Energy-Consumption Clustering in Cloud Data Centre

Conference Paper · March 2016

DOI: 10.1109/ICBDSC.2016.7460373

3 authors, including:

Salam Ismaeel
Ryerson University

Ayman Al-Khazraji
University of Bahrain

Energy-Consumption Clustering in Cloud Data Centre

Salam Ismaeel and Ali Miri
Department Computer Science
Ryerson University
Toronto, Canada
Email: salam.ismaeel, ali.miri@ryerson.ca
Ayman Al-Khazraji
Department of Electrical and Computer Engineering
Caledonian College of Engineering
Muscat, Oman
Email: ayman.alkhazraji@yahoo.fr

***Abstract*—In order to perform optimal Virtual Machine (VM) consolidation under QoS constraints based on energy consumption in Cloud Data Centres (CDCs) containing heterogeneous physical resources, one must build a framework that combines many subsystem algorithms, including prediction, selection, placement, etc. Several energy minimization strategies can be used in CDCs, but the most importantly is one in which minimization is done through switching off selected host underload servers after reallocating all VMs on the selected server.**

Predicting the resources needed in a given time period is the first and the most important step in dynamic provisioning to meet QoS expectations in the presence of variable workloads. In other words, we use previous usage patterns to estimate future VM request workloads in a data centre. The first step in the prediction process of the energy-consumption framework is to cluster the historical data. In this paper, we proposed a clustering for both user and VM requests. Real Google traces that feature over 25 million tasks collected over a 29-day period are used as an example in this paper.

***Index Terms*—Cloud computing, data centre management, energy efficiency, workload prediction, clustering**

I. INTRODUCTION

Data centre consist of a large number of components from various research areas such as computing, networking, management, and the like, making energy reduction in data centre equipment one of the most demanding and difficult challenges because the complexity of system [1]. Energy consumption is not only determined by hardware efficiency, but also by the resource management system deployed on the infrastructure and the efficiency of applications running on the system.

A VM is a computing engine with predetermined virtual resources such as processors, memory storage etc., and a computing stack consisting of an OS and/or middleware and one or more application programs. Virtualization technology enables multiple virtual servers to run on the same Physical Machine (PM), which is helpful in improving resource utilization [2]. One of the most efficient methods to improve the utilization of the resources and reduce energy consumption is the dynamic consolidation of VMs [3, 4, 5].

In Infrastructure as a Service (IaaS), VM instances are configured with an operating system. Based on a Service Level Agreement (SLA) with cloud providers, the tenants order groups of VMs which are placed on different hosts and allow communications between each other [6].

Energy-efficiency based on VM migration can be done by VM consolidation. This consolidation can be performed either statically or dynamically [3]. Predefined resources, or static provisioning can lead to inefficiency, as it can lead to increased costs for application providers during low demand resource periods, and poor Quality of Service (QoS), leading to loss of costumers and revenue, during high utilization periods [7].

In contrast, dynamic provisioning attempts to determine the amount of resources needed in a given time period to meet QoS expectations in the presence of variable workloads. Approaches taken to dynamic provisioning fall under *reactive* or *proactive* categories. Proactive methods use monitoring, historical date (when available) and prediction algorithms to achieve their goals [8, 9].

Prediction plays a key role in these proactive methods, and in general for efficient resource utilization strategies for dynamic cloud computing environments. In fact, performance predictability is often listed as 1 of 10 obstacles and opportunities for growth in the cloud computing paradigm [9].

Resource estimation underlies various workload management strategies including dynamic provisioning, workload-scheduling, and admission control. All these approaches possess a prediction module in common which provides estimations to determine, respectively, whether or not to add more resources, rearrange the order of query execution, and admit or reject a new incoming query [2]. Forecasting load for a period of seconds or minutes will be necessary for real-time control, resource allocation, capacity planning and data centre energy saving in cloud computing.

In previous work [2], we proposed a framework that combines k-means clustering and Extreme Learning Machine

(ELM) to forecast VM requests in a data centre. This work will not discuss the prediction algorithm itself, rather the major contribution of this work is to combine clusterings for users as well as CPU and memory workload in a prediction subsystem. The paper also discusses a suitable framework for dynamic consolidations of VM based on energy consumption in CDCs.

