Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 167 (2020) 1390–1399

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

Comparative analysis of VM consolidation algorithms for cloud computing

Nagma[a], Jaiteg Singh[b], Jagpreet Sidhu[c]*

[a,b]Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India
[c]Department of Computer Science and Information Technology, Jaypee University of Information Technology, Waknaghat, India

Abstract

Virtual machine consolidation is a major solution for addressing the issue of increasing energy consumption by cloud computing data centers. A lot of work is done on developing algorithms for detecting underloaded, overloaded hosts, selection of virtual machines and their placement to perform the consolidation. These algorithms are usually tested on publicly available Planet lab workload. There is a need to know how benchmarks algorithms used in consolidation of virtual machines respond to other workloads. This paper is an attempt to evaluate these algorithms on Google workload trace. An importer is made to use this dataset by extending the CloudSim toolkit. The comparison of results using Planet lab and Google workload traces is made which shows the difference of 46.41%, 14.84%, 12.86% and 44.83% respectively in terms of number of virtual machine migrations, service level agreement violation time per active host, number of hosts shutdown, and energy consumption. The objective comparison of results illustrated that there is a need to test the proposed algorithms on multiple datasets in order to be assessed as optimal.

Keywords: VM consolidation algorithms; comparative analysis; Google cluster trace; Planet lab workload; energy consumption

* Corresponding author. Tel.: +91-9888880237.
E-mail address: jagpreet.pu@gmail.com

10.1016/j.procs.2020.03.350

1. Introduction

Cloud computing has brought a revolution in the world by enabling utility-based computing following pay as you go model. Users can access resources dynamically from a shared pool by taking advantage of cloud deployment models [1]. Cloud computing has conquered the information technology industry by providing Infrastructure-as-a-service (IaaS), Platform-as-a-service (PaaS) and Software-as-a-service (SaaS) [2]. However, to give uninterrupted services to users, the cloud data centers demand enormous power for their operation. A report of International Energy Outlook states that from 2010 to 2040, the energy consumption in the world will rise by 56% and the major consumer will be IT industries [3]. So, cloud computing is not disruptive, but energy-efficient cloud computing has tremendous benefits. The current approach to address the problem of increasing energy consumption, and achieving energy efficiency is by using infrastructural, hardware and software-based solutions [3-5]. It is found that in spite of infrastructural and hardware changes, energy consumption did not reduce much. Therefore, virtualization based software solutions are the core of energy-efficient cloud computing [3]. Software-based solution includes consolidation techniques to switch off idle servers and shift Virtual Machines (VMs) of less utilized servers or highly utilized servers on moderately utilized servers for their optimum utilization. Fig. 1 shows the process of VM consolidation.

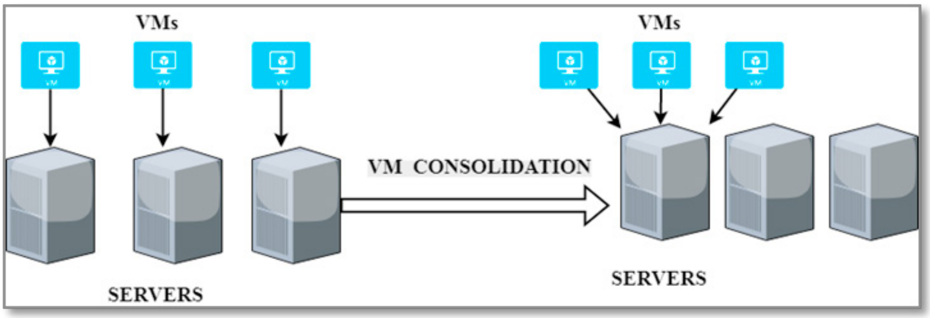

Fig. 1. VM consolidation process.

Consolidation involves finding underutilized, over-utilized hosts by using underload/overload detection algorithms and then choosing VMs through VM selection algorithms that are to be shifted to other hosts using VM placement algorithms. Fig. 2 shows the phases involved in consolidation of VMs.

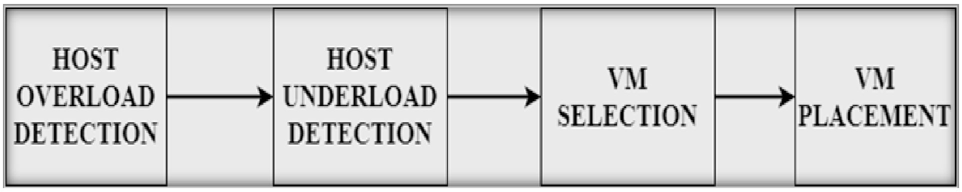

Fig. 2. VM consolidation phases.

