2017 IEEE 24th International Conference on High Performance Computing Workshops (HiPCW)

Thermal Profiling and Modeling of Hadoop Clusters using BigData Applications

Shubbhi Taneja, *Student Member, IEEE*, Ajit Chavan, *Student Member, IEEE*,
Yi Zhou, *Student Member, IEEE*, Mohammed Alghamdi, *Senior Member, IEEE*, Xiao Qin, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a thermal model called *tModel*, which projects outlet temperatures from inlet temperatures as well as directly measured multicore temperatures rather than deploying a utilization model. We perform extensive experimentation by varying applications types, their input data sizes, and cluster sizes. We collect inlet, outlet, and multicore temperatures of cluster nodes running a group of big-data applications. The proposed thermal model estimates the outlet air temperature of the nodes to predict cooling costs. We validate the accuracy of our model against data gathered by thermal sensors in our cluster. Our results demonstrate that *tModel* estimates outlet temperatures of the cluster nodes with much higher accuracy over CPU-utilization based models. We further show that *tModel* is conducive of estimating the cooling cost of data centers using the predicted outlet temperatures.**

***Keywords*-thermal profiling; thermal model; benchmarking; MapReduce applications; HPC clusters; BigData; multicore architecture; distributed computing; Hadoop.**

I. Introduction

Growing number of data centers are being deployed across the world in last decade. According to the EPA report, US data centers consumed approximately 2% of total U.S. energy consumption, with 24% increase in the last five years [8]. This continuous increase in energy consumption by the data centers is an effect of rapidly growing demand of computation and storage capacity. This demand has also caused increase in the power density of the data centers. As the power density of the data center grows, managing thermal emergancies in the data center bacomes one of the vital issues. Gartner predicts that by 2021, more than 90% of the data centers would have to revise their thermal management strategies [4].

Thermal profile of the data center has a significant impact on the cooling cost of the data centers. As mentioned earlier, thermal models are usually used to characterize the thermal profile of the data center. Number of thermal models have been proposed in the past to calculate the inlet temperature of the nodes in the data centers and its impact on the thermal profile and cooling cost of the data center. Most of the thermal models depends on the processor utilization to calculate the inlet and outlet temperature of the nodes. In this study, we specifically investigate the impact of MapReduce workload on the outlet temperature of the nodes. We first proposed the thermal model to calculate the outlet temperature of the nodes in the data center. With thermal model in place, we further investigate the thermal impact of the MapReduce workload on data center.

The four factors below make our thermal model indispensable for Hadoop clusters:

- excessively high energy cost of large-scale clusters,
- pressing needs of reducing thermal monitoring cost,
- cooling cost estimation of HPC clusters, and
- vital impacts of system utilization on multicore processors and disks of a Hadoop cluster.

With an dramatic increase in energy consumption of large-scale cluster computing systems, we have to urgently address the energy efficiency issues. Strong evidence indicates that cooling cost is a significant contributor of a cluster's operational cost [1][5]. For example, the power and cooling infrastructure supporting IT equipment can consume up to 50% of energy in a data center [1]. Prior studies confirmed that cutting cooling cost effectively improves the energy efficiency of data centers [11][21]. For example, intriguing workload placement strategies were implemented to balance temperature distribution [11][21].

One way to monitor the cluster's thermal behavior is to set up temperature sensors to monitor the inlet and outlet temperatures of the nodes in the data center. Although deploying sensors is feasible for small clusters, it is an expensive and tedious solution for large data centers housing thousands of servers. Sensors are also subjected to failures, which may lead to invalid or faulty data collection. Therefore, it is important to use an inexpensive models to estimate the inlet and outlet temperature of the nodes in the data centers.

Minimizing the cooling cost of the data center is one of the critical desing issue in deploying data centers. Need to continuously monitor the thermal behavior of data center to anticipate the thermal emergencies and proactive thermal management to avoid such emergencies is very important in order to achieve high availability and avoid hardware failures. The thermal model proposed in this paper facilitates monitoring and estimating the thermal behavior of the nodes in the data center.

Our overall goal is to investigate a thermal modeling approach to estimating outlet temperatures of server nodes on Hadoop cluster without relying on the power consumption models of the nodes. It is noteworthy that conventional thermal

S. Taneja⋆, A. Chavan, and Y. Zhou are doctoral students at Department of Computer Science and Software Engineering in Auburn University, Auburn, Alabama 36849-5347. E-Mail: apc0013@auburn.edu, yzz0074@auburn.edu
⋆corresponding author: shubbhi@auburn.edu

M. Alghamdi is with the Department of Computer Science, Al-Baha University, Al-Baha City, Kingdom of Saudi Arabia. E-mail: mialmushilah@bu.edu.sa.

