JCSE International Journal of Computer Sciences and Engineering Open Access

Review Paper Vol.-7, Special Issue-14, May 2019 E-ISSN: 2347-2693

Dynamic Resource Adaptation in Cloud Computing

Jyoti Chalikar[1*], Gopal K Shyam[2]

[1,2]School of C&IT, Reva University, Bengaluru, INDIA

Corresponding Author:jyotivlakkannavar@gmail.com

DOI: https://doi.org/10.26438/ijcse/v7si14.401408 | Available online at: www.ijcseonline.org

Abstract- The need for the cloud resources is increasing and with increase in demand the cost of these resources is also increasing. Cloud environment gives the flexibility of utilizing the resources as per the need and the customer would pay for his usage. The consumer need not invest on the resources and thereby the cost of investment is drastically reduced for the consumer. But since the demand for the cloud resource is increasing, the cost is rising high. This can be reduced with an approach as proposed in this paper. This paper mainly focusses on the optimal way of resource adaption and hence reduction of cost and power consumption. Based on the analysis, there are some open challenges for the optimal resource adaptation. The resource's idle time is utilized by other consumer in need and hence reduces the cost and power consumption. This can be achieved by adopting k-means algorithm initially to segregate the different kinds of resources, then the idle time is calculated with time and at what time using some of the prediction algorithms. The idle time of the resources is then distributed using algorithms such as round robin, FCFS etc.

Keywords - Cloud, Resource, adaptation, machine learning

I. INTRODUCTION

Cloud computing is an emerging technology for the provision of computational services for a wide range of users which includes, software developers and research candidates. Infrastructure Providers (IPs) are the ones who manage the base infrastructure which includes servers, storage and network connectivity and these represent the virtual machines (VMs) other providers can either rent or lease these resources and resell value added services (VARs) as Platform as a Service (PaaS) or Software as a service (SaaS).

VARs make use of cloud to lower the costs incurred by the operations by paying only for the resources which would be used. They need not know the resource usage ahead for the capacity utilization. As a fact the VARs end up paying more for per hour compared to that of the infrastructure in house. IPs have a huge challenge in providing the benefits to VAR's. IPs are modeled to build the capacity to match to the varying demands for computing resources where huge investment is required for infrastructure, trained labor and cost of power usage.

Further, with the high competition & commoditization of cloud services, IPs are forced to reduce the cost. Amazon reduced its cost on 41 different occasions in the last few years [1]. Cloud computing adoption is open in a new market for IPs, where previously hosted in inhouse infrastructure.

The revenue for the IP would be generated if they are meeting the SLAs. To achieve this, we need to adapt the infrastructure configuration with respect to time. Adaptation basically refers to either theincrease or decrease of cloud resource for a workload.

For example, the CPU share which is allocated to a VM running on a web server can be reconfigured to a lower share adhering to the SLAs. The capacity gain can be used in accepting the varying workloads or in reduction of the consumption of power.

Several surveys put together results of different features of cloud resource management. In [2] the authors survey an elastic approach in cloud computing, providing a high-level overview of the approach. Our survey is a little different as it investigates adaptation and, as we demonstrate later, adaptation is a superset of elasticity. In [3], the authors discuss approach to efficient data centers, choosing to focus on power consumption. Our work covers power as an adaptation objective and covers SLA and revenue. In [4], the authors survey autoscaling, and classify the literature based on the adaptation techniques used. Their work focused on the Infrastructure as a Service (IaaS)client's perspective, while we focus on the IaaS provider, thus their work excluded VM migration and server consolidation. In [5], the authors provide an overview of the mechanisms and techniques adopted to manage elasticity from the perspective of a SaaS provider, while we focus on the IaaS provider. In [6], the authors investigate cloud resource management and in [7], the authors present common aspects used in cloud computing environments, such as metrics, tools and strategies. In [8], the authors survey the VM allocation problem and models and algorithmic approach. In [9], the authors present analysis of autonomic resource management in general, and specifically Quality of Service aware autonomic resource management. In [10], the authors survey SLAbased cloud research including the techniques used for adaptive resource allocation. In [11], the authors survey cloud computing elasticity using a classic systematic review covering metrics and tools. In [12], the authors summarize different methods and theory used in cloud resource allocation and monitoring. In [13], the authors depict a broad literature analysis of resource management in the cloud. While there is some overlap from these surveys with our work, they choose a different classification scheme for our work, which focuses on adaptation of resource configuration, enabling us to analyze the factors that influence the adaptation process. In addition, we investigate factors affecting scalability of the various proposals in the literature. To the best of our knowledge there is no other work that uses our chosen dimensions.