It is a dynamic research area and tremendous amount of work is being carried out on the development of algorithms for VM consolidation. However, Planet lab traces are usually considered to evaluate the algorithms. It is not known how algorithms will respond to some other dataset. Many hypotheses regarding these consolidation algorithms for different phases appears to be ill-defined [6]. This work assesses standard algorithms for various phases of VM consolidation on Google cluster trace and results are compared with the execution of these algorithms on Planet lab trace. The major contribution of this study is as follows:

- This paper uses Google cluster trace to evaluate standard VM consolidation algorithms.
- Evaluation of algorithms is done on the basis of multiple parameters that are SLA violation time per active

host (SLATAH), number of hosts shutdown, number of migrations, and energy consumption.
- The comparison of results of execution using Planet lab and Google workload traces is made.
- Results show the difference of 46.41%, 14.84%, 12.86% and 44.83% respectively in terms of number of virtual machine migrations, SLATAH, number of hosts shutdown, and energy consumption among the best algorithms on these two datasets.

This paper consists of 5 sections and is organized as follows: Section 2 reviews the work done on consolidation algorithms and comparative analysis done by other researchers. Section 3 describes the environment for experimentation or comparison methodology. Section 4 presents the outcomes. Section 5 concludes the overall study and a future research strategy is presented.

2. Related Work

The work on energy-efficient cloud computing accomplished with the use of consolidation techniques can be divided on the basis of used datasets (real or synthetically generated), type of consolidation (QoS-aware or QoS un-aware), nature of threshold values (adaptive or fixed), components considered in power model (CPU, disk, memory, network).

In early stages, work related to energy-aware management of resources was carried on for mobile equipment to improve lifetime of battery [7]. Later on, the context has been shifted to data centers [8] and virtual computing environments such as clouds. Nathuji and Schwan [9] started the work for energy-efficient management of virtualized data centers. Both local and global approaches were used by authors to design an energy-aware resource management system. Authors designed a framework for energy-efficient management of resources by using both global and local policies. The limitation of research was that automatic resource management was not explicitly done for the global level.

Beloglazov and Buyya [10] proposed an energy-efficient resource management system for virtual cloud data centers that reduced operational cost and provided QoS. The policies for consolidation of VMs involved use of single static threshold value. No description of algorithms was given. Random data was used to validate the efficiency of the proposed system. The history of resource utilization values was not considered to calculate current resource utilization thresholds. Limitation of the work was that the designed power model considered CPU only as the source of power consumption and the proposed framework was not generic as it could not handle mixed workloads.

Beloglazov et al. [11] proposed an architectural framework and principles for energy-efficient cloud computing, resource provisioning and allocation policies. VM consolidation algorithms were based on double and fixed thresholds. Algorithms were not based on an analysis of historical data. Random data was used to validate the efficiency of the proposed model. The proposed system was not generic as it could not handle mixed workloads.

Beloglazov and Buyya [12] presented novel solution by dynamically consolidating the VMs considering historic values of CPU utilization to calculate resource utilization threshold. Two adaptive thresholds were calculated by authors. The designed policies are now the part of CloudSim 3.0.3. Evaluation of the system was done using Planet lab data. The proposed system could handle mixed workloads. It considered CPU as the only resource of power consumption. These policies act as a base for the design of other advanced algorithms and this initiated abundance of work in this field.

Beloglazov and Buyya [13] tried to improvise their preceding by introducing a policy that maximized the mean inter-migration time interval for the migrations initiated by overloaded hosts using Markov chain model. This work was an improved version of [12].

Alboaneen et al. [14] introduced a new policy called Maximum Requested Bandwidth (MBW) for selection of VMs for migration and a policy (Mn) for overloaded hosts detection. The algorithms of underload detection and VM placement were taken from [12]. CloudSim platform was used for conducting experiments using Planet lab data. The experiments demonstrate that the mean and maximum requested bandwidth (Mn_MBW) policy consumed less energy as compared to other algorithms.