The rest of the paper is organized as follows: Section II summarizes the proposed general block diagram of dynamic VM migration and consolidation based on energy consumption. VM clustering and user clustering will be discussed in Section III. Section IV concludes the paper.

II. Proposed Real-Time VM Consolidation based on Energy Minimization

As stated in the previous section, dynamic consolidation of VMs can improve the utilization of resources and reduce energy consumption by live migration and/or optimal placement of new or predicted VMs to switch off the selected underload hosts. The proposed system to make real-time VM consolidation based on energy consumption consists of the following related subsystems, as shown in Figure 1.

A. ***Workload Estimation***

Workload estimation is workload prediction, in particular here future VM request prediction, based on available historical data. In other words, previous usage patterns are used to estimate future VM requests in a data centre. It consists of two steps, VM clustering, which is the main topic of this paper, and prediction for each cluster [2]. This will help the data centre operator to place unneeded PMs in a low-power state to save energy [10].

Resource estimation underlies various workload-management strategies including dynamic provisioning, scheduling, and admission control. All these approaches possess a prediction module in common which provides estimations to determine, respectively, whether or not to add more resources, rearrange the order of query execution, and admit or reject a new incoming query [2]. Prediction of the future resource behaviour is a crucial issue for efficient resource utilization in dynamic cloud-computing environments because:

- Estimates of future performance or workload of each VM ensure the service quality and minimize costs [11].
- Such predications help administrators take appropriate action to prevent the system from suffering traffic surges or the *Slashdot effect* caused by high loads. [9].
- Effective prediction can facilitate proactive job scheduling or host-load balancing decisions. These improve resource utilization and job performance [12].
- Prediction is very important to reducing the cost of energy consumed in data centres by switching servers to lower power states when they are not in use [6]. In recent years, the energy costs of cloud data centres have become a practical concern, as discussed in [13, 14].
- Workload estimation is not only used to decide whether to add or remove resources, but also to rearrange the order of query execution, and admit or reject new incoming resource requests [15].
- Accurate predication host loads is also an important key in satisfying SLAs. However, such accurate predictions are extremely challenging, as fluctuations in the load can happen on small timescales[12].

In summary, forecasting load for a period of seconds or minutes will be necessary for real-time control, resource allocation, capacity planning and datacentre energy saving in cloud computing. That is why performance predictability is listed as 1 of 10 obstacles and opportunities for the growth of cloud computing [9].

B. ***Resource Demands***

This subsection discusses physical component considerations and monitoring. Monitoring is the continuous measurement, and assessment of performance, reliability and power usage of infrastructure and applications behaviours. It is performed while maintaining good quality of service. In other words, it is the dynamic profiling of the QoS parameters related to hardware and software resources [16, 17].

Monitoring power consumption is required not only for understanding how power is consumed, but also for assessing the impact of energy management policies [18]. It will help in detecting and tracing the variations or failure of resources and applications [17]. There are many tools used in cloud monitoring such as *Collectd*, *Nagios*, and *Ganglia*, which provide the capability to monitor computing, networking and storage resource utilization [19]. For our proposed system, we used the following to monitor a large-scale CDC on an OpenStack platform:

a) *OpenStack Ceilometer* can be used to reliably collect measurements of the utilization of physical and virtual resources comprising deployed clouds [20]. Ceilometer collects data from different levels of the computing infrastructure (e.g., VM container, hypervisors, storage, and network) and software resources (e.g., web server, application server, database server, and virtual applications) [17, 21].

b) *Data centre Infrastructure Manager (DCIM)* provide detailed information about server configurations, hardware, network connections, installed software, and so on [22]. DCIM profiles the power consumed by each part of the hardware in a data centre [23].

Physical component considerations include:

a) *Host underload* is when a host is underutilized, and so it should be switched off. In these situations, all VMs on this host should be migrated elsewhere. The Least utilized host and the static threshold are the basics techniques to decide if a host is underloaded.

b) *Host overload* is a host that some of the VMs should be migrated from to avoid violating QoS requirements. Static utilization thresholds, Adaptive utilization base, and Regression-based techniques are some commonly techniques.