II. CLOUD SYSTEMS SETUP

Cloud System is defined as "Is the delivery of the IT services such as network, storage, services & applications over the internet". In this section we will introduce the constituents of cloud systems [14].

2.1 Compute Resource: A compute resource can be a host, or a pool in a virtualization platform on which the provision of machines is done. CPUs are basically used for executing the software instructions. The CPU available would be mainly the multicore processor, CPU cache & primary storage memory.

2.2 Storage Resource: The resources do not retain information and are called as non-volatile. This resource is cheaper than the primary resource & hence many operating systems will be able to use.

2.3 Network Resource: Network resources are the resources which are used to connect servers and infrastructure such as repeaters, load balancers, switches and firewalls. Networks make use of different protocols and topologies to have the Security, resilience & Quality of Service

2.4 Virtual Resource: The various kinds of resources are sliced into different units to distribute the workload evenly. Usually this happens in the data center where huge set of resources are stored at one place and they are served to the customers on need basis across world [2].

III. CLOUD SYSTEM ADAPTATION

In this section, the introduction to IP's objectives and approaches to adapting the cloud infrastructureis done. IP's can use elasticity [15] to meet the workload demands to reconfigure resources in an automatic manner. This might not be true in all the cases since IP has limited number of resources and it might apply some logic on the requests. Additionally, the IP may think of paying a penalty for violating the SLA rather than scheduling the requests since this iscost effective. Elasticity has certain activities to be performed which are complex. This can be refined by separating the decision-making process from how the cloud environment is reconfigured by defining the elasticity as the ability for on demand, to scale horizontally or vertically segmented resources in discreate units. IPs undergo decision making process which can change the infrastructure, a process which is termed Cloud Systems Adaptation. Cloud Systems Adaptation is defined as a change to provider revenue, data center power consumption, capacity or end-user experience which results in a reconfiguration of various resources such as computes, storage or network. This is shown in Figure 1 along with dependencies. The main motto of any cloud resource is its elasticity nature which would help in scaling.

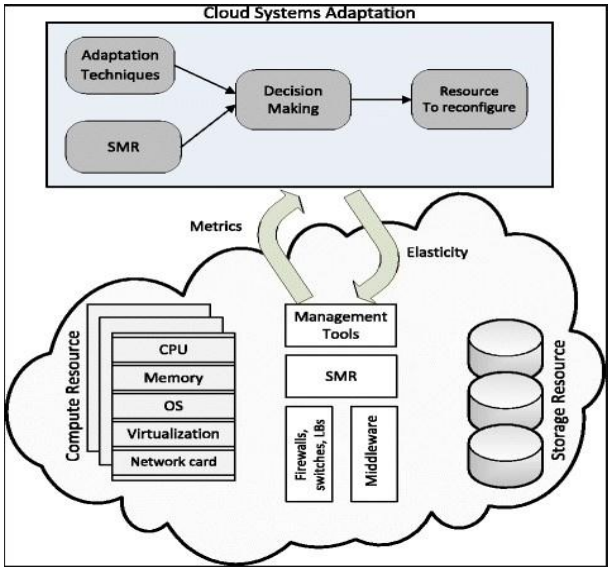

Fig.1 Cloud Systems Adaptation Sequence

3.1 Adapted Resource: VM level adaptations are applied to optimize the utilization of resource thereby reducing the workload on each resource. It might require that the CPU should have adjusting capabilities to the workload. Node level adaptation could be applied to add capacity by supplying power to a node. Power consumption could be reduced by Dynamic voltage & Frequency Scaling (DVFS) [17], before the node is powered off when it is not needed. Node configuration can be adapted when a VM's requirements extend beyond the capacity of its hosting node and can be migrated to another node which has the required capacity. Migration can reduce power consumption where the VM's are consolidated into fewer nodes & enabling some nodes to be switched off.

3.2 Adoption Objective: The focus here is to best utilize the resources on timely bases and allocate the same resource or share them with different customers thereby utilizing the resource optimally. The intension here is to not use any additional resource thereby reducing the cost for the resource and hence the power. This reduces the cost incurred by the customer for using the resources.