Chowdhury et al. [15] proposed bin packing solutions for the problem of VM placement. CloudSim was used to evaluate the work on Planet lab data. This work was also based on heuristics provided by [12]. The algorithms for overload detection, underload detection, VM selection were the same as used by [12]. However, for VM placement, the authors proposed a new algorithm based on bin packing solutions.

Mann and Szabo [16] objectively compared the performance of VM placement algorithms in terms of different parameters and workloads. Datasets including traces of Google and Planet Google trace as well as Planet lab data but this work did not consider all the combinations of policies proposed in [12]. The work done in [17] is also motivated from [12].

Arockia and Sahayadhas [18] performed a comparative analysis of VM consolidation algorithms considering the parameters like total power consumption, VM migrations and SLA. However, evaluation was done using Planet lab trace only.

Ashraf et al. [19] performed a systematic mapping study of distributed VM consolidation algorithms and found that most of the evaluations are done using synthetic traces. However, this study uses real workload traces for evaluation.

Kumaraswamy and Nair [20] performed a comparative analysis of bin packing heuristics for VM placement. The study presented a systematic survey and comparison of existing bin packing methods for VM deployment. However, VM selection and host overload/underload detection algorithms were not considered.

Bermejo et al. [21] presented a summary of the past ten years of research on consolidated virtualized systems. The comparison was made to illustrate the energy-performance trade-off. So, this study has a different scope.

From the literature it is clear that the basic algorithms for VM consolidation are developed in [12]. These policies are considered to be standardized policies for research in the area of VM consolidation. Researchers have developed better algorithms using these algorithms as base. Thus, it becomes significant to perform behavioral analysis of standard policies on diverse datasets. Therefore, this paper is an effort to analyze the behavior of standard algorithms (Section III) using Google cluster data.

3. Experimental Setup

This work uses the experimental environment as described in this Section to perform comparative analysis of policies presented in [12] on Google cluster data.

3.1. Simulation Environment

VM consolidation algorithms are executed in CloudSim [18]. CloudSim is a toolkit developed by CLOUDS lab in the University of Melbourne for resource provisioning and virtualized environment modeling. It allows simulation of data centers having different hosts and VMs configurations. The power package is provided for performing simulations related to achieving energy efficiency in Cloud Computing. The classes in this package can be extended for developing advanced energy-aware algorithms. CloudSim is a preferred simulation environment for experimental evaluation as setting up of real virtualized environment is not possible [10-15]. The same default configuration of VMs and physical machines is used in this study as provided by [12].

3.2. Google cluster data

Google cluster data was made freely available to public in 2011. This dataset includes resource and workload data of a period of 29 days from a cluster of roughly 12,000 PMs [19]. The nature of this data is dynamic. For simulation purposes, host data, static and dynamic VM data are needed. Therefore, synthetic generation of host data and static VM data is done as motivated from [12] whereas dynamic data is taken from Google trace. Google cluster dataset is used into the experiment with the help of designed importer. So, CloudSim toolkit is altered for incorporating this dataset by extending the Power package.

3.3. Resource Management Policies

Resources can be consolidated onto a lesser number of machines by halting underutilized machines or by migrating the resources from over-utilized machines to manage energy consumption. VM consolidation involves (i)

Detection of overloaded hosts (ii) Detection of underloaded hosts (iii) VM Selection (iv) VM Placement. Fig. 3 shows the VM consolidation algorithms used by [12].

A comparative analysis is done for various combinations of host overload detection and VM selection algorithms whereas host underload detection policy used is Mu and VM placement is done by using PABFD in all runs.

So, this results in 5 (host overload detection algorithms) * 4 (VM selection algorithms) i.e. 20 combinations: "MadMu, MadMmt, MadMc, MadRs, IqrMu, IqrMmt, IqrMc, IqrRs, LrMu, LrMmt, LrMc, LrRs, LrrMu, LrrMmt, LrrMc, LrrRs, thrMu, ThrMmt, ThrMc, ThrRs". These 20 combinations of algorithms are completely evaluated and analyzed.

<table>
<tr><td>Median Absolute Deviation (MAD)</td><td>Interquartile Range (IQR)</td><td>Local Regression (LR)</td><td>Robust Local Regression (LRR)</td><td>Static CPU Utilization Threshold (THR)</td><td>Host Overload Detection Algorithms</td></tr>
<tr><td colspan="5">Hosts with Minimum Utilization</td><td>Host Underload Detection Algorithms</td></tr>
<tr><td>Minimum Migration Time (MMT)</td><td>Random Selection (RS)</td><td>Maximum Correlation (MC)</td><td colspan="2">Minimum Utilization (MU)</td><td>VM Selection Algorithms</td></tr>
<tr><td colspan="5">Heuristics like Power-aware Best Fit Decreasing (PABFD)</td><td>VM Placement Algorithms</td></tr>
</table>

Fig. 3. VM consolidation algorithms.