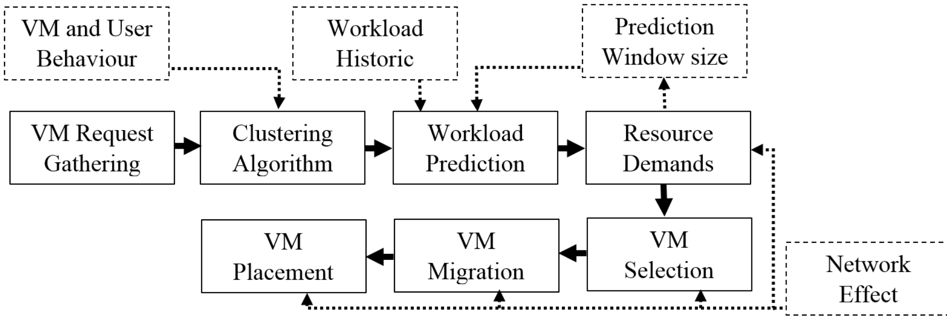

Figure 1: Dynamic VM Consolidation System

c) *PM selection algorithms* are used to determine the best placement of new VMs or to identify the VMs selected for migration to other servers. These choices will depend on many factors, such as workload dependencies, security, network load, …etc.

C. VM Placement (VMP)

VM placement is the process of mapping VMs to PMs in such a way that the hosts are utilized to their maximum efficiency. This will help to shut down unused servers depending on load conditions [3]. VM Placement algorithms can be divided into *Deterministic*, *Heuristic*, *Approximation* and *Meta-heuristic* algorithms [24].

a) *Deterministic algorithms* are based on optimization techniques like linear programming, binary integer programming, constraint programming, convex optimization theory and pseudo-Boolean optimization.
b) *Heuristic algorithms* are algorithms such as *First Fit (FF)*, *Best Fit (BF)*, *First Fit Decreasing (FFD)*, *Best Fit Decreasing (BFD)*, and *First-Come First-Served (FCFS)*. The minimum number of PMs suggested by these algorithms can provide suboptimal solutions [24, 25].
c) *Meta-heuristic algorithms* are a way of solving the *bin packing* problem with certain constraints. Most of them are based on the Genetic algorithm.
d) *Approximation algorithms* depends on prediction algorithms where prices of resources are not known but for example their probability distributions are known.

A suitable VMP algorithm should take into consideration the VM migration algorithm used, the VM selection approach and the effect all of that on the cloud network.

D. VM Migration

VM migration includes migrating VMs from underloaded servers to improve the utilization of resources and minimize energy consumption. This decision must include two main situations: determining the best time to migrate VMs to minimize energy consumption, and satisfying the defined QoS constraints [4].

E. VM Selection

VM selection is the process of selecting one or more VMs from a full set of VMs allocated to the server and the future predicted new VMs, to be located or reallocated to other servers [4]. The problem consists in determining the best subset of VMs to migrate in order to provide the most beneficial system reconfiguration in terms of energy consumption and many other parameters like security, bandwidth, etc.

F. Network Consideration

Communication between all the previously discussed blocks is also an energy consideration. It is necessary to keep the network infrastructure topology and routing protocol [26] and to try to migrate or consolidation with minimum network load.

G. Prediction Window

One of the most important parameters in workload predication is the length of the time period used. It is based on this time period that we decide whether or not PMs need to be switched off. Dabbagh *et al* [6, 27] estimate this period based on the difference between the energy cost for keeping the PM idle and the PM OFF/ON power cost. This is described in the following equation:

$$E_{sleep} = E_0 + P_{sleep} \cdot (T_p - T_0) \tag{1}$$

where T_p is the length of the prediction window, P_{sleep} is the consumed power when in sleep mode, E_0 is the energy needed to switch the PM to sleep mode plus the energy needed to wake up it later, and T_0 is the transitional switching time. The estimated time required to keep the PM ON and idle (T_b) consumes an amount of energy that is equal to the energy consumed due to mode transition plus that consumed while the PM is in the sleep mode during that same period:

$$P_{idle} \cdot T_b = E_0 + P_{sleep} \cdot (T_b - T_0) \tag{2}$$

In this equation, T_b is the beak-even time. This means that energy can be saved by switching PM to sleep mode if and only if the PM stay idle for a time period longer than T_b. That is, $T_p \geq T_b$ must hold in order for the power switching decision to be energy efficient.