3.3 Adoption technique: This paper focuses on three different techniques. Heuristic, control theory and Machine Learning [18]. Heuristic is used to solve a problem with well-defined knowledge. Control theory works on feedback received and hence provide QoS, which adjusts the behaviors of the systems which are based on some outputs. Machine Learning techniques are broadly classified into two categoriesSupervised Learning where the algorithm analyzes the timing data and gives the output. Whereas unsupervised learning algorithms learn based on input fed.

3.4 Adaptation engagement: Cloud systems adaptation needs to be invoked to evaluate the infrastructure and determine whether resource configuration is required. The approaches in the literature are classified into Reactive, Proactive and Hybrid engagement.

Reactive approach involves adaptation when a monitored resource i.e., CPU utilization reaches a specific threshold. Proactive approach predicts the demands for the infrastructure and invokes adaptation ahead of the predicted resource attaining contention point.

Hybrid approach is a combination of both proactive & reactive approaches which engage adaptation for long- and short-term-time scales.

3.5 Decision Engine Architecture: Decision Engine Architecture makes use of the mechanism which is fed to the engine and decides whether the resource must be adapted or not. It keeps an eye on the engine and checks on its operation. Centralized architecture uses a mechanism with a global view of the managed infrastructure and can adapt the resources.Hierarchical architecture partitions the infrastructure into multiple partitions in which an engine is placed in each partition. At the global level is placed another engine which controls all these engines. Distributed architecture uses peer-to-peer protocol which is used to enable the node to communicate to the resources directly [19].

3.6 Managed Infrastructure: Cloud systems have a wide variety of compute and storage resources. Some proposals would make use of managed infrastructure in the decision-making process whereas others assume homogeneous infrastructure where every node has same capacity and power consumption [20].

IV. ADAPTATION OF CLOUD RESOURCE CONFIGURATION

Here the attempt is to adapt cloud resource configuration & focus on computing and storage resources. Analysis is done based on cloud resources which would be reconfigured to use. Reconfigured resources are

1. CPU & Memory
2. VM Migration
3. Node Power Usage
4. Storage

The purpose is to adhere to 100% SLA and if this is not met reiterations are done to meet the SLA 100%. Ideally SLAs should be met at 100%. But, they can vary from 95% to 99.99%. Companies like Microsoft Azure adheres to 99.97% SLA for most of their services.

In general, cloud systems adaptations are applied to minimize violations to SLAs and provide a secondary objective by recognizing that the business objective is not only to meet SLAs for IPs. To achieve this, different techniques and adaptations are used during proposals at different stages. In addition to this, the complexity of execution of the proposals impacts their ability to scale their approach in data centers with thousands of nodes. Therefore, the secondary objectives such as adaptation techniques, adaptation engagement & decision engine architecture are used to bifurcate many proposals in their cloud resource adaptation configuration.

4.1 CPU and Memory: CPU and memory adaptation have been researched as the main computing resources. Many service providers would help scaling the infrastructure horizontally by adding newer VM's [21-31]. This might look simpler but leads to wastage of work overload & in turn increase the power consumption. **Table 1**Summary of literature that adapt cloud resources, ordered by the Decision Engine Architecture.

4.2 VM adaptation: This paper mainly focuses on VM adaptation for various resources such as memory and CPU. The basic idea behind this VM adaptation is to use one of the ML algorithms and predict that which resource would be free and which resource is busy on a giventime and automatically shifts the VM to the other customer in need. Figure 2 shows thecomponents that may get adapted on compute resources. [10]