3.4. Comparison Parameters

The Google cluster data is used for comparing various policies on the basis of energy consumption and SLA related metrics. The parameters of focus are energy consumption, SLATAH, number of hosts shutdown, number of migrations.

- Energy consumption: Decreasing consumption of energy by cloud data centers needs sincere attention. It is computed as the summation of power consumption in a period of time as shown in Equation (1).

$$e(t) = \int_t p(t) \tag{1}$$

- SLATAH: This metric is related to SLA. It indicates the percentage of total SLA violation time for which a host has experienced 100% utilization. It is calculated as shown in Equation (2).

$$SLATAH = \frac{1}{n}\sum_{x=1}^{n}\frac{t_{s_x}}{t_{a_x}} \tag{2}$$

Where n stands for number of hosts; ts_x is the time for which the physical machine x is 100% utilized and ta_x is the time for which host x has experienced 100% utilization.

- Number of hosts shutdown: It represents the number of hosts turned off because of being underloaded during the process of VM consolidation.
- Number of migrations: This metric represents the number of migrations initiated by VM manager during VM deployment.

.

4. Results and Discussion

All arrangements of host overload detection and VM selection policies resulting in five categories or classes of policies – "Lr family (LrMc, LrMmt, LrMu, LrRs), Lrr family (LrrMc, LrrMmt, LrrMu, LrrRs), Mad family (MadMc, MadMmt, MadMu, MadRs), Iqr family (IqrMc, IqrMmt, IqrMu, IqrRs), Thr family (ThrMc, ThrMmt, ThrMu, ThrRs)" are run and their behavior for the comparison parameters including energy consumption, SLATAH, number of migrations and number of hosts shutdown is noted. The best algorithm for both the datasets Planet lab and Google workload trace in terms of particular parameter is found out.

4.1. Energy Consumption

The energy consumption of algorithms using Google workload and Planet lab traces is shown in Fig. 4 and 5 respectively. In terms of energy consumption using Google trace, LrMmt (LrrMmt also) is the best algorithm, whereas ThrMu showed worst performance in comparison to other algorithms. However, using Planet lab dataset, ThrMu and LrRs are worst and best algorithms respectively in terms of energy consumption.

Using Google trace workload, policies organized in increasing order of energy consumption are (i) in Iqr category are IqrRs, IqrMmt, IqrMc, and IqrMu (ii) in Lr category are LrMmt, LrRs, LrMu, and LrMc (iii) in Lrr category are LrrMmt, LrrRs, LrrMu, and LrrMc (iv) in Mad category are MadRs, MadMc, MadMmt, and MadMu (v) in Thr category are ThrMmt, ThrMc, ThrRs, ThrMu. Whereas, on Planet lab dataset, order in case of energy consumption is similar only for Lrr and Mad category. For other classes (Thr, Iqr, Lr), the increasing order is (i) ThrMc, ThrRs, ThrMmt, ThrMu (ii) IqrRs, IqrMc, IqrMmt, IqrMu (iii) LrRs, LrMc, LrMmt, LrMu respectively. Thus, it can be concluded that the behaviour of algorithms on different workloads is not the same. Different workloads exhibit different trends. So, in order to assess the performance of algorithms, they need to be tested on diverse workloads.

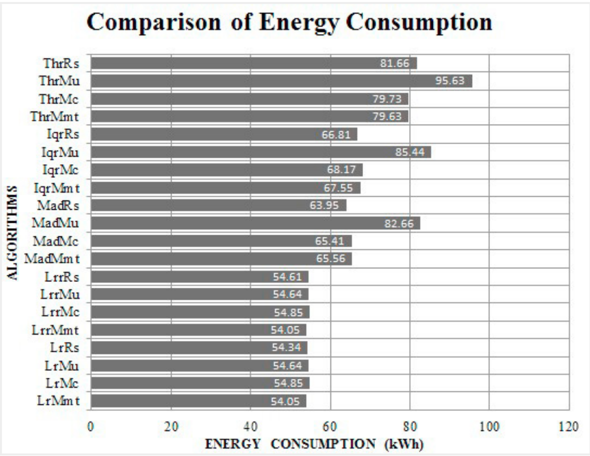

Fig. 4. Energy consumption of different algorithms on Google trace workload.

