting an efficient solution to evenly distribute heat within a DC to avoid hotspots and cold spots. In this work, a data mining technique is chosen for hotspots detection and thermal profiling for preventive measures. The novel contribution of the research presented in this paper is the use of real thermal characteristics big dataset for ENEA High Performance Computing (HPC) CRESCO6 compute nodes. Analysis conducted are as follows: hotspots localisation; users categorisations based on submitted jobs to CRESCO6 cluster; compute nodes categorisation based on thermal behaviour of internal and surrounding air temperatures due to workload related waste heat dissipation. This analysis aims to minimise employ thermal gradient within a DC IT room through the consideration of the following: different granularity levels of thermal data; energy consumption of calculation nodes; IT room ambient temperature. An unsupervised learning technique has been employed to identify hotspots due to the variability of thermal data and uncertainties in defining temperature thresholds. This analysis phase involves the determination of optimal workload distribution to cluster nodes. Available thermal characteristics (i.e. exhaust temperature, CPUs temperatures) are inputs to the clustering algorithm. Subsequently, a series of clustering results are intersected to unravel nodes (identified by IDs) that frequently fall into high-temperature areas. The paper is organised as follows: Section I – Introduction; Section II – Background: Related Work; Section III – Methodology; Section IV – Results and Discussion; Section V – Conclusions and Future Work.

2 Background: Related Work

In the context of High Performance Computing Data Center (HPC-DC), it is essential to satisfy service level agreements with minimal energy consumption. This will involve the following: DC efficient operations and management within recommended IT room requirements, specifications, and standards; energy efficiency and effective cooling systems; optimised IT equipment utilisation. DC energy efficiency has been a long standing challenge due to multi-faceted factors that affect DC energy efficiency and adding to the complexity, is the trade-off between performance in the form of productivity and energy efficiency. Interesting trade-offs between geolocations and DC energy input requirements (e.g. cold geolocations and free air-cooling; hot, sunny geolocations and solar powered renewable energy) are yet to be critically analysed [8]. One of the thermal equipment-related challenge is raising the setpoint of cooling equipment or lowering the speed of CRAC (Computer Room Air Conditioning) fans to save energy, may in the long-term, decrease the IT systems reliability (due to thermal degradation). However, a trade-off solution (between optimal cooling system energy consumption and long-term IT system reliability) is yet to be researched on [8]. Another long-standing challenge is IT resource over-provisioning that causes energy waste due to for idle servers. Relevant research explores optimal allocation of PDUs (Power Distribution Units) for servers, multi-step algorithms for power monitoring, and on-demand provisioning reviewed in [8]. Other related work addresses workload management, network-level issues as optimal routing, Virtual Machines (VM) allocation, and balance between power savings and network QoS (Quality of Service) parameters as well as appropriate metrics for DC energy efficiency evaluation. One standard metric used by a majority of industrial DCs is Power Usage Effectiveness (PUE) proposed by Green Grid Consortium [2]. It shows the ratio of total DC energy utilisation with respect to the energy consumed solely by IT equipment. A plethora of DC energy efficiency metrics evaluate the following: thermal characteristics; ratio of renewable energy use; energy productivity of various IT system components, and etc. There is a pressing need to provide a holistic framework that would thoroughly characterise DCs with a fixed set of metrics and reveal potential pitfalls in their operations. Though some existing research work has made such attempts but to date, we are yet to have a standardised framework [9] [13]. To reiterate, the thermal characteristics of the IT system ought to be the primary focus of an energy efficiency framework because it is the main energy consumer within a DC. Several researches have been conducted to address this issue. Sungkap et al. [11] propose an ambient temperature-aware capping to maximize power efficiency while minimising overheating. Their research includes an analysis of the composition of energy consumed by a cloud-based DC. Their findings for the composition of DC energy consumption are approximately 45% for compute systems; 40% for refrigeration-based air conditioning; remaining 15% for storage and power distribution systems. This implies that approximately half of the DC energy is consumed by non-computing devices. In [6], Wang and colleagues present an analytical model that describes DC resources with heat transfer properties and workloads with thermal features. Thermal modelling and temperature estimation from thermal sensors ought to consider the emergence of server hotspots and thermal solicitation due to the increase in inlet air temperature, inappropriate positioning of a rack or even inadequate room ventilation. Such phenomena are unravelled by