Project		Objective									Resource		**Adapt trigger**	**Arch**	Infra Workload	Setup [#nodes]
	P	SLA	Rev	Cust cost	Whole VM/ Node	*CPU*	*Mem*	*Migrate*	Disk	I/O	DVFS	Node off	ST			
Zheng [28]		x			x			x					GA	**P**	**Central**	Hom Generic	Simulation[200]
Zhang [29]		x			x								QT	**P**	**Central**	Hom Multi tier	Simulation
Zuo [51]		x			x			x			x		Heuristic	**R**	**Central**	Het Generic	Simulation
Tchana [48]			x		x			x					CSP	**R**	**Central**	Het Generic	Private + AWS
Beloglazov [33, 49]		x			x			x			x		Heuristic	**R**	**Central**	Het Generic	Simulation [100] [800]
Wesam [31]		x				x	x						Heuristic	**R**	**Central**	Het Multi tier	Xen test bed
Gmach [45]		x						x			x		CT	**R**	**Central**	Hom Generic	Simulation
Fargo [32]	x	x				x	x			x	x		Heuristic	**P**	**Central**	Hom Web	Xen test bed
Won Choi [50]		x						x					Heuristic	**R**	**Central**	Hom Generic	Linux test bed
Iqbal [46]		x						x			x		Heuristic	**R + P**	**Central**	Hom Generic	Eucalyptus
Roy [27]		x	x		x								CT	**P**	**Central**	Hom Multi tier	NA
Xiangping Bu [38]		x				x	x						RL	**R**	**Central**	Hom Multi tier	Xen test bed
Padala [30]		x				x			x				CT	**P**	**Layered**	**Hom Multi tier**	Xen test bed
Xu [41]		x				x							CT	**P**	**Central**	Hom Web	ESX test bed
Jamshidi [42]		x		x	x								CT	**R + P**	**Central**	Hom Web	Azure
Bodik [22]		x		x	x								CT	**P**	**Central**	Hom Multi tier	Simulation
Lama [23]				x	x								SML + Heuristic	**P**	**Central**	Het Hadoop	ESX test bed
Koehler [37]				x	x							x	Utility	**P**	**Central**	Hom Hadoop	KVM test bed
Kusic [35]	x	x			x	x					x		CT+ Utility+ TS	**P**	**Central**	Het Multi tier	ESX test bed
Zhu [40]		x				x					x		CT + Utility	**R**	**Central**	Hom Web	HP-UX
Hasan [44]		x				x							Heuristic	**R**	**Central**	Hom Generic	Test bed
Cardosa [36]	x				x								**Utility + Heuristic**	**R**	**Central**	Hom Generic	ESX test bed
Shen [34]	x	x				x	x	x		x			**TS**	**P**	**Central**	Het Web	Xen test bed
Nathuji [38]		x				x							**CT**	**P**	**Central**	Het Generic	Hyper-V test bed
Malkowski [24]		x			x								**CT + Heuristic**	**P**	**Central**	Hom Multi tier	Xen test bed
Lim [52]		x										x	**CT**	**R**	**Central**	Hom Hadoop	Xen test bed
Ali-Eldin [25]		x			x								**CT**	**R + P**	**Central**	Hom Generic	Simulation
Zhani [26]	x		x		x			x					**Heuristic**	**R**	**Central**	Hom Generic	Simulation [400]
Han [39]		x	x			x	x		x				**Heuristic**	**R**	**Central**	Hom Generic	IC Cloud
Han [43]		x		x	x								**QT**	**R**	**Central**	Hom Generic	Simulation
Gulati [47]		x				x	x	x	x				**Greedy Heuristic**	**R**	**Central**	Het Generic	ESX test bed