According to the equation above, if we have PM profiles we can easily estimate the value of T_p. Dabbagh used the energy measurement study of PMs conducted in [28] to estimate the break-even time, T_b.

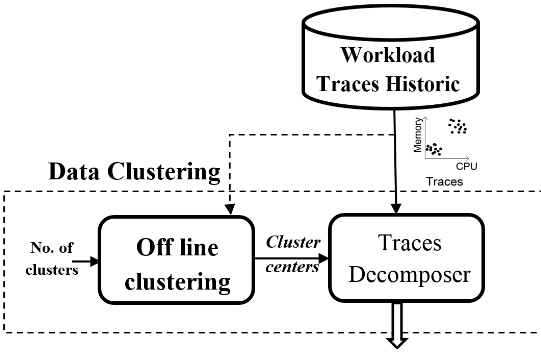

Figure 2: Data Clustering Subsystem

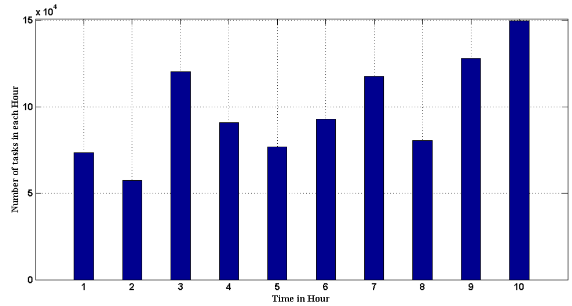

Figure 3: Number of tasks in each hour

On the other hand, Prevost *et al* [10], presented a dynamic prediction quantization method to determine the optimal number of prediction calculation intervals to be performed within required future load SLAs.

III. Data Clustering

As described in our previous work [2], data clustering consists of two parts. Figure 2 shows these two parts: (1) a cluster algorithm to create a set of clusters, and (2) use a trace decomposer which is responsible for mapping each request received during an observation window into one cluster for a predefined observation interval [2]. The workload in a CDC is driven not only by task characteristics, but also by users' behavioural patterns [29]. In next sections, we will discuss our proposed based on VM requests and user behaviour clustering using k-means and fuzzy c-means clustering algorithms. Google trace data [30] is used as a case study to compare workload characteristic analysis.

A. VM Clustering

VM clustering is used to create a set of clusters of different types of VMs or tasks during a specified time frame, or an observation window. In other words, VM clustering maps each VM request into one and only one cluster. Any clustering algorithms can be used. In this work, we compared k-means and fuzzy c-means clusterings.

Using k-means clustering, observations are partitioned into k clusters in which each VM request (CPU and Memory) belongs to the cluster with the nearest mean. After determine the centroids for each cluster. The main feature of k-means is its simplicity and ease of implementation. k-means clustering algorithm has three steps: the selection of the number of the clusters k, assigning all points in the data set to the closest cluster centre (centroid) to form k clusters, re-calculating centroids for each cluster to improve accuracy [2]. The second and the third steps are repeated until the clusters stabilize.

Fuzzy c-means is an extension of k-means algorithm where every VM request in the dataset belongs to every cluster to a certain degree. In other words, a certain request that lies close to the centre of a cluster will have a high membership to that cluster and another request that lies far away from the centre of a cluster will have low membership in that cluster.

The number of clusters should balance between the two conflicting objectives: reducing errors and maintaining low overhead. For example, Rasheduzzaman *et al* [31] chose to use cluster sizes of chose $k = 5$, and 6. However, they did not take into consideration the affect of increasing k on the performance of the predictor, as it has been discussed in [27] for which authors suggest using $k = 4$. In our previous work [2], we used $k = 3$, similar to the choice made by Moreno *et al* [29], as they included user behavioural patterns in their work.

To compare and discuss the effect of the number of clusters, we used 10 hours of available Google trace data. Figure 3 shows the number of task requests in each hour, where the total number of tasks was 1,029,342, each of which has a value of CPU and Memory.