thermal-aware location analysis. The thermal-aware server provisioning approach to minimise the total DC energy consumption calculates the value of energy by considering the maximum working temperature of the servers. This approach should consider the fact that any rise in the inlet temperature rise may cause the servers to reach the maximum temperature resulting in thermal stress, thermal degradation, and severe damage in the long run. Typical different identified types of thermal-aware scheduling are reactive, proactive and mixed. However, there is no reference to heat-modelling or thermal-monitoring and profiling. Kong and colleagues [4] highlight important concepts of thermal-aware profiling, thermal-aware monitoring, and thermal-aware scheduling. Thermal-aware techniques are linked to the minimisation of waste heat production, heat convection around server cores, task migrations, and thermal-gradient across the microprocessor chip and microprocessor power consumption. Dynamic thermal management (DTM) techniques in microprocessors encompasses the following: Dynamic Voltage and Frequency Scaling (DVFS), Clock gating, task migration, and Operating System (OS) based DTM and scheduling. In [5], Parolini and colleagues propose a heat model; provide a brief overview of power and thermal efficiency from microprocessor micro-level to DC macro-level. To reiterate, it is essential for DC energy efficiency to address thermal awareness in order to better understand the relationship between both the thermal and the IT aspects of workload management. In this paper, the authors incorporate thermal-aware scheduling, heat modelling, thermal aware monitoring and thermal profiling using a big thermal characteristic dataset of a HPC-Data Center. This research involves measurement, quantification, and analysis of compute nodes and refrigerating machines. The aim of the analysis is to uncover underlying causes that causes temperatures rise that leads to the emergence of thermal hotspots. Overall, effective DC management requires energy use monitoring, particularly, energy input, IT energy consumption, monitoring of supply air temperature and humidity at room level (i.e. granularity level 0 in the context of this research), monitoring of air temperature at a higher granularity level (i.e.at Computer Room Air Conditioning/Computer Room Air Handler (CRAC/CRAH) unit level, granularity level 1). Measurements taken are further analysed to reveal extent of energy use and economisation opportunities for the improvement of DC energy efficiency level (granularity level 2). DC energy efficiency metrics will not be discussed in this paper. However, the discussion in the subsequent section primarily focuses on thermal guidelines from American Society of Heating, Refrigerating And AC Engineers (ASHRAE) [7].

3 Methodology

To reiterate, our research goal is to reduce DC wide thermal-gradient, hotspots and maximise cooling effects. This entails the identification of individual server nodes that frequently occur in the hotspot zones through the implementation of a clustering algorithm on the workload management platform. The big thermal characteristics dataset of ENEA Portici CRESCO6 computing cluster is employed for the analysis. It has 24 measured values (or features) for each single calculation node (see Table 1) and comprises measurements for the period from May 2018 to January 2020. Briefly, the cluster CRESCO6 is a High-Performance Computing System (HPC)

consisting of 434 calculation nodes with a total of 20832 cores. It is based on Lenovo Think System SD530 platform, an ultra-dense and economical two-socket server in a 0.5 U rack form factor inserted in a 2U four-mode enclosure. Each node is equipped with 2 Intel Xeon Platinum 8160 CPUs (each with 24 cores) and a clock frequency of 2.1 GHz; a RAM size of 192 GB, corresponding to 4 GB/core. A low-latency Intel Omni-Path 100 Series Single-port PCIe 3.0 x16 HFA network interface. The nodes are interconnected by an Intel Omni-Path network with 21 Intel Edge switches 100 series of 48 ports each, bandwidth equal to 100 GB/s, and latency equal to 100ns. The connections between the nodes have 2 tier 2:1 no-blocking tapered fat-tree topology. The power consumption massive computing workloads amount to a maximum of 190 kW.

Table 1. Thermal dataset – description of features.

Node Name	server ID, integer from 1 to 434
Timestamp	timestamp of a measurement
System, CPU, Memory Power	Node instantaneous power, memory power, CPU power use in three corresponding columns (expressed in Watt)
Fan 1a, Fan1b, ..., Fan 5a, Fan 5b	Speed of a cooling fan installed inside the node (expressed in RPM - revolutions per minute)
System, CPU, Memory, I/O utilisation	Percentage values of CPU, RAM memories and I/O utilisation
Inlet, CPU1, CPU2, Exhaust temperature	Temperature at the front, inside (on CPU1 and CPU2) and at the rear of every single node (expressed in Celsius)
SysAirFlow	Speed of air traversing the node expressed in CFM (cubic foot per minute)
DC Energy	Meter of total energy used by the node, updated at corresponding timestamp and expressed in kWh