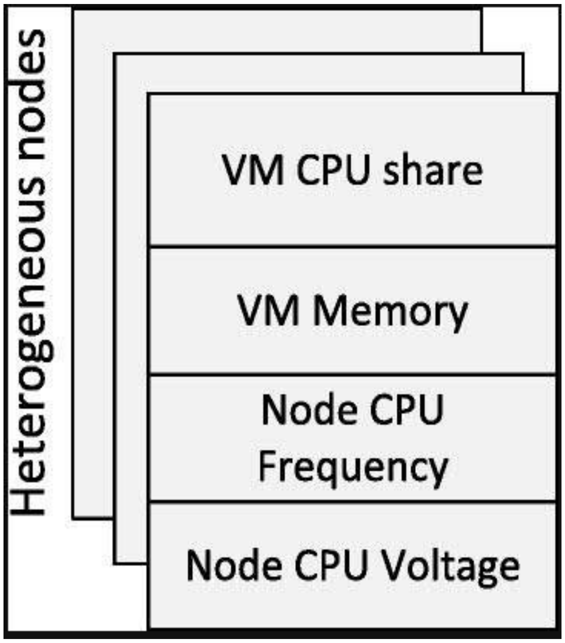

Fig.2 Compute Resource Components

V. SOLUTION PROPOSED

It requires one to identify the same type of customer who use the same type of resources in a region. This can be done using K-means algorithm (according to the literature survey and result analysis in the research articles). Once you get the clusters the VM should be shared across these types of customers on different clusters on timely bases so that there is no overlapping of the VM usage by the customers. But the VM migration should not impact the performance of the systems. There can also be a mechanism in the cloud environment where there is a check on the resource end to get to know the idle time of the resource. By this we get to know how many resources are idle and for how long. By studying these types of resources, we can come to a finding that if the idle time is more than a stipulated time then that resource can be utilized by another task. This can happen internally within the organization to reduce the resource consumption and hence the cost for the consumers. And, this technique would help the IPs to reduce the number of resources provided to a customer. All this is done because of the increase in demand for the cloud resources. Thus, with this mechanism we are ensuring that that resources are utilized to the maximum without compromising even on the quality of service. This would require a machine learning algorithm to be associated or adapted with the monitor which would have the ability to learn and predict the correct idle time for a resource. If a resource is idle for let's say 1 hour every day, then this resource can be redirected to other customer in need. This should be simulated under various factors such as urgent need for the parent, have at least five resources in place with 6 customers in need. The 6th customer is served by the idle times of each of the other resources. A Machine learning algorithm must be applied for this mechanism where the idle time needs to be predicted. The prediction should be at 100% in which case there would not be any comeback for the resource. This study will have many customers' resources in place, must study on for how much time the resource is idle and then this time can be utilized by assigning the resource to a different customer in need. The study also should be done for the customer in place who can finish up the tasks in short period of time. There can also be a case where the customer's needs can be discrete in nature and that the tasks can be carried out for the customer. The mechanism can involve many algorithms which might suffice the requirements. For example, if the customers are equally prioritized then Round Robin method can be incorporated. All these adaptations assume that the mechanism used to interpret the idle time is accurate. This mechanism if incorporated can save resources and thereby power consumption. Once the number of resources is reduced, there is cost savings for the Cloud Service Provider as he might not need to have additional resources. Also, the maintenance cost for the Provider would be saved. This drastically reduces the cost for the Cloud Service Consumer.

Cloud service Consumer would not get to know any of these underlying changes and hence there would be smooth execution of their process. One question arises here as how we exactly predict the idle time and then how do we utilize this time of the resource for other customer. This can be done using the collective idle times of the resources where the other customer can be served. For example, if there are 12 customers using the same kind of resources and there is one more customer who is also in need of the same type of resource as these 12 customers. Then the algorithm will study and predict exactly at what time all these 12 resources would be idle there by adapting that resource to the 13th customer. So, let us assume all the 12 customers' resource would be idle for at least 2 hours in a day, and that the idle time doesn't coincide with other customer's idle time. Then, this customer's resource can be adapted to the 13th resource. Thereby achieving 24-hour service for the 13th resource without interruption. This is well explained in **Table 3**.

Table 3 Customer resources adaptation

Customer Resources	Idle time of customer	13th Resource adapted
1st Customer resource	12pm – 2pm	Yes
2nd Customer resource	2pm – 4pm	Yes
3rd Customer Resource	4pm – 6pm	Yes
4th Customer Resource	6pm – 8pm	Yes
5th Customer Resource	8pm – 10pm	Yes

6th Customer Resource	10pm – 12 am	Yes
7th Customer Resource	12am – 2 am	Yes
8th Customer Resource	2am – 4am	Yes

Table 3*continued*Customer resources adaptation

The idle time of the resource utilization can be made use by using any of the algorithms like Round Robin, First come first serve (FCFS)…

Round robin algorithm is used for a pool of idle resources when I have many resources in place and whose idle time lies continuously with respect to time as shown in the Table 3 above.

If there are many resources which are idle at the same time, then the first idle resource is given to the first customer, 2nd idle resource is given to second customer and so on. This is mainly done at the customer end but then it is monitored by the IPs in place. Various Machine learning algorithms need to be applied at the customer end which will help categorize the type of resources. Once the categorization is done, a predictive algorithm needs to be used which can predict the idle period and time of the resource to an accuracy of 100%. This can be achieved by using trial and error methods for the available algorithms which would at the same time not hinder the existing customer.

VI. OPEN RESEARCH CHALLENGES

Currently there is no such mechanism that the same VM is shared across two different customers and that they would be using it in two different time intervals.

The need for the machine is so random that a dedicated machine is allocated to the customers. In case of shut down or server down time the switch over of servers at different locations is done automatically where the customer is unable to notice the change.

As discussed this can also be done by monitoring the idle time of a resource. The mechanism of predicting the idle time of a resource and then adapting it to another customer needs to be studied. There is also a need to predict at 100% the idle time such that there is no comeback for the resource assigned. The prediction algorithm needs to be studied based on the type of resource. The simulation depends on literature survey, expert knowledge and data availability. So, with the help of all these data one can simulate the required model and see how it works being more effective in terms of power and resources.