X. Qin is with the Department of Computer Science and Software Engineering, Shelby Center for Engineering Technology, Samuel Ginn College of Engineering, Auburn University, AL 36849-5347. E-mail: xqin@auburn.edu.

0-7695-6331-7/17/$31.00 ©2017 IEEE
DOI 10.1109/HiPCW.2017.00012

IEEE computer society

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

models are built on top of power consumption models. We intend to eliminate the power consumption models as a middle man in the existing thermal models.

There are three contributions in this work. First, we conduct a thermal profiling study of a cluster on which various Hadoop applications are investigated. When the applications process various data size under different number of nodes, we profile our server nodes to monitor multicore CPU temperatures. Second, we build a thermal model to estimate temperatures using inlet temperatures and CPU core temperatures. Third, as a use case of our model, we extend the model to predict the cooling cost of a Hadoop cluster.

The rest of this paper is organized as follows. In Section II, we discussed some of the similar research done in the past. In Section III, we discuss our system framework and discuss an existing thermal model. Section IV introduces our proposed model's design and an essential modeling parameter, *K*. Section V provides the thermal profiling evaluations. Finally, Section VI concludes our study by summarizing our findings.

II. Related work

Our work is related to three areas of interest, namely, energy-efficient computing, thermal management in data centers and importance of thermal models.

A. Energy-Efficient Computing Systems

An increasing number of energy-saving techniques have been designed to reduce the energy costs of data centers. For example, a few common strategies adopted by modern data centers include raising supplied air temperatures set points, using atmospheric air directly for cooling, turning off idle-servers, and consolidating workloads to minimize the number of active nodes at a given time.

One way to improve energy efficiency is to develop thermal-aware job scheduling mechanisms (see, for example, [3], [14], [2], [18], [17]). Alsubaihi and Gaudiot developed *PETS* (i.e., Performance, Energy and Thermal-Aware Scheduler) that considers various scheduling constraints like job scheduling, core scaling, and thread allocation. PETS is able to improve execution time and energy consumption under peak power [2]. Cao *et al.* proposed a cooling strategy optimizing job-to-node mapping and reducing hotspot temperature by allocating power-hungry jobs to compute nodes [3]. Varsamopoulos [20] devised a cooling model applied in thermal-aware job scheduling algorithms that profiles the behavior of a CRAC system, thereby making estimation of jobs' power consumption.

Unlike the above models, *tModel* is conducive to saving energy by predicting outlet temperatures of the server nodes running big-data applications. After the initial profiling process is accomplished, the outlet temperature of each of these nodes can be predicted with very high accuracy using CPU temperatures instead of power consumption and CPU utilization.

B. Data Center Thermal Management

Traditionally, reactive thermal management strategies have been employed in data centers to deal with thermal emergencies. In particular, *CFD* (i.e., Computational Fluids Dynamics) simulations that have been used to evaluate thermal performance of data centers with a given configuration. A downside of CFD simulations is the elongated time to obtain these simulation results. Since thermal emergencies should be dealt in a timely fashion to safely operate data centers, researchers have continuously looking into improving thermal management.

Tang *et al.* developed a heat flow model that accelerates the thermal prediction as opposed to CFD to characterize hot air recirculation [19]. Tang's heat flow model relies on the use of sensors for gathering temperature data for the simulations-based study. To manage thermal emergencies without significant performance degradation, Ramoz and Bianchini created an online future temperature prediction framework by combining DVFS, request distribution, and request-admission control for Internet services [15].

Our study is different from the prior studies in that our findings suggest that outlet air temperatures can be modeled as a function of multi-core temperatures and inlet temperatures without necessarily relying on CPU utilization. Utilization-based thermal models may introduce errors due to inaccurate mappings from system utilization to outlet temperatures. Our model addresses this concern in the existing models by eliminating utilization models as a middle man.

C. Importance of Thermal Modeling

Like performance modeling that captures irregular resource usage patterns in clusters, thermal modeling is beneficial to various energy-optimizing tasks. Thermal models are also deployed in energy-aware control algorithms for identifying cooling-efficient zones for computationally intensive workloads. A few schemes (see, for example, [9]) offer system performance analysis by modeling system behaviors. Thermal profiling was employed to examine an array of best practices like air management, optimizing the size of data centers, and utilizing free cooling by using chilled water.

A handful of thermal models tailored for thermal management in data centers have been proposed in the past decade [15][10][9]. For instance, Li *et. al* conducted a preliminary study on load distribution techniques, applying an analytical model for minimizing computing and cooling energy collectively [10]. Parolini *et al.* presented a cyber-physical system that takes advantage of thermal-aware file placement for data centers [13]. Similarly, Kaushik and Nahrstedt [7] investigated proactive, thermal-aware placement, which saves cooling costs without performance degradation.