3.1 Energy Saving Approach

This work incorporates thermal-aware scheduling, heat modelling, and thermal monitoring followed by subsequent user profiling based on “waste heat production” point of view. Thermal-aware DC scheduling is designed based on data analytics conducted on real data obtained from running cluster nodes in a real physical DC. For the purpose of this work, approximately 20 months’ worth of data has been collected. Data collected are related to: relevant parameters for each node (e.g. inlet air temperature, internal temperature of each node, energy consumption of CPU, RAM, memory, etc...); environmental parameters (e.g. air temperatures and humidity in both the hot and cold aisles); cooling system related parameters (e.g. fan speed); and finally, individual users who submit their jobs to cluster node. This research focuses on the effect of dynamic workload assignment on energy consumption and performance of both the computing and cooling systems. The constraint is that each arrived job must be assigned irrevocably to a particular server without any information about impending incoming jobs. Once the job has been assigned, no pre-emption or migration is allowed, which a rule is typically adhered to for HPC applications due to high data reallocation incurred costs. In this research, we

particularly explore an optimised mapping of nodes that have to be physically and statically placed in advance to one of the available rack slots in the DC. This will form a matrix comprising computing units with specific characteristics and certain resource availability level at a given time t . The goal is to create a list of candidate nodes to deliver “calculation performance” required by a user’s job. When choosing the candidate nodes, the job-scheduler will evaluate the suitability of the thermally cooler nodes (which at the instant t) based on their capability to satisfy the calculation requested by a user (in order to satisfy user’s SLA). To enhance the job scheduler decision making, it is essential to know in advance, the type of job a user will submit to a node(s) for computation. Such insight is provided by several years’ worth of historical data and advanced data analytics using machine learning algorithms. Through Platform Load Sharing Facility (LSF) accounting data we code user profiles into 4 macro-categories:

1. *CPU_intensive*
2. *MEMORY_intensive*
3. *CPU&MEMORY_intensive*
4. *CPU&MEMORY_not intensive*

This behavioural categorisation provides an opportunity to save energy and better allocate tasks to cluster nodes to reduce overall node temperatures. Additionally, when job allocation is evenly distributed, thermal hotspots and cold spots could be avoided. The temperatures of the calculation nodes could be evened out, thus, resulting in a more even distribution of heat across the cluster.

3.2 Users and workload understanding profiled log

Based on thermal data, it is necessary to better understand in-depth what users do and how they manage to solicit the calculation nodes for their jobs. The three main objectives of understanding users’ behaviour are as follows: Identify parameters based on the diversity of submitted jobs for user profiling; Analyse the predictability of various resources (e.g. CPU, Memory, I/O) and identify their time-based usage patterns; Build predictive models for estimating future CPU and memory usage based on historical data carried out in the LSF platform. Abstraction of behavioural patterns in the job submission and its associated resource consumption is necessary to predict future resource requirements. This is exceptionally vital for dynamic resource provisioning in a DC. User profile is created based on submitted job-related information and to reiterate, the 4 macro categories of user profiles are: 1) CPU-intensive, 2) disk-intensive, 3) both CPU and MEMORY- intensive, or 4) neither CPU- nor MEMORY-intensive. A crosstab of the accounting data (provided by the LSF platform) and resource consumption data help guide the calculation of relevant thresholds that code jobs into several distinct utilisation categories. For instance, if the CPU load is high (e.g., larger than 90%) during almost 60% of the job running time for an application, then the job can be labelled as a CPU-intensive one. The goal is for the job-scheduler to optimise task scheduling when a job with the same AppID (i.e. the same type of job) or same username is re-submitted to a cluster. In case of a match with the previous AppID or username, relevant utilisation stats from the profiled log are retrieved. Based on the utilisation patterns, this particular user/application will be placed into one of the 4 previously discussed categories. Once a job is categorised, a thermally suitable node is selected to satisfy the task calculation requirements. A task

with high CPU and memory requirement will not be immediately processed until the node temperature is well under a safe temperature threshold. Node temperature refers to the difference between the node's outlet exhaust air and inlet air temperatures (note: this generally corresponds to the air temperature in the aisles cooled by the air conditioners).