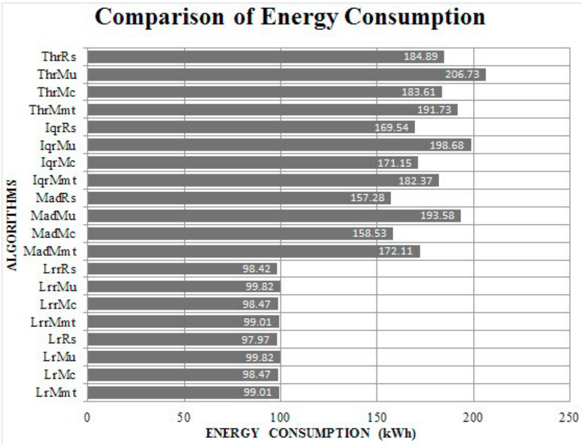

Fig. 5. Energy consumption of different algorithms on Planet lab trace.

4.2. Number of VM Migrations

The number of VM migrations of algorithms using Google workload and Planet lab traces is shown in Fig. 6 and 7 respectively. In terms of number of VM migrations using Google trace, the maximum migrations are found in IqrMu algorithms and minimum in LrMC or (LrrMc) algorithms in contrast to others. On the other hand, MadMu shows maximum migrations and LrMc shows minimum number of migrations in case of Planet lab workload.

Using Google trace workload, policies organized in increasing order of number of VM migrations (i) in Lr category are, LrMc, LrRs, LrMmt, and LrMu (ii) in Iqr category are, IqrRs, IqrMmt, IqrMc, and IqrMu (iii) in Lrr category, are LrrMc, LrrRs, LrrMmt, and LrrMu (iv) in Mad category are, MadRs, MadMc, MadMmt, and MadMu (v) in Thr class are,ThrMc, ThrRs, ThrMmt, and ThrMu. However, in case of Planet lab data, Thr, Lr, Lrr and Iqr policies show the same increasing order pattern in case of Google trace. For Iqr category the increasing order is IqrRs, IqrMc, IqrMmt, and IqrMu. Thus, it can be concluded that the behaviour of majority of algorithms on different workloads is of similar kind but it cannot be said for all classes of algorithms.to behave in a similar way or different datasets.

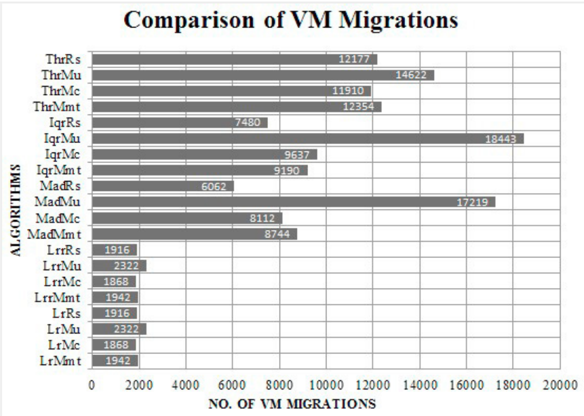

Fig. 6. Number of VM migrations of different algorithms for Google trace workload.

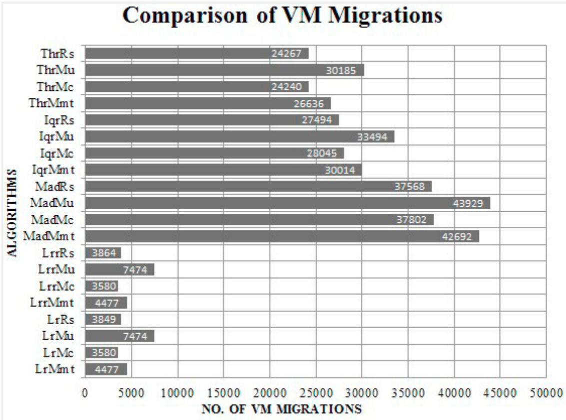

Fig. 7. Number of VM migrations of different algorithms for Planet lab workload.