In a few studies, processor variation was incorporated to create effective thermal models and schedulers. For example, owing to the variation in processor power efficiency, Rountree *et al.* [16] studied processor performance under power clamping; their study demonstrated that a power bound converts variation in processor power to variation in performance.

None of the aforementioned models take CPU core temperatures in consideration for prediction outlet temperatures. Our tModel aims to predict outlet air temperatures to estimate cooling costs for CRACs in data centers. In the tModel

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

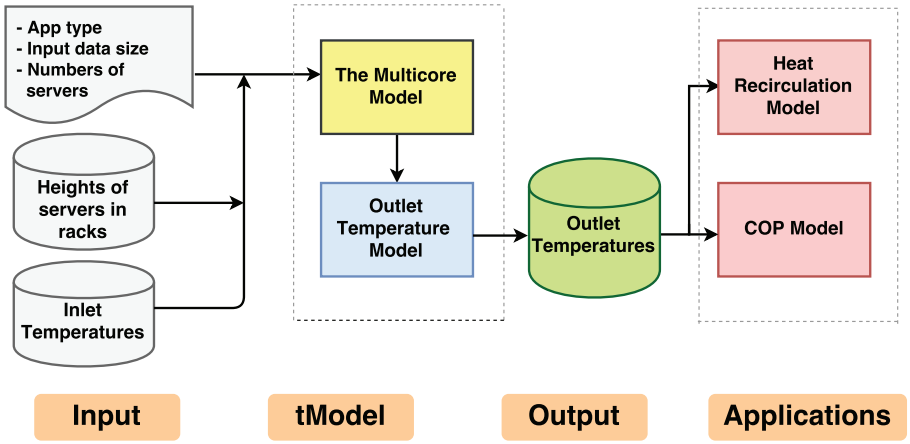

Fig. 1: The system framework of our thermal models.

project, we put thermal modeling of Hadoop clusters under a microscope. In particular, we investigate how the workload would have impact on the temperatures of multiple cores in Hadoop applications, which in turn affect server outlet temperatures.

III. The System Framework

There is a growing demand to build thermal-prediction models for high-performance clusters. We design a systematically approach to developing thermal models for Hadoop applications deployed in data centers. The thermal model proposed in this study immediately benefits usage cases. First, outlet temperatures of server blades are applied to estimate cooling cost [18], which depends on outlet temperatures and heat dissipated by the servers. Second, temperatures captured in our thermal model paves a way to investigate a heat re-circulation model [12] [19], which is affected by cross-interference coefficients.

We start this section by delineating a modeling framework (see Section III-A. Then, we develop a baseline thermal model to estimate outlet temperatures of Hadoop clusters (see Section III-B).

A. The Modeling Framework

Fig. 1 outlines our thermal-modeling framework, which consists of two key components: the outlet-temperature model and the multicore model.

The input data of *tModel* include application profiles (e.g., application type, data size), heights of servers in racks, and inlet temperatures. It is worth noting that all these input parameters are directly fed into the multicore model which generates temperatures of multicore processors. The multicore-processor temperatures are assimilated into the outlet temperature model (see Section IV) to estimate outlet temperatures of any given computing node. Thus, the outlet-temperature model builds up the correlations between inlet and outlet temperatures by incorporating multicore temperatures residing in computing nodes.

Recall that tModel relies on inlet temperatures monitored by sensors, which tend be expensive for large-scale data centers. To reduce such hefty cost, we only install temperature sensors on a few selected blade servers (a.k.a., nodes) in each rack (e.g., 4 out of 16). We also may use the inlet temperatures measured on one rack to speculate inlet temperatures of nodes mounted on another rack.

Outlet temperatures produced by tModel benefits applications like cooling cost models (i.e., the COP model) and heat recirculation models. The COP model computes cooling costs by taking into account outlet temperatures offered by the outlet-temperature model in tModel. The main strengths of the tModel framework are three-fold. First, temperatures of multicore processors are modeled as a function of system configurations and application access patterns. Second, a thermal model characterizes the relationship among inlet and outlet temperatures as well as multicore processor temperatures. Third, tModel opens an avenue for data center designers to investigate heat recirculation and cooling cost.