3.3 Real-time workload management based on Thermal awareness: Cluster evaluation

It is necessary to have a snapshot of relevant thermal parameters (e.g. temperatures of each component in the calculation nodes) for each cluster to facilitate efficient job allocation by the job-scheduler. Generally, a snapshot is obtained through direct interrogation of the nodes and installed sensors in their vicinity, or inside the calculation nodes. For each individual node, the temperatures of the CPUs, memories, instantaneous energy consumption and speed of the cooling fans are evaluated undeniably, the highly prioritised parameter is the difference between the node's inlet and exhaust air temperatures. If there is a marked difference, it is evident that the node is very busy (with jobs that require a lot of CPU or memory-related resource consumption). Therefore, for each calculation node, relevant data is monitored in real time, and subsequently, virtually stored in a matrix that represents the state of the entire cluster. Each matrix cell represents the states of a node (represented by relevant parameters). For new job allocation, the scheduling algorithm will choose a node based on its states depicted in the matrix (e.g. recency or Euclidean distance). Through this, generated waste heat is evenly distributed over the entire "matrix" of calculation nodes so that hotspots could be significantly reduced. Additionally, a user profile is an equally important criterion for resource allocation. This is due to the fact that user profiles provide insights into user consumption patterns and the type of submitted jobs and their associated parameters. For example, if we know that a user will perform CPU-intensive jobs for 24 hours, we will allocate the job in a "cell" (calculation node) or a group of cells (when the number of resources requires many calculation nodes) that are physically well distributed or with antipodal locations. This selection strategy aims to evenly spread out the high-density nodes followed by the necessary cooling needs. This will help minimise DC hotspots and ascertain efficient cooling with reduction in cooling-related energy consumption.

4 Results and Discussion

As previously discussed, we have created user profiles based on submitted job-related information. Undeniably, these profiles are dynamic because they are constantly revised based on user resource consumption behaviour. For example, a user may have been classified as "CPU intensive" for a certain time period. However, if the user's submitted jobs are no longer CPU intensive, then the user will be re-categorised. The deployment of the thermal-aware job scheduler generally aims to reduce the overall CPU/memory temperatures, and outlet temperatures of cluster nodes. The following

design principles guide the design and implementation of the job: 1) Job categories - assign an incoming job to one of these 4 categories: CPU-intensive, memory-intensive, neither CPU nor memory-intensive, and both CPU and memory-intensive tasks; 2) Utilisation monitoring - monitoring CPU and memory utilisation while making scheduling decisions; 3) Redline temperature control - ensure operating CPUs and memory under threshold temperatures; 4) Average temperatures maintenance - monitor average CPU and memory temperatures in a node and manage an average exhaust air temperature across a cluster. To reiterate, user profile categorisation is facilitated by maintaining a log profile of both CPU and memory utilisation for every job (with an associated user) processed in the cluster. A log file contains the following user-related information: (1) user ID; (2) Application identification; (3) the number of submitted jobs; (4) CPU utilisation; (5) memory utilisation.

4.1 Data Analysis: Results and Discussion

A list of important thermal management-related terms is as follows: 1) CPU-intensive - applications that is computation intensive (i.e. requires a lot of processing time); 2) Memory-intensive-a significant portion of these applications require RAM processing and disk operations; 3) Maximum (redline) temperature - the maximum operating temperature specified by a device manufacturer or a system administrator; 4) Inlet air temperature - the temperature of the air flowing into a data node (i.e. temperature of the air sucked in from the front of the node); 5) Exhaust air temperature - the temperature of the air coming out from a node (the temperature of the air extracted from the rear of the node). By applying these evaluation criteria, we have built an automated procedure that provides insight into the 4 user associated categories (based on present and historical data). Obviously, the algorithm always makes a comparison between a job just submitted by a user and the time series (if any) of the same user. If the application launched or the type of submitted job remains the same, then the user will be grouped into one of the 4 categories (based on a supervised learning algorithm) During each job execution, the temperature variations of the CPUs and memories are recorded at pre-established time intervals. Finally, it continuously refines the user behaviour based on the average length of time the user uses for the job. This will provide a more accurate user (and job) profile because it provides reliable information on the type of job processed in a calculation node and its total processing time. The job scheduler will exploit such information for better job placement within an ideal array of calculation nodes in the cluster. A preliminary study is conducted. To provide insight into the functioning of the clusters. For 8 months, we have observed the power consumption (figure 1) and temperature (figure 2) profiles of the nodes with workloads. We have depicted energy consumed by the various server components (CPU, memory, other) in figure 3 and presented a graph that highlights the difference in energy consumption between idle and active nodes (figure 4).

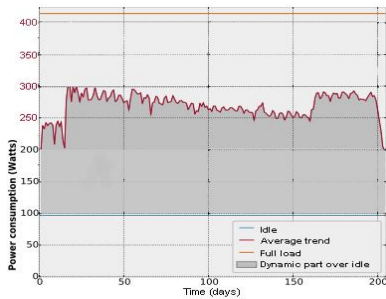


Figure 1. The representative shape of Power profile portion on average for all available nodes. Power consumption dataset for a subset of 200 days.