VII. CONCLUSION

Thus, if the mechanism proposed is adapted then we can optimize the cloud resource adaptation without the need of additional resources. The automatic reconfiguration of the resources is done implementing a machine learning algorithm which predicts the idle time. The solution proposed uses a mechanism which reduces the need for additional backup. This will in in turn reduce the maintenance cost for the various resources. The mechanism proposed takes care of resource adaption in an optimal way hence reducing the cost for both Cloud Service Provider as there will be minimum resources to be kept as backup resources and for the Cloud Service Consumer as they do not have to pay more. Thereby reducing the cost and power consumption. This proposed solution makes use of Adaptation technique and others to adapt to the changes. Thus, this solution as it mainly focusses on reducing the cost is more efficient than other technique for adaptation. Power consumption on the other hand can be reduced by incorporating the resources for adaption.

REFERENCES

[1]. Jassy A Amazon Web Services Summit. https://aws.amazon.com/summits/san-francisco/. Accessed May **2016**

[2]. Galante G, Bona LCEd "A survey on cloud computing elasticity" In: Proceedings of the **2012** IEEE/ACM Fifth International Conference on Utility and Cloud Computing, UCC '12. IEEE Computer Society, Washington, DC, USA. pp **263–270**

[3]. Beloglazov A, Buyya R, Lee YC, Zomaya A, "A taxonomy and survey of energy-efficient data centers and cloud computing systems". Adv Comput**82:47–111, 2011**

[4]. Botran TL, Miguel-Alonso J, Lozano JA, "Auto-scaling techniques for elastic applications in cloud environments". J Grid Comput**12(4):559–592,2014**

[5]. Najjar A, Serpaggi X, Gravier C, Boissier O, "Survey of Elasticity", Management Solutions in Cloud Computing. In: Computer Communications and Networks. Springer, 236 Gray's Inn Road, Floor 6, London WC1X 8HB, UK. pp **235–263, 2014**

[6]. Jennings B, Stadler R, "Resource management in clouds: Survey and research challenges". J Netw Syst Manag**23(3):567–619**

[7]. Coutinho EF, Carvalho Sousa FR, Rego PAL, Gomes DG, Souza JN, "Elasticity in cloud computing: a survey. Ann Telecommunannales des télécommunications" **70(7):289–309**. doi:10.1007/s12243-014-0450-7,**2014**

[8]. Mann ZA, "Allocation of virtual machines in cloud data centers &mdash;a survey of problem models and optimization algorithms". ACM Computing Survey **48(1):11–11134**. doi:10.1145/2797211,**2015**

[9]. Singh S, Chana I, "Qos-aware autonomic resource management in cloud computing: A systematic review. ACM Computing Survey" **48(3):42–14246**. doi:10.1145/2843889,**2015**

[10]. Faniyi F, Bahsoon R, "A systematic review of service level management in the cloud". ACM Computing Survey **48(3):43–14327**. doi:10.1145/2843890,**2015**

[11]. Naskos A, Gounaris A, Sioutas S, "Cloud Elasticity: A Survey". In: Karydis I, Sioutas S, Triantafillou P, Tsoumakos D (eds). "Algorithmic Aspects of Cloud Computing: First International Workshop", ALGOCLOUD 2015, Patras, Greece, September 14-15, 2015. Revised Selected Papers. Springer, Cham. pp **151–167**,**2016**

[12]. Mohamaddiah MH, Abdullah A, Subramaniam S, Hussin M, "A survey on resource allocation and monitoring in cloud computing". Int J Mach Learn Comput**4(1):31–38,2014**

[13]. Singh S, Chana I, “A survey on resource scheduling in cloud computing: Issues and challenges”. J Grid Comput**14(2):1–48,2016**

[14]. NIST Sp**800-145**: “Definition of cloud computing. Technical report”, NIST, 100 Bureau Drive, Gaithersburg, USA (Sep 2011). NIST. http://csrc.nist.gov/ publications/PubsSPs.html. Accessed May **2016**

[15]. Herbst NR, Kounev S, Reussner R, “Elasticity in cloud computing: What it is, and what it is not”. In: 10th International Conference on Autonomic Computing. pp **23–27,2013**

[16]. Maurer M, Brandic I, Sakellariou R, “Adaptive resource configuration for cloud infrastructure management”. Future General Computing System **29(2):472–487,2013**

[17]. Magklis G, Semeraro G, Albonesi DH, Dropsho SG, Dwarkadas S, Scott ML, “Dynamic frequency and voltage scaling for a multiple-clock-domain microprocessor”. IEEE Micro **23:62–68,2003**