B. A Baseline Thermal Model

To predict outlet temperatures for nodes in a Hadoop cluster, we construct a baseline thermal model speculating outlet temperatures derived from inlet temperatures, multicore temperatures, input data size, and the height of servers placed in racks. In a data center equipped with cooling systems, the inlet temperatures of the servers are affected by the supplied room temperature mixed with recirculated hot air. Given a set of nodes, we denote T_i^{in} as the inlet temperature of the ith node. It is worth noting that T_i^{in} can be directly and accurately monitored by a temperature sensor. Alternatively, T_i^{in} can be projected using monitored temperatures from other neighboring servers.

Our baseline thermal model is inspired by the *T** thermal model proposed by Kaushik *et al.* [7]. The T* model predicts servers' outlet temperatures to estimate the cooling cost for a data center.

Let $T_{i,t}^{out}$ be the outlet temperature of the ith server at a given time t. $T_{i,t}^{out}$ depends on (a) server i's power consumption $P_{i,t}$ at time t, (b) server temperature $T_{i,t}$, and (c) heat

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

exchange rate θ_i. Thus, we have

$$T_{i,t}^{out} = T_{i,t} - \frac{P_{i,t}}{\theta_i}. \quad (1)$$

For simplicity, we consider in this study statistical temperature measures rather than temperature in a specific time. After removing t from (1), we obtain

$$T_i^{out} = T_i - \frac{P_i}{\theta_i}, \quad (2)$$

where P_i is server i's average power consumption in a given monitoring period. Since utilization-based thermal models may introduce errors due to inaccurate mappings from system utilization to outlet temperatures.

IV. *t*Model: Modeling Outlet Temperatures

In this section, we develop a thermal model that aims at predicting outlet temperatures by considering the impacts of inlet air temperature and CPU core temperatures on the outlet temperatures (see Section IV-A). Next, we discuss the role of our multicore model in predicting outlet air temperatures . We also demonstrate a sample usage of *tModel* in Section IV-B).

A. A Simple Yet Efficient Model

The model proposed in Section III-B relies on on power consumption and other factors. In this part of the study, we intend to construct a simple yet effective thermal model to predict outlet temperatures without explicitly using power consumption. We show in Section V-C that the simplified model described in this section is more accurate than the complicated baseline model.

The goal of our simplified thermal model is to predict outlet temperatures for Hadoop cluster nodes from (a) inlet temperatures and (b) multicore temperatures without relying on any other parameters. Most existing thermal models make use of CPU utilization to estimate power consumption, which in turn facilitates outlet-temperature predictions. Such utilization-based thermal models may introduce errors due to inaccurate mappings from system utilization to outlet temperatures. To address this concern in the existing models, we eliminate utilization models as middle men from the thermal model. Rather than deploying a utilization model, our tModel projects outlet temperatures from inlet temperatures as well as directly measured multicore temperatures by leveraging on-chip sensors. Again, to cut cost of maintaining an excessive number of sensors monitoring inlet temperatures, we mount sensors on a small percentage of nodes; inlet temperatures of the other nodes can be modeled from their neighboring nodes due to the fact that the heat measured at each sensor is the combination of heat generated by the workload and the heat from the ambient air at the server's inlet air grill. Thereby, we also eliminate another middle man, the ambient temperature.

The outlet temperature T_i^{out} of the ith node is modeled from its inlet temperature T_i^{in} and the average multicore temperature of the node. Thus, we express temperature T_i^{out} as

$$T_i^{out} = T_i^{in} + K_i \times \frac{\sum_{j=1}^{m} T_{ij}^{core}}{m}. \quad (3)$$



Fig. 2: The environmental sensors attached to every data node on the front (left-hand side) and back (right-hand side) of the cluster in a traditional rack.

where T_{ij} is the j core in node i, and $\frac{\sum_{j=1}^{m} T_{ij}}{m}$ is the average core temperature in node i. Here, n denotes the total number of active nodes in the cluster; m is the number of cores in each node. Let T_i^{core} be the average core temperature. Thus, we have

$$T_i^{core} = \frac{\sum_{j=1}^{m} T_{ij}^{core}}{m}. \quad (4)$$

Applying (4) to (3), we simply temperature T_i^{out} as

$$T_i^{out} = T_i^{in} + K_i \times T_i^{core}. \quad (5)$$

B. A Sample Usage

Now we show how to apply the multicore model through a sample usage, where the well-known *COP* model (a.k.a., the Coefficient Of Performance model [11]) is incorporated. In this usage case, we make use of our model coupled with the COP model to estimate the cooling cost of a data center. After our tModel sheds light on outlet temperatures, we plug the temperature into the following equation (see (6)) to obtain a value of the COP parameter. A large COP value implies that a data center's energy efficiency is high and vice versa.