4.3. Number of Hosts Shutdown

The number of hosts shutdown is given in Fig. 8 and 9. Using Google trace workload, policies organized in increasing order of number of hosts shutdown (i) in Lr class are LrMc, LrMmt, LrRs, and LrMu (ii) in Lrr class, are LrrMc, LrrMmt, LrrRs, and LrrMu (iii) in Mad class are MadRs, MadMc, MadMmt, and MadMu (iv) in Iqr class are IqrRs, IqMmt, IqrMc, IqrMu and (v) in Thr class are ThrMc, ThrRs, ThrMmt, and ThrMu. In case of Planet lab, the sequence is same for Mad class. The arrangement in other class of algorithms exhibit the following increasing order on Planet lab dataset (i) in Lr class is LrMc, LrRs, LrMmt, LrMu. (ii)in Lrr category, LrrMc, LrrRs, LrrMmt, and LrrMu (iii) in Iqr class is IqrRs, IqrMc, IqrMmt, IqrMu (v) in Thr class isThrMc, ThrRs, ThrMmt, and ThrMu.

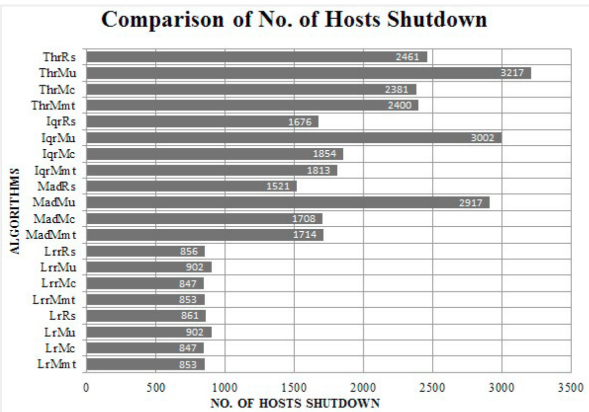

Fig. 8. Number of hosts shutdown in different algorithms during VM consolidation for Google trace workload

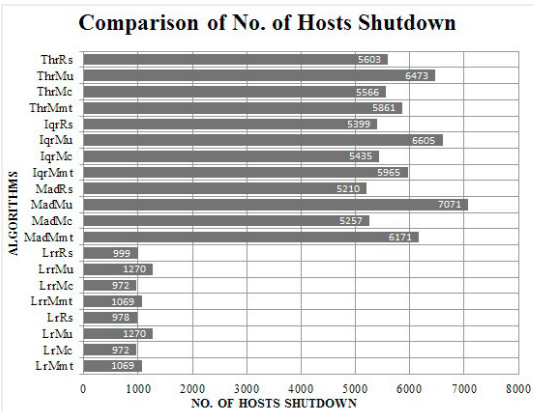

Fig. 9. Number of hosts shutdown in different algorithms during VM consolidation for Planet lab workload

Thus, it can be concluded that the behaviour of algorithms on different workloads is not the same. Different workloads exhibit different trends.

4.4. SLATAH

Fig. 10 and 11 illustrate the SLATAH on all algorithms on Google workload and Planet lab traces respectively. LrRs shows highest SLATAH as compared to other algorithms. However, ThrMmt shows minimum SLATAH. LrMmt (or LrrMmt) and ThrMmt respectively shows highest and lowest SLATAH on Planet lab dataset.

The algorithms arranged in increasing order of SLATAH on Google trace (i) in Lr class LrMc LrMu, LrMmt and LrRs, (ii) in Lrr class, are LrrMc, LrrMu, LrrMmt and LrrRs, (iii) in Mad class are MadMmt, MadMc, MadMu and MadRs, (iv) in Iqr class are IqrMmt, IqrMc, IqrRs and IqrMut and (v) in Thr class are ThrMmt, ThrMc, ThrRs and ThrMu. It can be found that in the case of Planet lab the increasing order of SLATAH remains the same for two categories of algorithms i.e. Iqr and Thr. For rest of the categories Lr, Lrr and Mad, the respective increasing orders of SLATAH on Planet lab workload are (i) LrMmt, LrMc, LrMu and LrRs (ii) LrrMc, LrrRs, LrrMu and LrrMmt (iii) MadMmt, MadMc, MadRs and MadMu.