[18]. Addis B, Ardagna D, Panicucci B, Zhang L, “Autonomic management of cloud service centers with availability guarantees”. In: 2010 IEEE 3rd International Conference on Cloud Computing. IEEE, Washington, DC, USA. pp **220–227,2010**

[19]. Sedaghat M, Hernández-Rodriguez F, Elmroth E, “Autonomic resource allocation for cloud data centers: A peer to peer approach”. In: IEEE International Conference on Cloud and Autonomic Computing. IEEE, Washington, DC, USA. pp **131–140,2014**

[20]. Reiss C, Tumanov A, Ganger GR, Katz RH, Kozuch MA, “ Heterogeneity and dynamicity of clouds at scale: Google trace analysis”. In: Proceedings of the Third ACM Symposium on Cloud Computing, SoCC ’**12**. ACM, New York, NY, USA. pp **7–1713**. doi:**10.1145/2391229.2391236** http://doi.acm.org/10.1145/2391229.2391236,**2012**

[21]. Van HN, Tran FD, Menaud J-M, “Sla-aware virtual resource management for cloud infrastructures”. In: IEEE International Conference on Computer and Information Technology. IEEE, Washington, DC, USA **2:357-362,2009**

[22]. Bodík P, Griffith R, Sutton C, Fox A, Jordan M, Patterson D, “Statistical machine learning makes automatic control practical for internet datacenters”. In: Proceedings of the 2009 Conference on Hot Topics in Cloud Computing, HotCloud’09. USENIX Association, Berkeley, CA, USA,**2009**

[23]. Lama P, Zhou X , “Aroma: Automated resource allocation and configuration of mapreduce environment in the cloud”. In: Proceedings of the 9th International Conference on Autonomic Computing, ICAC ’12. ACM, New York, NY, USA. pp **63–72,2012**

[24]. Malkowski SJ, Hedwig M, Li J, Pu C, Neumann D, “Automated control for elastic n-tier workloads based on empirical modeling”. In: Proceedings of the 8th ACM International Conference on Autonomic Computing, ICAC ’11. ACM, New York, NY, USA. pp **131–140,2011**

[25]. Ali-Eldin A, Tordsson J, ElmrothE, “An adaptive hybrid elasticity controller for cloud infrastructures”. In: 2012 IEEE Network Operations and Management Symposium. IEEE, Washington, DC, USA. pp **204–212,2012**

[26]. Zhani MF, Cheriton DR, Zhang Q, Simon G, Boutaba R, “Vdc planner: Dynamic migration-aware virtual data center embedding for clouds”. In: IEEE International Symposium on Integrated Network Management. IEEE, Washington, DC, USA. pp **18–25,2013**

[27]. Roy N, Dubey A, Gokhale A, “Efficient Autoscaling in the Cloud Using Predictive Models for Workload Forecasting”. In: IEEE International Conference on Cloud Computing. pp **500–507,2011** Computing. IEEE, Washington, DC, USA. pp **507–508,2014**

[28]. Zheng S, Zhu G, Zhang J, Feng W, “Towards an adaptive human-centric computing resource management framework based on resource prediction and multi-objective genetic algorithm”. Multimedia Tools and Applications:**1–18,2015**

[29]. Zhang Q, Chen H, Shen Y, Ma S, Lu H, “Optimization of virtual resource management for cloud applications to cope with traffic burst”. FuturGenerComput Syst **58:42–55**. doi:10.1016/j.future.2015.12.011,**2016**

[30]. Padala P, Hou K-Y, Shin KG, Zhu X, Uysal M, Wang Z, Singhal S, Merchant A, “Automated control of multiple virtualized resources”. In: Proceedings of the 4th ACM European Conference on Computer Systems, EuroSys ’09. ACM, New York, NY, USA. pp **13–26,20109**

[31]. Dawoud W, Takouna I, MeinelC,Elastic virtual machine for fine-grained cloud resource provisioning. Glob Trends ComputCommun Syst **269:11–25,2011**

[32]. Fargo F, Tunc C, Al-Nashif Y, Akoglu A, Hariri S, “Autonomic workload and resource management of cloud computing services”. In: IEEE International Conference on Cloud and Autonomic Computing. IEEE, Washington, DC, USA. pp **101–110,2014**

[33]. Beloglazov A, Abawajyb J, Buyya R, “Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing”. Future General Computing Syst **28:755–768,2012**