$$COP(T) = 0.0068 * T^2 + 0.0008 * T + 0.458. \quad (6)$$

COP in (6) is defined as the ratio of heat removed to the energy cost of the cooling system for heat removal. Let P_{AC} be cooling cost in terms of power consumed by the cooling system. We calculate cooling power P_{AC} from (1) the CRAC's supply temperature T and (2) power consumption P_C of the computing infrastructure. Thus, we have

$$P_{AC} = \frac{P_C}{COP(T)} \quad (7)$$

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

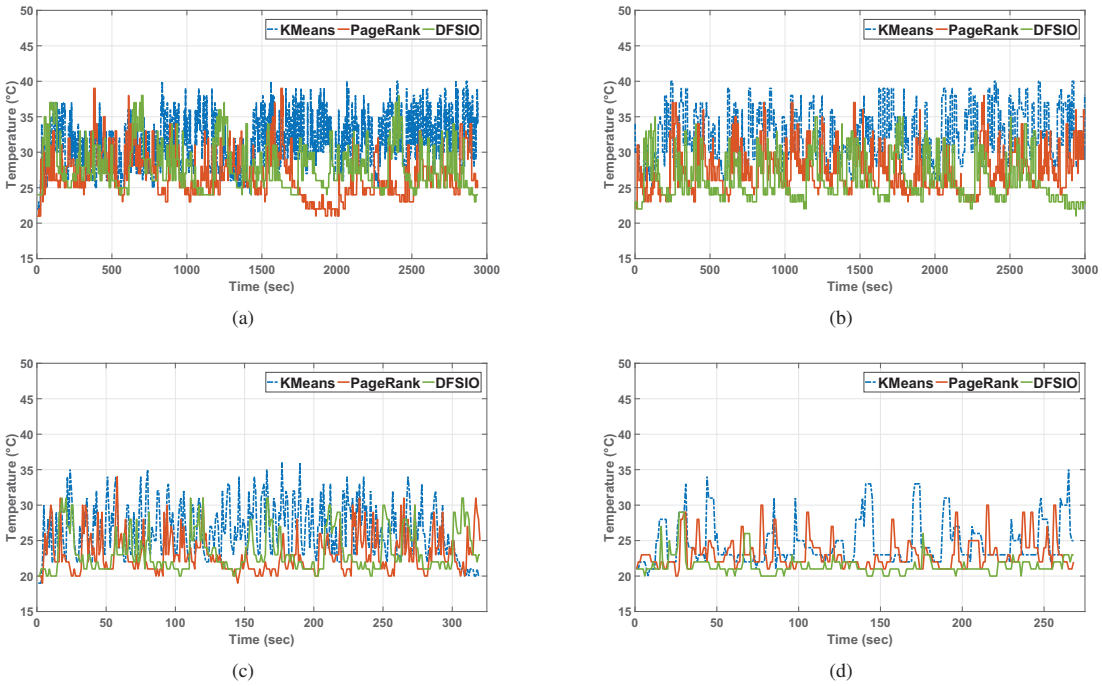

Fig. 3: One CPU core's temperature when the KMeans, PageRank, and DFSIO benchmarking applications are running on a cluster comprising of 4, 6, 8, and 12 nodes respectively.

where the cooling cost is inversely proportional to the COP value. It is worth noting that P_{AC} is a metric quantifying the data center's energy efficiency.

V. Experimental Evaluation

We implemented *tModel* on a testbed of one traditional rack consisting of 14 SuperMicro Model 825-7 servers with Intel (R) Xeon (R) X5650@2.67GHz processors. (see Fig. 2). The computer rack is located in Auburn University's High-Performance Computing Lab. The cooling unit supplies cool air from the ceiling and the room temperature is set to 63F. We allocate 16GB RAM for Hadoop leaving 8GB for operating system processors. Each map is allowed 4GB RAM and each reducer is allocated a minimum of 8GB RAM. This way, Hadoop can run upto 4 mappers and 2 reducers per node. To measure processor temperatures, we use a Linux utility program called *lm-sensors*; we deploy APC Environment sensors placed to the back and front of the cluster nodes and two APC Power Distribution Units (PDU) to measure an average power consumption of our cluster nodes. Further, we apply *Ganglia* to facilitate real-time monitoring for our Hadoop cluster. We measure execution time in terms of elapsed wall clock time. It is worth mentioning that throughout this study, we measure actual execution times, CPU temperatures, inlet and outlet temperatures the aforementioned MapReduce applications without relying on any simulation data.

In the following section, we evaluate the effectiveness of the proposed thermal model, *tModel*, in accurately estimating the outlet temperatures of the servers. We begin by profiling three bigdata applications on our testbed. The purpose of profiling is to understand the trends among inlet and multicore temperatures which can be used in calculating K used in our modeling equation. This K is employed to predict outlet temperature for determining the overall cooling costs for data centers.