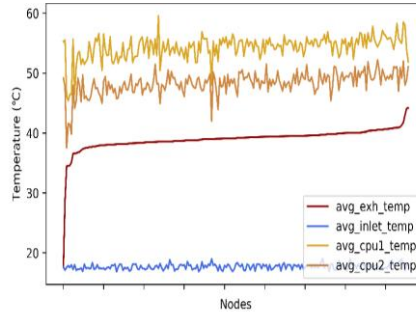


Figure 2. Temperature profiles (subset of 1 month) on average for all available nodes. Nodes are sorted in the order of exhaust air temperature increase.

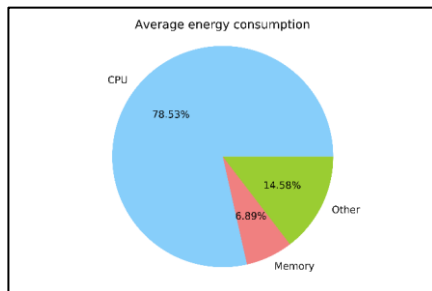


Figure 3. Average energy partitioning for all nodes of cluster CRESCO6

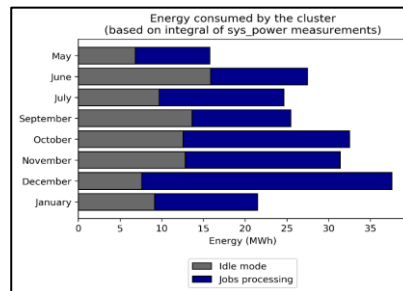
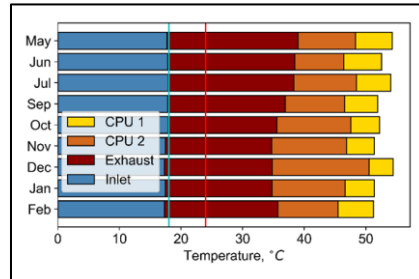


Figure 4. Energy consumption in idle and active nodes (subset of 8 months)

It is observed that for each node, an increase in load effects an increase in temperature difference between inlet and exhaust air for that particular node. Figure 5 depicts the average observed inlet air temperature (blue segment, and in the cold aisle), and exhaust air temperature at their rear side) amaranth segment, in the hot aisle). Note the temperature measurements are also taken two CPUs adjacent to every node. The setpoints of the cooling system are approximately 18°C at the output and 24°C at the input of the cooling system – as respectively shown in Figure. 5 as blue and red vertical lines. However, it appears that the lower setpoint is variable (supply air at 15-18°C) while the higher setpoint varies from 24-26°C. As observed from the graph, the cold aisle maintains the setpoint temperature at the inlet of the node, which affirms the efficient design of the cold aisle (i.e. due to the use of plastic panels to isolating the cold aisle from other spaces in the IT room). However, the exhaust air temperature has registered on average, 10°C higher level than the hot aisle setpoint. Notably, exhaust temperature sensors are directly located at the rear of the node (i.e. in the hottest parts of the hot aisle).

Figure 5. Temperature observed on average in all nodes during 9 months with vertical lines corresponding to cold and hot aisle setpoints.