[34]. Shen Z, Subbiah S, Gu X, Wilkes J, “Cloudscale: Elastic resource scaling for multi-tenant cloud systems”. In: Proceedings of the 2Nd ACM Symposium on Cloud Computing, SOCC ’11. ACM, New York, NY, USA. pp **5–1514,2011**

[35]. Kusic D, Kephart JO, Hanson JE, Kandasamy N, Jiang G, “Power and performance management of virtualized computing environments via lookahead control”. In: Autonomic Computing ICAC. IEEE, Washington, DC, USA. pp **3–23,2008**

[36]. Cardosa M, Korupolu MR, Singh A, “Shares and utilities based power consolidation in virtualized server environments”. In: 11th IFIP/IEEE International Conference on Symposium on Integrated Network Management. pp **327–334,2009**

[37]. Koehler M, “An adaptive framework for utility-based optimization of scientific applications in the cloud”. J Cloud Comput Adv Syst App **3:4,2014**

[38]. Nathuji R, Kansal A, Ghaffarkhah A, “Q-clouds: Managing performance interference effects for qos-aware clouds”. In: Proceedings of the 5th European Conference on Computer Systems, EuroSys ’10. ACM, New York, NY, USA. pp **237–250,2010**

[39]. Han R, Guo L, Ghanem MM, Guo Y, “Lightweight resource scaling for cloud applications”. In: International Symposium on Cluster, Cloud and Grid Computing. IEEE, Washington, DC, USA. pp **644–651,2012**

[40]. Zhu X, Wang Z, Singhal S, “Utility-Driven Workload Management Using Nested Control Design”. In: American Control Conference. IEEE, Washington, DC, USA,**2006**

[41]. Xu J, Zhao M, Fortes J, Carpenter R, Yousif M, “Autonomic resource management in virtualized data centers using fuzzy logic-based approaches”. ClustComput**11:213–227,2008**

[42]. Jamshidi P, Ahmad A, Pahl C, “Autonomic resource provisioning for cloud-based software”. In: Proceedings of the 9th International Symposium on Software Engineering for Adaptive and SelfManaging Systems, SEAMS 2014. ACM, New York, NY, USA. pp **95–104,2014**

[43]. Han R, Ghanem MM, Guo L, Guo Y, Osmond M, “Enabling cost-aware and adaptive elasticity of multi-tier cloud applications”. FuturGenerComput Syst **32:82–98,2014**

[44]. Hasan MZ, Magana E, Clemm A, Tucker L, Gudreddi SLD, “Integrated and autonomic cloud resource scaling”. In: Network

Operations and Management Symposium. IEEE, Washington, DC, USA. pp **1327–1334,2012**

[45]. Gmach D, Rolia J, Cherkasova L, Kemper A "Resource pool management: Reactive versus proactive or lets be friends". Computer Networks: The International Journal of Computer and Telecommunications Networking **53:2905–2922,2009**

[46]. Iqbal W, Dailey MN, Carrera D, Janecek P, "Adaptive resource provisioning for read intensive multi-tier applications in the cloud". FuturGenerComput Syst **26:871–879,2011**

[47]. Gulati A, Shanmuganathan G, Holler A, Ahmad I, "Cloudscale resource management: Challenges and techniques". In: Proceedings of the 3rd USENIX Conference on Hot Topics in Cloud Computing, HotCloud'11. USENIX Association, Berkeley, CA, USA. pp **3–3,2011**

[48]. Tchana A, Palma ND, Safieddine I, Hagimont D, Diot B, Vuillerme N, "Euro-par 2015: Parallel processing: 21st international conference on parallel and distributed computing", Vienna, Austria, August 24-28, 2015, proceedings: **305–316,2015**

[49]. Beloglazov A, Buyya R, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers". ConcurrComputPract Experience **24:1397–1420,2012**

[50]. Choi HW, Kwak H, Sohn A, Chung K, "Autonomous learning for efficient resource utilization of dynamic vm migration". In: Proceedings of the 22Nd Annual International Conference on Supercomputing, ICS '08. ACM, New York, NY, USA. pp **185–194,2008**

[51]. Zuo L, Shu L, Dong S, Zhu C, Zhou Z, "Dynamically weigh0ted load evaluation method based on self-adaptive threshold in cloud computing". Mob Networks Appl :1–15. doi:10.1007/s**11036-016- 0679-7,2016**

[52]. Lim HC, Babu S, Chase JS, "Automated control for elastic storage". In: Proceedings of the 7th International Conference on Autonomic Computing, ICAC '10. ACM, New York, NY, USA. pp **1–10,2010**