A. Profiling

To facilitate the development of our *tModel*, we benchmark three bigd-ata applications namely, *DFSIO*, *KMeans*, and *PageRank*. KMeans is a CPU-intensive application; DFSIO is an intensive one. PageRank is comprised of CPU-intensive and I/O-intensive modules. The MapReduce implementation and various input data sizes for these applications can be found in *HiBench* benchmarking suite [6]. Developed by the Intel Labs, *HiBench* provides a collection of real-world Hadoop applications. HiBench enables a clear understanding of benchmark characteristics by running the actual Hadoop-implemented code of the applications on a Hadoop cluster as opposed to other trace-based workloads.

We conduct extensive experiments in three phases to estimate an accurate value of K proposed in our model. In phase I, we submit a large sized input for these benchmarks running on our cluster; we observe the thermal behavior of our cluster nodes. In phase II, we pick a representative benchmark to process different input data sizes available in the *HiBench* benchmark suite. In phase III, keeping the same representative benchmark and the input data size, we vary the number of running nodes (i.e., 4,6, 8, and 12) to further enhance of understanding of the variability in K of the applications on the cluster.

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

In each phase, we measure CPU core temperatures, inlet and outlet air temperatures of the master and data nodes in the Hadoop cluster every 10 seconds. The real-time thermal data allows us to (1) quantitatively assess our thermal model and (2) perform validations with a high degree of confidence. Figure 3 shows the CPU multicore temperatures of one representative cluster node running the KMeans, DFSIO, and PageRank applications. The four sub-figures in the Fig. 3 show thermal results for four different cluster sizes consisting of (a) 4 nodes, (b) 6 nodes, (c) 8 nodes and (d) 12 nodes. Thanks to the fair sharing of data by the Hadoop scheduler among all the cores, all the cores share the same temperature trends; and therefore, we choose to present only one out of 12 cores in these figures.

It must be noted that the execution time needed for processing a given data set with different number of nodes takes different time; we present the results keeping the execution of shortest job in each of the three cases. It is worth noting that the peak CPU temperature for KMeans is much higher than its counterparts - PageRank and DFSIO, owing to the KMeans nature of being a CPU-intensive one. The trend is similar given a fixed input data size and different number of nodes in a cluster. This type of profiling helps our multicore model in predicting an average multicore temperature, which is fed into the outlet-temperature module of *tModel* (see Fig. 1).

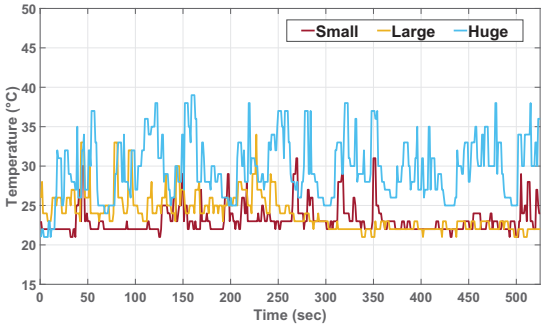

Fig. 4: The temperature of one of twelve cores of a representative data node running KMeans on an eight-node cluster with diverse input data size - small, large, and huge available with *HiBench*.

B. Thermal Impacts of Data Size

In this set of experiments, we evaluate the impacts of data size on the thermal behavior of KMeans. We set three input data sizes - small (number of samples: 30000), large (number of samples: 3000000), and huge (number of samples: 100000000) see also *HiBench* discussed in Section V. KMeans takes 4 minutes, 15 minutes, and almost two hours respectively to finish under above mentioned input sizes. Although the execution time of each job is different, we present thermal results for first four minutes for the sake of comparison. To validate our tModel, we use K_i value calculated from first job execution with large input size to predict the outlet temperature of nodes for subsequent experiments.

Fig. 4 shows the impacts of different data size of the same application on a node's processor temperature. Clearly, as the data size becomes large, the average core temperature goes up. For instance, Fig. 4 shows that the peak temperature with small data size is $31C$, whereas the peak temperatures for *large* and *huge* data sizes are $34C$ and $40C$, respectively. Fig. 5 confirms that processors' core temperature rises with an increase in input size of the applications - large (number of samples: 3000000), huge (number of samples: 100000000), and gigantic (number of samples: 200000000). We have seen similar trends in all the other applications.