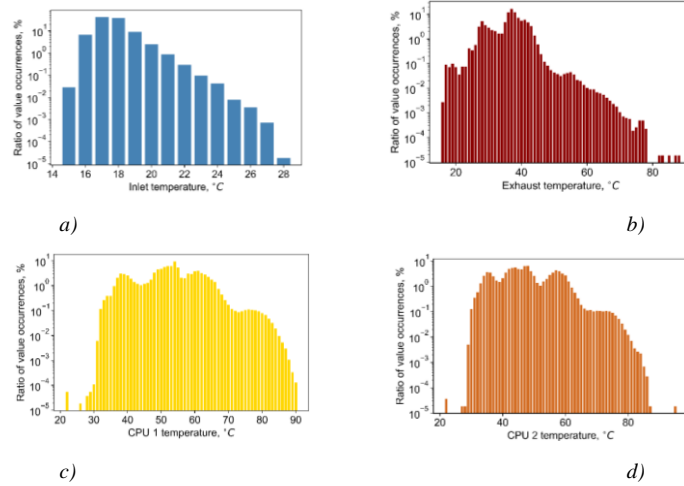


Therefore, it is observed that hotspots are immediately located at the back of server racks, while the hot aisle air is cooled down to the 24-26°C. This is due the cooling system at the CRAC (computer room air conditioning) which results in hot air intake, air circulation and cold-hot air mix in the hot aisle. Meanwhile, the previously mentioned temperature difference of 10°C between the hotspots and the ambient temperature unravels the cooling system weak points because it could not directly cool the hotspots. In the long term, the constant presence of the hotspots might affect the servers' performance (i.e. thermal degradation) which should be carefully addressed by the DC operator. Remarkably, although the hotspots are present at the rear of the nodes, the cooling system does cool temperatures around the nodes. Cold air flows through the node and is measured at the inlet, then at CPU 2 and CPU 1 locations (directly on the CPUs) and finally, at the exhaust point of the server. The differences between observed temperature ranges in these locations are averaged for all the nodes. An investigation on the observed temperature distribution contributes to the overall understanding of the thermal characteristics, as it provides an overview of the prevailing temperatures shown in Figure 5 and Figure 6. For every type of thermal sensors, the temperature values are recorded as an integer number, so the percentage of occurrences of each value is calculated. The inlet air temperature is registered around 18°C in the majority of cases and has risen up to 28°C in around 0.0001% of cases. It could be concluded that the cold aisle temperature remains around the 15-18°C setpoint for most of the monitored period. Ranges of the exhaust temperature and those of CPUs 1 and 2 are in the range 20-60°C with most frequently monitored values in the intervals of 18-50°C. Although these observations might incur measurement errors, they reveal servers that are at risks of frequent overheating when benchmarked with manufacturer's recommendation data sheets.

Additionally, this study focuses on variation between subsequent thermal measurements with the aim of exploring temperature stability around the nodes. All temperature types have distinct peaks of zero variation which decreases symmetrically and assumes a Gaussian distribution. It could be concluded that temperature tends to be stable in the majority of monitored cases. However, the graphs for exhaust and CPUs 1 and 2 temperature variation (Figure 6 reveal that less than 0.001% of the recorded measurements show an amplitude of air temperature changes of 20°C or more occurring at corresponding locations.

Sudden infrequent temperature fluctuations are less dangerous compared to prolonged periods of constantly high temperatures. Nevertheless, further investigation is needed to uncover causes of abrupt temperature changes so that appropriate measures could

Figure 6. Distribution of monitored temperature values taken for all nodes and months.



be undertaken by DC operators to maintain prolonged periods of constantly favourable conditions. We propose a scheduler upgrade which aims to optimise CPU and memories-related resource allocation, as well as exhaust air temperatures without relying on profile information. Prescribed targets for the proposed job scheduler are shown in Table 2.

Table 2. Schema with prefixed target for improved job scheduler.

	Proposed job scheduler
Strategy	Schedules job based on utilisation and temperature information gathered at run-time
Job Assignment	Assigns a job to the coolest node in a cluster at any point in time
Job Scheduling	Schedules a job on the coolest node in a cluster
Temperature control	Maintains uniform temperate across a cluster
Node Activity	At least 50% are active nodes at any given time in a cluster
Pros	Works better with a large cluster
Cons	Overhead of communication of temperature and utilisation information

The design of the proposed job schedule ought to address four issues: 1) Differentiate between CPU-intensive tasks and memory-intensive tasks; 2) Consider CPU and memory utilisation during the scheduling process; 3) Maintain CPU and memory temperatures under the threshold redline temperatures; 4) Minimise the average exhaust air temperature of nodes to reduce cooling cost. The job scheduler receives feedback of node status through queried *Confluent* platform [15] (monitoring software installed on each node). When all the nodes are busy, the job scheduler will manage the temperatures, embarks on a load balancing procedure by keeping track of the coolest nodes in the cluster. In doing so, the scheduler continues job executions even in hot yet undamaging conditions. The job scheduler maintains the average cluster

CPU and memory utilisation represented by $U_{\{CPUavg\}}$ and $U_{\{MEMavg\}}$, CPU and memory temperatures represented by $T_{\{CPUavg\}}$, $T_{\{MEMavg\}}$, respectively. The goal of our enhanced job scheduler is to maximise the COP (coefficient of performance). Below are the 7 constraints (at nodes level) for our enhanced scheduler:

1. check constraint $T_{CPU}^i (instant\ i) < T_{CPUAvg}$
2. otherwise, check constraint $T_{Mem}^i < T_{Memavg}$
3. $T_{Memavg} < T_{MemMax}$ and $T_{CPUavg} < T_{CPUMax}$
4. $T_{out}^i \leq (\sum_{i=1}^N T_{out}) / N$
5. Each job is assigned to utmost one node
6. Minimise response time of job