The Sum of Squared Distances (SSD) is plotted as a function of number of clusters. SSD represents the error when each point in the data set is represented by its corresponding cluster center, as shown in following equation [2]:

$$SSD = \sum_{i=1}^{\mathbf{k}} \sum_{r=C_i} d(r, c_i)^2 \tag{3}$$

where C_i denotes the cluster i, i.e., the set of all points belonging to the i^{th} cluster. $d(r, c_i)$ is the Euclidean distance between r and c_i.

Figure 4 shows the comparison between k-means and fuzzy c-means for different numbers of clusters. We notice that, although the fuzzy c-means algorithm needs long off-line time, it produces better results than the k-means based method. This lead to choosing fewer number of clusters, which will affect the overall performance of proposed system for a cloud data centre based on energy consumption by balancing between reducing errors and maintaining low overhead.

Figure 5 shows the results for 4 clusters using fuzzy c-means, where each category is marked by a different color and the Centres of these clusters are marked by an **x**.

B. User Clustering

User and VM behaviours have strong influences on the overall cloud workload [29]. Comprehensive workload models must reflect realistic conditions in tasks, users and even VM

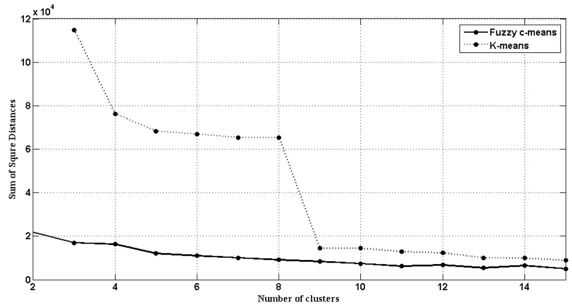

Figure 4: Number of VM clusters vs sum of square error

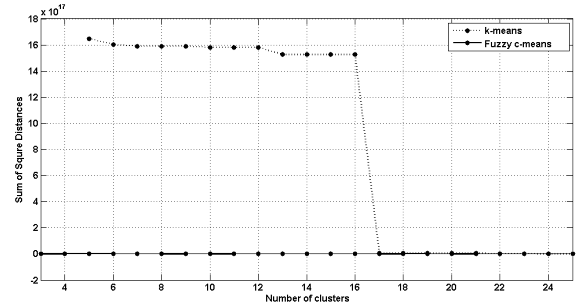

Figure 6: Number of user clusters vs sum of square error

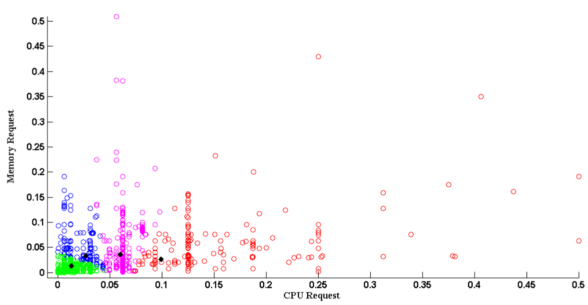

Figure 5: The resulting 4 (VM) task clusters for 10 hours using fuzzy c-means

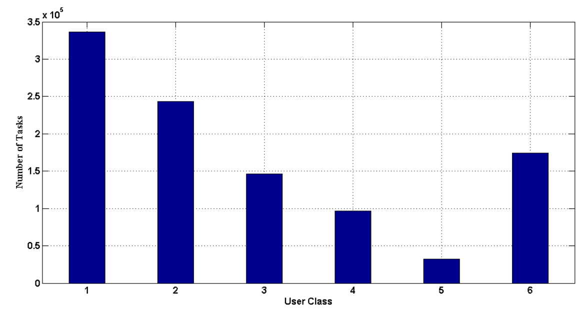

Figure 7: Number of task of 6 user classes

behaviour by excluding unwanted VMs or users form the workload estimation process.