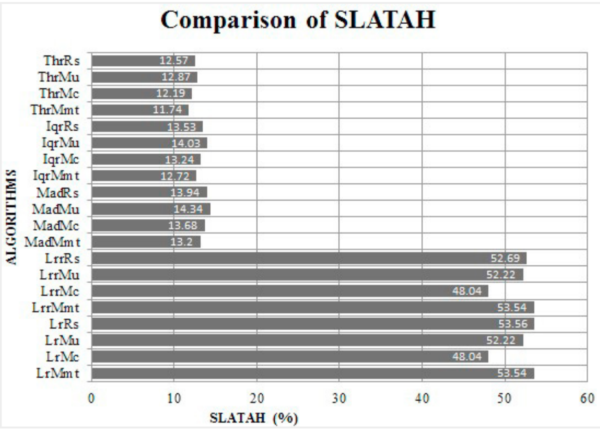

Fig. 10. SLA violation time per active host of different algorithms on Google trace.

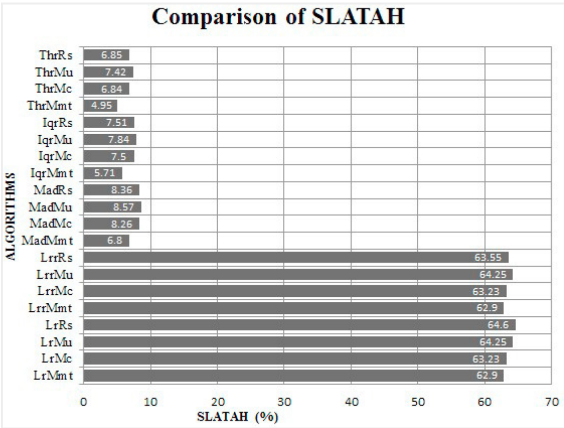

Fig. 11. SLA violation time per active host of different algorithms on Planet lab workload.

Fig. 12 shows the difference between the results on execution of algorithms using Planet lab workload and Google cluster trace.

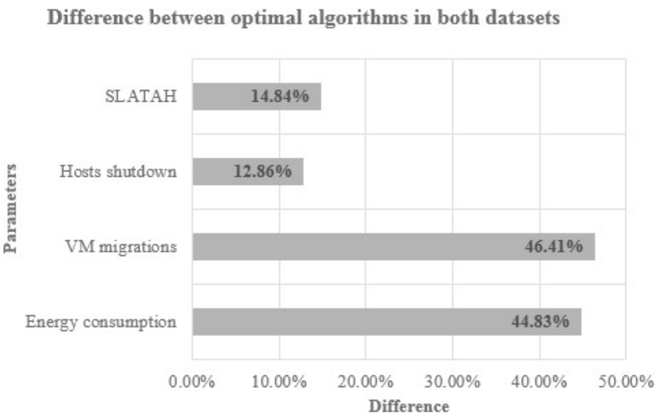

Fig. 12. Comparison between the results on Planet lab workload and Google cluster trace.

5. Conclusion

The research on VM consolidation in cloud computing environment is in very active stage currently. There is a plethora of work ongoing in this field based on benchmark algorithms. However, evaluation of the work is mostly done using Planet lab data. In this work, evaluation of the most commonly used algorithms is done using the most significant, publicly available Google workload trace. Comparison of different combination of policies is done using different datasets to evaluate the impact of workload on performance of algorithms. Results show that different policies behave differently on diverse workloads.

In terms of energy consumption and the number of migrations, LrMmt (LrrMmt also) and LrMc (LrrMc also) re the best algorithms respectively on Google trace. On the other hand, LrRs and LrMc are best algorithms on Planet lab data set.

Results indicate that energy consumption, number of virtual machine migrations, number of hosts shutdown and service level agreement violation time per active host of best algorithms differ by 44.83%, 46.41%, 12.86% and 14.84% on these two datasets.

Therefore, an algorithm must be tested on multiple workloads in order to call it optimal. One workload cannot decide the optimality of algorithm. The performance of algorithms should be tested on various parameters in order to evaluate its performance. Future work includes The future strategy is analysis of behaviour of algorithm using more datasets, to design an algorithm which may perform similar for different traces and parameters.

References

[1] Bhandayker, Yeshwanth Rao (2019). "A Study on the Research Challenges and Trends of Cloud Computing." *Research Review International Journal of Multidisciplinary* 4(2): 441-447.

[2] Liu, Ji, et al (2015). "A survey of data-intensive scientific workflow management." *Journal of Grid Computing* 13(4): 457-493.