With the first and second constraints are satisfied, ensure that the memory and CPU temperatures remain below the threshold temperatures. If a cluster's nodes exceed the redline threshold, then optimise the temperature by assigning jobs to the coolest node in the cluster. The third constraint specifies that if the average temperature of memory or CPU rises above the maximum temperature, then the scheduler should stop scheduling tasks as it might encounter hardware failures. The fourth constraint states that the exhaust air temperature of a node should be the same or less than the average exhaust air temperature of the cluster (taking into consideration N number of nodes). The fifth constraint ensures that a node gets utmost one job at a single point in time. The last point aims at reducing the completion time of a job to achieve optimal performance. The following is the description of our algorithm:

```

****matrix of node with position r-ow and c-column****
Cluster= matrix[r,c]
user=getUSERfromSubmittedJob_in_LSF
Jobtype= getJobProfile(user)

****push the values of utilization and temperature for cpu and memory into matrix****
for (i=0; i=number_of_node;i++) do
    nodename = getnodeName(i)
    Ucpu = getCPU_Utilization(nodename)
    Umemory = getMEMORY_Utilization(nodename)
    Tcpu = getCPU_Temperature(nodename)
    Tmemory = getMEMORY_Temperature(nodename)
End for

*****if a user is not profiled *****
if Jobtype= null then
    *****try to understand job type at run time*****
    if (Ucpu <= Uthreshold_cpu && (Umemory <= Uthreshold_memory) then
        Jobtype=easyJob
    else if (Ucpu>Uthreshold_cpu && (Umemory < Uthreshold_memory) then
        Jobtype=CPUintensiveJob
    else if (Ucpu<Uthreshold_cpu && (Umemory > Uthreshold_memory) then
        Jobtype=MEMORYintensiveJob
    else
        Jobtype=CPU&MEMORYintensiveJob
    end if
end if

***** I try to find the candidate nodes for each type of job*****
avgTempCluster= avgTemp(Cluster)
minT_nodename= getTempNodename(minTemp(Cluster))
maxT_nodename=getTempNodename(maxTemp(Cluster))

*****intervals of temperatures for candidate nodes*****

```

```

bestCPUIntensiveNode=getNode (minT_nodename, minT_nodename+25%)
bestMEMORYIntensiveNode= getNode(minT_nodename+50%, minT_nodename+75%)
bestCPU&MEMORYIntensiveNode= getNode(minT_nodename+25%, minT_nodename+50%)
bestEasyJob= getNode(maxT_nodename, maxT_nodename-25%)

```

```

*****job assignments*****
if Jobtype= CPUintensiveJob then
    assignJob (bestCPUIntensiveNode)
else if Jobtype= MEMORYintensiveJob then
    assignJob (bestMemoryIntensiveNode)
else if Jobtype= CPU&MEMORYintensiveJob then
    assignJob(bestCPU&MEMORYIntensiveNode)
else
    assignJob(bestEasyJob)
end if

```

The algorithm feeds into the node matrix by considering the physical arrangement of every single node inside the racks. Firstly, obtain the profile of the user who puts in a resource request for resources. This is done by retrieving the user's profile from a list of stored profiles. The algorithm is executed for all the nodes to appreciate resource utilisation level and temperature profiles each node. If the user profile does not exist, then when a user executes a job for the first time, the algorithm calculates a profile instantaneously. All the indicated threshold values are operating values calculated for each cluster configuration and are periodically recalculated and revised according to the use of the cluster nodes. Subsequently, some temperature calculations are made from the current state of the cluster (through a snapshot of thermal profile). Finally, the last step is to assign the job to the node based on the expected type of job. Through this, the algorithm helps avert the emergence of hotspots and cold spots by uniformly distributing the jobs in the cluster.