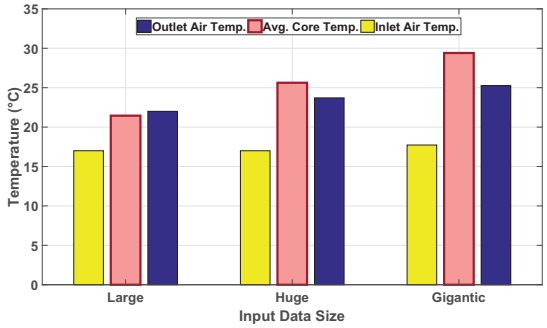

Fig. 5: The average temperature of all cores of one representative data node running *KMeans* on a 13-nodes cluster with diverse input data size - large, huge, and gigantic available with *HiBench*.

C. Improving Accuracy

Recall that we introduced K_i to reduce the cost of temperature sensors in order to predict the outlet temperature of nodes in the data centers (see Section IV). Through our preliminary profiling experiments (see Section V-A), we observe and analyze the value of K_i introduced earlier in Section IV using various applications and data sizes. This estimation is important in terms of costs involved in temperature sensors. We performed profiling in two steps. First, we measured processors' core temperatures/outlet temperature of the nodes in the clusters. Next, we used non-linear regression model to calculate corresponding values of K_i Then, we employ a non-linear regression model to estimate the value of K_i, which drives the estimation of outlet temperatures. For profiling K_i initially, we use outlet temperatures to train the model (i.e., obtain the K_i value). Next, we make use of the K_i value to predict the future outlet temperatures.

To evaluate the performance of tModel in terms of accuracy, we compared our model with existing beta model ([7]) and actual observed values. We use one application, KMeans, to calculate K values. We then used calculated K values to predict outlet temperature for other applications with same input data size. Fig.6 depicts our comparison results for KMeans application. It can be observed from Fig.6 that tModel outperforms beta model for all the nodes. It important to note that tModel successfully predicts the outlet temperature of the nodes more than 95% accuracy. This is a significant improvement, when compared with beta model, which is only 42% accurate.

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.

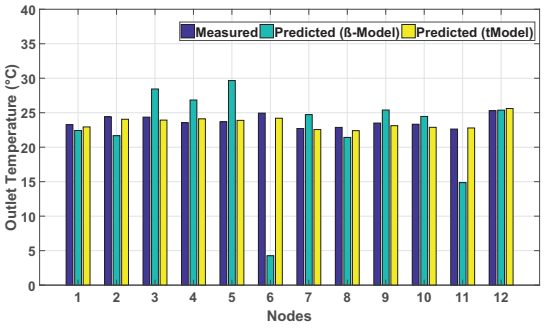

Fig. 6: The graph shows a node-wise comparison in estimating outlet temperature with *tModel* and β-model.

VI. Conclusion

We proposed a thermal model called *tModel* aiming to predict outlet temperatures of servers in Hadoop clusters running big-data applications. tModel eliminates a pressing need to estimate power consumed by Hadoop nodes prior to speculating their outlet temperatures. tModel makes use of a vital parameter K_i to resemble correlation between outlet and inlet temperatures of the ith node. Parameter k_i is obtained through profiling in the sampling phase. We apply the procured values for each individual core i to predict the thermal behaviors of big-data applications processing various data sizes. tModel is more accurate than traditional models that rely on power consumption and CPU utilization, because modeling core temperatures from CPU utilization is likely to introduce prediction errors.

Our thermal model offers two compelling benefits. First, *tModel* makes it possible to cut back thermal monitoring cost by immensely reducing the number of physical sensors needed for large-scale clusters. Monitoring temperatures is a vital issue for safely operating data centers; however, it is prohibitively expensive to acquire and set up a large number of sensors in a large data center. Second, *tModel* enables data center designers to evaluate thermal management strategies during the center design phase.

Acknowledgment

Xiao Qin's work is supported by the U.S. National Science Foundation under Grants IIS-1618669, CCF-0845257 (CAREER), CNS-0917137, CNS-0757778, CCF-0742187, CNS-0831502, CNS-0855251, and OCI-0753305. Xiao Qin's study is also supported by the 111 Project under grant No. B07038. Mohammed Alghamdi's research was supported by AL-Baha University.

References

[1] U.S. Environmental Protection Agency. Report to congress on server and data center energy efficiency. Technical report, August 2007.

[2] Shouq Alsubaihi and Jean-Luc Gaudiot. Pets: Performance, energy and thermal aware scheduler for job mapping with resource allocation in heterogeneous systems. In *Performance Computing and Communications Conference (IPCCC), 2016 IEEE 35th International*, pages 1–2. IEEE, 2016.

[3] Thang Cao, Wei Huang, Yuan He, and Masaaki Kondo. Cooling-aware job scheduling and node allocation for overprovisioned hpc systems. In *Parallel and Distributed Processing Symposium (IPDPS), 2017 IEEE International*, pages 728–737. IEEE, 2017.