Figure 6 shows k-means and fuzzy c-means clustering for different numbers of user clusters for the same period of Google data described in previous sections. The number of users over the 10 hours was 299. As expected, fuzzy c-means got better results for small number of clustering which will be very useful in reducing the number of input in a prediction system. This in turn will improve the overall proposed CDC energy consumption optimization. If we choose 6 classes of users, the number of tasks given to each class is shown in Figure 7.

IV. Conclusion

In order to perform optimal VM consolidation under QoS constraints based on energy consumption in a CDC containing heterogeneous physical resources, it is necessary to build a framework that combines all of the subsystems described in this paper. Minimizing energy should be done through switching selected underloaded host servers off after reallocating all VM on the selected server.

Monitoring should be used to collect data from different levels of the entire computing infrastructure (e.g., VM container, hypervisors, storage, and network) and software resources (e.g., web server, application server, database server, and virtual applications) using tools such as the OpenStack Ceilometer. The power consumed by each part of hardware in data centre can be monitored using tools such as the Data centre Infrastructure Manager (DCIM).

Our approach to VM forecasting is to combine user clustering and VMs clustering to obtain better predication of CDC energy consumption. Fuzzy c-means algorithm produces better results than the k-means based method for both VM and User clustering for small number of clustering which very important in reducing the number of input in a prediction system.

At the end, regardless of the clustering algorithm used, two goals should be considered: reducing errors and maintaining low overhead. In other words, although increasing the number of clusters in an algorithm reduces the error, this will also complicates the problem of prediction and consequently of CDC energy optimization.

References

[1] M. A. Koksal, I. H. Rowlands, and P. Parker, "Energy, cost, and emission end-use profiles of homes: An ontario (Canada) case study," *Applied Energy*, vol. 142, pp. 303–316, 2015.

[2] S. Ismaeel and A. Miri, "Using ELM techniques to predict data centre VM requests," in *Proceedings of The 2nd IEEE International Conference on Cyber Security and Cloud Computing (CSCloud 2015)*. New York, NY, USA: IEEE, Nov. 2015, pp. 80–86.

[3] A. Abdelsamea, E. E. Hemayed, H. Eldeeb, and H. Elazhary, "Virtual machine consolidation challenges: A review," *International Journal of Innovation and Applied Studies*, vol. 8, no. 4, pp. 1504–1516, 2014.

[4] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management

of data centers for cloud computing," *Future generation computer systems*, vol. 28, no. 5, pp. 755–768, 2012.

[5] E. Arianyan, H. Taheri, and S. Sharifian, "Novel energy and SLA efficient resource management heuristics for consolidation of virtual machines in cloud data centers," *Computers & Electrical Engineering*, 2015.

[6] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "Toward energy-efficient cloud computing: Prediction, consolidation, and overcommitment," *IEEE Network*, vol. 29, no. 2, pp. 56–61, 2015.

[7] K. B. Bey, F. Benhammadi, A. Mokhtari, and Z. Guessoum, "CPU load prediction model for distributed computing," in *Proceedings of The 8th IEEE International Symposium on Parallel and Distributed Computing. (ISPDC'09)*, 2009, pp. 39–45.

[8] R. Calheiros, E. Masoumi, R. Ranjan, and R. Buyya, "Workload prediction using ARIMA model and its impact on cloud applications' QoS," *IEEE Transactions on Cloud Computing*, vol. PP, no. 99, pp. 1–11, 2014.

[9] J. Cao, J. Fu, M. Li, and J. Chen, "CPU load prediction for cloud environment based on a dynamic ensemble model," *Software: Practice and Experience*, vol. 44, no. 7, pp. 793–804, 2014.

[10] J. Prevost, K. Nagothu, M. Jamshidi, and B. Kelley, "Optimal calculation overhead for energy efficient cloud workload prediction," in *World Automation Congress (WAC), 2014*, Aug 2014, pp. 741–747.

[11] D. Xu, S. Yang, and H. Luo, "A fusion model for CPU load prediction in cloud computing," *Journal of Networks*, vol. 8, no. 11, pp. 2506–2511, 2013.

[12] S. Di, D. Kondo, and W. Cirne, "Host load prediction in a google compute cloud with a bayesian model," in *in Proceeding of The IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, Nov 2012, pp. 1–11.