[3] Kaur, Tarandeep, and Inderveer Chana (2015). "Energy efficiency techniques in cloud computing: A survey and taxonomy." *ACM computing surveys (CSUR)* 48(2): 22.

[4] Angel, Eric, Evripidis Bampis, and Fadi Kacem (2012). "Energy aware scheduling for unrelated parallel machines." *2012 IEEE International Conference on Green Computing and Communication*: 533-540.

[5] Orgerie, Anne-Cecile, Marcos Dias de Assuncao, and Laurent Lefevre (2014). "A survey on techniques for improving the energy efficiency of large-scale distributed systems." *ACM Computing Surveys (CSUR)* 46(4): 47.

[6] Mann, Zoltán Ádám (2015). "Rigorous results on the effectiveness of some heuristics for the consolidation of virtual machines in a cloud data center." *Future Generation Computer Systems* 51: 1-6.

[7] Zeng, Heng, Carla Ellis, Alvin Lebeck and Amin Vahdat (2002). “ECOSystem.” *ACM SIGOPS Operating System Review* 36(5): 123-132.

[8] Chase, Jeffrey S., et al (2001). "Managing energy and server resources in hosting centers." *ACM SIGOPS operating systems review* 35(5): 103-116

[9] Nathuji, Ripal, and Karsten Schwan (2007). "Virtual power: coordinated power management in virtualized enterprise systems." *ACM SIGOPS Operating Systems Review* 41(6): 265-278.

[10] Beloglazov, Anton, and Rajkumar Buyya (2010). "Energy efficient resource management in virtualized cloud data centers." *Proceedings of the 2010 10_{th} IEEE/ACM international conference on cluster, cloud and grid computing*: 826-831.

[11] Beloglazov, Anton, Jemal Abawajy, and Rajkumar Buyya (2012). "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing." *Future generation computer systems* 28(5): 755-768.

[12] Beloglazov, Anton, and Rajkumar Buyya (2012). "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers." *Concurrency and Computation: Practice and Experience* 24(13): 1397-1420.

[13] Beloglazov, Anton, and Rajkumar Buyya (2012). "Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints." *IEEE transactions on parallel and distributed systems* 24(7): 1366-1379.

[14] Alboaneen, Dabiah Ahmed, Bernardi Pranggono, and Huaglory Tianfield (2014). "Energy-aware virtual machine consolidation for cloud data centers." *Proceedings of the 2014 IEEE/ACM 7_{th} International Conference on Utility and Cloud Computing*: 1010-1015

[15] Chowdhury, Mohammed Rashid, Mohammad Raihan Mahmud, and Rashedur M. Rahman (2015). "Implementation and performance analysis of various VM placement strategies in CloudSim." *Journal of Cloud Computing* 4(1): 20.

[16] Mann, Zoltán Adám, and Máté Szabó (2017). "Which is the best algorithm for virtual machine placement optimization?" *Concurrency and Computation: Practice and Experience* 29(10).

[17] Monil, Mohammad Alaul Haque, and Rashedur M. Rahman (2015). "Implementation of modified overload detection technique with VM selection strategies based on heuristics and migration control." *IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS*): 223-227.

[18] Ranjini, Arockia and Sahayadhas Arun (2017). “A comparison study of various virtual machine consolidation algorithms in cloud datacenter.” *ARPN Journal of Engineering and Applied Sciences* 12(1*)*: 125-129.

[19] Ashraf, Adnan, Benjamin Byholm and Ivan Porres (2018). “Distributed virtual machine consolidation: A systematic mapping study.” *Computer Science Review 28*: 118-130.

[20] Kumaraswamy, S and Mydhili Nair (2019). "Bin packing algorithms for virtual machine placement in cloud computing: a review." *International Journal of Electrical and Computer Engineering 9*(1): 512-524.
[21] Bermejo, Belen, Carlos Juiz and Carlos Guerrero (2019). "Virtualization and consolidation: a systematic review of the past 10 years of research on energy and performance." *The Journal of Supercomputing* 75(2): 808-836.
[22] Calheiros, Rodrigo N., et al (2011). "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms." *Software: Practice and experience* 41(1): 23-50.
[23] Reiss, Charles, et al (2012). "Heterogeneity and dynamicity of clouds at scale: Google trace analysis." *Proceedings of the 3rd ACM Symposium on Cloud Computing:* 7.