[4] Gartner. Four megatrends impacting the data center, 2016.

[5] http://www.datacenterdynamics.com/research/energy-demand 2011-12. Global data center energy demand forecasting. Technical report, institution, 2011.

[6] Shengsheng Huang, Jie Huang, Jinquan Dai, Tao Xie, and Bo Huang. The hibench benchmark suite: Characterization of the mapreduce-based data analysis. In *Data Engineering Workshops (ICDEW), 2010 IEEE 26th International Conference on*, pages 41–51. IEEE, 2010.

[7] Rini T Kaushik and Klara Nahrstedt. T*: A data-centric cooling energy costs reduction approach for big data analytics cloud. In *High Performance Computing, Networking, Storage and Analysis (SC), 2012 International Conference for*, pages 1–11. IEEE, 2012.

[8] Jonathan G. Koomey. Estimating total power consumption by servers in the U.S. and the world. Technical report, Lawrence Derkley National Laboratory, February 2007.

[9] Lei Li, Chieh-Jan Mike Liang, Jie Liu, Suman Nath, Andreas Terzis, and Christos Faloutsos. Thermocast: a cyber-physical forecasting model for datacenters. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '11, pages 1370–1378, New York, NY, USA, 2011. ACM.

[10] Shen Li, Hieu Le, Nam Pham, Jin Heo, and Tarek Abdelzaher. Joint optimization of computing and cooling energy: Analytic model and a machine room case study. In *Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on*, pages 396–405. IEEE, 2012.

[11] Justin Moore, Jeff Chase, Parthasarathy Ranganathan, and Ratnesh Sharma. Making scheduling "cool": temperature-aware workload placement in data centers. In *Proceedings of the annual conference on USENIX Annual Technical Conference*, ATEC '05, pages 5–5, Berkeley, CA, USA, 2005. USENIX Association.

[12] Ehsan Pakbaznia and Massoud Pedram. Minimizing data center cooling and server power costs. In *Proceedings of the 2009 ACM/IEEE international symposium on Low power electronics and design*, pages 145–150. ACM, 2009.

[13] L. Parolini, B. Sinopoli, B. H. Krogh, and Z. Wang. A cyber physical systems approach to data center modeling and control for energy efficiency. *Proceedings of the IEEE*, 100(1):254–268, Jan 2012.

[14] Marco Polverini, Antonio Cianfrani, Shaolei Ren, and Athanasios V Vasilakos. Thermal-aware scheduling of batch jobs in geographically distributed data centers. *IEEE Transactions on cloud computing*, 2(1):71–84, 2014.

[15] Luiz Ramos and Ricardo Bianchini. C-oracle: Predictive thermal management for data centers. In *High Performance Computer Architecture, 2008. HPCA 2008. IEEE 14th International Symposium on*, pages 111–122. IEEE, 2008.

[16] Barry Rountree, Dong H Ahn, Bronis R De Supinski, David K Lowenthal, and Martin Schulz. Beyond dvfs: A first look at performance under a hardware-enforced power bound. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International*, pages 947–953. IEEE, 2012.

[17] Osman Sarood and Laxmikant V. Kale. A 'cool' load balancer for parallel applications. In *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '11, pages 21:1–21:11, New York, NY, USA, 2011. ACM.

[18] Qinghui Tang, Sandeep Kumar S. Gupta, and Georgios Varsamopoulos. Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: A cyber-physical approach. *IEEE Trans. Parallel Distrib. Syst.*, 19(11):1458–1472, November 2008.

[19] Qinghui Tang, Tridib Mukherjee, Sandeep KS Gupta, and Phil Cayton. Sensor-based fast thermal evaluation model for energy efficient high-performance datacenters. In *Intelligent Sensing and Information Processing, 2006. ICISIP 2006. Fourth International Conference on*, pages 203–208. IEEE, 2006.

[20] Georgios Varsamopoulos, Ayan Banerjee, and Sandeep KS Gupta. Energy efficiency of thermal-aware job scheduling algorithms under various cooling models. *Contemporary Computing*, pages 568–580, 2009.

[21] Nedeljko Vasic, Thomas Scherer, and Wolfgang Schott. Thermal-aware workload scheduling for energy efficient data centers. In *Proceedings of the 7th international conference on Autonomic computing*, ICAC '10, pages 169–174, New York, NY, USA, 2010. ACM.

Authorized licensed use limited to: Auburn University. Downloaded on November 17,2026 at 23:07:39 UTC from IEEE Xplore. Restrictions apply.