[13] N. R. D. C. (NRDC), "Data Center Efficiency Assessment," Natural Resources Defense Council (NRDC), Tech. Rep., August 2014.

[14] D. Sitaram, H. Phalachandra, S. HV, S. TP *et al.*, "Energy efficient data center management under availability constraints," in *Systems Conference (SysCon), 2015 9th Annual IEEE International*. IEEE, 2015, pp. 377–381.

[15] A. Khoshkbarforoushha, R. Ranjan, R. Gaire, P. P. Jayaraman, J. Hosking, and E. Abbasnejad, "Resource usage estimation of data stream processing workloads in datacenter clouds," *CoRR*, vol. abs/1501.07020, 2015.

[16] G. Aceto, A. Botta, W. De Donato, and A. Pescapè, "Cloud monitoring: A survey," *Computer Networks*, vol. 57, no. 9, pp. 2093–2115, 2013.

[17] A. Hameed, A. Khoshkbarforoushha, R. Ranjan, P. P. Jayaraman, J. Kolodziej, P. Balaji, S. Zeadally, Q. M. Malluhi, N. Tziritas, A. Vishnu *et al.*, "A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems," *Computing*, pp. 1–24, 2014.

[18] F. Rossigneux, L. Lefevre, J.-P. Gelas, D. Assuncao, and M. Dias, "A generic and extensible framework for monitoring energy consumption of openstack clouds," in *Proceedings of The 4th IEEE International Conference on Big Data and Cloud Computing (BdCloud)*. IEEE, Dec. 2014, pp. 696–702.

[19] B. Jennings and R. Stadler, "Resource management in clouds: Survey and research challenges," *Journal of Network and Systems Management*, vol. 23, no. 3, pp. 567–619, 2015.

[20] S. Ismaeel, A. Miri, D. Chourishi, and S. R. Dibaj, "Open source cloud management platforms: A review," in *Proceedings of The 2nd IEEE International Conference on Cyber Security and Cloud Computing (CSCloud 2015)*. New York, NY, USA: IEEE, Nov. 2015, pp. 470–475.

[21] C. A. Lee and A. F. Sill, "A design space for dynamic service level agreements in openstack," *Journal of Cloud Computing*, vol. 3, no. 1, pp. 1–13, 2014.

[22] D. Cole, "White paper: DCIM–what problem are you trying to solve?" Tech. Rep., 2014.

[23] M. Kamaludeen, "White paper: Data center energy reduction: A call for action in ontario," Tech. Rep., 2015.

[24] F. L. Pires and B. Barán, "Virtual machine placement literature review," *CoRR*, vol. abs/1506.01509, 2015.

[25] T. Mastelic, A. Oleksiak, H. Claussen, I. Brandic, J.-M. Pierson, and A. V. Vasilakos, "Cloud computing: Survey on energy efficiency," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, p. 33, 2014.

[26] K. L. LaCurts, "Application workload prediction and placement in cloud computing systems," Ph.D. dissertation, Massachusetts Institute of Technology, 2014.

[27] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, "Energy-efficient resource allocation and provisioning framework for cloud data centers," *IEEE Transactions on Network and Service Management*, vol. 12, no. 3, pp. 377–391, Sept 2015.

[28] I. Sarji, C. Ghali, A. Chehab, and A. Kayssi, "Cloudese: Energy efficiency model for cloud computing environments," in *Proceedings of The International Conference on Energy Aware Computing (ICEAC)*, Nov 2011, pp. 1–6.

[29] I. Moreno, P. Garraghan, P. Townend, and J. Xu, "Analysis, modeling and simulation of workload patterns in a large-scale utility cloud," *IEEE Transactions on Cloud Computing*, vol. 2, no. 2, pp. 208–221, April 2014.

[30] C. Reiss, J. Wilkes, and J. L. Hellerstein, "Google cluster-usage traces: format+ schema," *Google Inc., Mountain View, CA, USA, Technical Report*, 2011.

[31] M. Rasheduzzaman, M. Islam, T. Islam, T. Hossain, and R. Rahman, "Task shape classification and workload characterization of google cluster trace," in *Proceedings of the 2014 IEEE International Advance Computing Conference (IACC)*, Feb 2014, pp. 893–898.