5. Conclusions and Future Work

In order to support sustainable development goals, energy efficiency ought to be the ultimate goal for a DC with a sizeable high-performance computing facility. To reiterate, this work primarily focuses on two of major aspects: IT equipment energy productivity and thermal characteristics of an IT room and its infrastructure. The findings of this research are based on the analysis of available monitored thermal characteristics-related data for CRESCO6. These findings feed into recommendations for enhanced thermal design and load management. In this research, clustering performed on big datasets for CRESCO6 IT room temperature measurements, has grouped nodes into clusters based on their thermal ranges followed by uncovering the clusters they frequently subsume during the observation period. Additionally, a data mining algorithm has been employed to locate the hotspots and approximately 8% of the nodes have been frequently placed in the hot range category (thus labelled as hotspots). Several measures to mitigate risks associated with the issue of hotspots have been recommended: more efficient directional cooling, load management, and continuous monitoring of the IT room thermal conditions. This research brings about two positive effects in terms of DC energy efficiency. Firstly, being a thermal design pitfall, hotspots pose as a risk of local overheating and servers thermal degradation due to prolonged exposure to high temperatures. Undeniably, information of hotspots

localisation could facilitate better thermal management of the IT room where waste heat is evenly distributed. Thus, it ought to be the focus of enhanced thermal management in the future. Secondly, we discussed ways to avert hotspots through thermal-aware resource allocation (i.e. select the coolest node for a new incoming job), and selection of nodes (for a particular job) that are physically distributed throughout the IT room.

References

1. J. Koomey: Growth in Data Center Electricity use 2005 to 2010. Analytics Press., 1–24. <https://doi.org/10.1088/1748-9326/3/3/034008>, 2011.
2. Greenpeace: How Dirty Is Your Data? A Look at the Energy Choices That Power Cloud Computing, 2011.
3. V. D. Reddy, et al: Metrics for Sustainable Data Centers, *IEEE Trans. Sustain. Comput.*, vol. 2, no. 3, pp. 290–303, Jul. 2017.
4. J Kong, S. W. Chung, k. Skadron: Recent thermal management techniques for microprocessors. *ACM Computing Surveys*, vol. 44, no. 3. <https://doi.org/10.1145/2187671.2187675>, 2012.
5. L. Parolini, B. Sinopoli, B. H. Krogh, Z. Wang: A Cyber-Physical Systems Approach to Data Center Modeling and Control for Energy Efficiency. *Proceedings of the IEEE*. 100. 254-268. 10.1109/JPROC.2011.2161244, 2012.
6. L. Wang, et al: Thermal aware workload placement with task-temperature profiles in a datacenter. *J. Supercomput.* 2012, 61, 780–803, doi:10.1007/s11227-011-0635-z, 2012.
7. ASHRAE Technical Committee 9.9, “Thermal Guidelines for Data Processing Environments – Expanded Data Center Classes and Usage Guidance,” 2011.
8. X. Jin, F. Zhang, A. V. Vasilakos, and Z. Liu: Green Data Centers: A Survey, Perspectives, and Future Directions,” *arXiv*, vol. 1608, no. 00687, 2016.
9. M. Chinnici, A. Capozzoli, and G. Serale: Measuring energy efficiency in data centers. In *Pervasive Computing Next Generation Platforms for Intelligent Data Collection*, Chapter 10, pp. 299-351, 2016.
10. M. Chinnici, et al: Data Center, a Cyber-Physical System: Improving Energy Efficiency Through the Power Management,” in *2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf (DASC/PiCom/DataCom/CyberSciTech)*, 2017, pp. 269–272.
11. S. Yeo, et al.: ATAC: Ambient Temperature-Aware Capping for Power Efficient Datacenters, in *Proceedings of the 5th ACM Symposium on Cloud Computing*, Seattle, WA, USA, doi:10.1145/2670979.2670966, 2014.
12. A. Capozzoli, et al: Thermal Metrics for Data Centers: A Critical Review,” *Energy Procedia*, vol. 62, pp. 391–400, Jan. 2014.
13. A. Capozzoli, et al: Review on performance metrics for energy efficiency in data center: The role of thermal management,” *LNC*, vol. 8945, pp. 135–151, 2015.
14. Vxchnge site, <https://www.vxchnge.com/>, last accessed 2020/3/27.
15. Confluent site, <https://docs.confluent.io/platform.html>, last accessed 2020/3/27.
16. T. Van Damme, C. De Persis and P. Tesi: Optimized Thermal-Aware Job Scheduling and Control of Data Centers, in *IEEE Transactions on Control Systems Technology*, vol. 27, no. 2, pp. 760-771, March 2019, doi: 10.1109/TCST.2017.2783366
17. G. Varsamopoulos, et. al.: Energy Efficiency of Thermal-Aware Job Scheduling Algorithms under Various Cooling Models, In S. Ranka et al. (Eds.): *IC3 2009, Communications in Computer and Information Science*, vol 40. Springer, Berlin, Heidelberg, pp. 568–580, 2009, doi: https://doi.org/10.1007/978-3-642-03547-0_54