IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 3, 2019, accepted June 21, 2019, date of publication July 2, 2019, date of current version July 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926291

Context Aware VM Placement Optimization Technique for Heterogeneous IaaS Cloud

ASHWIN KUMAR KULKARNI AND B. ANNAPPA, (Senior Member, IEEE)
Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal 575025, India

Corresponding author: Ashwin Kumar Kulkarni (ashwin.kulkarni7@gmail.com)

ABSTRACT Ever increasing demand for cloud adoption is prompting researchers and engineers around the world to make cloud computing more efficient and beneficial for cloud service providers and users. Cloud computing brings profits for all when the cloud infrastructure is used efficiently, and its services are made affordable to businesses of all scales. Managing cloud data center incurs a significant cost, which includes investing in IT infrastructure at the beginning and data center management costs for power, repair, space, and so on at later stages. The power costs are contributing to a significant share in overall data center management costs, and saving in power consumption can help reduce management costs for data center owners. This paper proposes an efficient context-aware adaptive heuristic-based solution for the virtual machine (VM) placement optimization in the heterogeneous cloud data centers. The proposed VM placement technique takes into the account of physical machine characteristics and load (peak and non-peak) conditions in the heterogeneous data centers to save power and also improve performance efficiency for data center owners. The experiments conducted with real cloud workloads and also synthetic workloads against a well-known adaptive heuristic-based technique indicate significant performance improvements and energy saving with our proposed solution.

INDEX TERMS Cloud resource management, VM placement optimization, cloud VM load balancing.

I. INTRODUCTION

Cloud computing is being adopted very rapidly by businesses of all scales because of its lucrative benefits. Cloud computing helps businesses avoid higher investment for IT infrastructure at the beginning and takes away the hardware maintenance costs, administration costs, and worries from its users. Cloud computing services are offered by multiple vendors in three distinct models such as Infrastructure-as-a-Service, Platform-as-a-Service, and Software-as-a-Service to its users based on their business needs. Cloud data centers are a farm of heterogeneous computing servers that are provisioned dynamically to user applications. The virtualization technology adopted by cloud enables data centers to maximize the utilization by sharing of physical computing nodes between users/applications and also scales computing infrastructure based on the demand from users. If cloud computing has to succeed, it has to be very efficient and cost effective for both cloud service providers and its users.

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Anwar Hossain.

Variety of services are being offered in cloud platform at an ever-increasing pace, and the users spread across the globe consumes its services. The cloud service providers are building their data centers at geo-distributed locations to cater to users located at different geo-regions to improve performance, fault tolerance, and also to provide reliable services round the clock. The cloud service providers make a significant investment at the beginning to set up data centers for IT infrastructure and other logistics, and later they incur significant data center management costs to keep their data centers running. The data center management costs include power/electricity costs, hardware & software maintenance costs, and other logistics costs. The data center management costs vary greatly based on the total power usage, electricity cost at that location, and space renting costs. As per a published study [14], the electricity/power costs contribute to around 13% of the overall cost for data center management, which is a significant share contributing to cost of data center owners. It is vital to reduce the power consumption of the data center whenever and wherever there is scope without affecting the cloud application performance to reduce operations cost for data center owners.

The data center is a server farm consisting of a large number of heterogeneous physical machines connected by a high speed shared network. These physical machines often tend to vary in terms of their computing capacity, composition, and also in their power consumption characteristics at different load conditions. Such heterogeneity in the composition of physical machines results in some of these machine being more power efficient than others during their operation in the data center. It is essential to optimize power consumption in the data center by efficiently scheduling VMs on more power and performance efficient physical machines while identifying and switching off other physical machines with lower power efficiency and having lower utilization during non-peak hours.

In this paper, the problem of optimizing power consumption by efficient utilization of heterogeneous physical machines in a data center having an inherent variability in power consumption and performance metrics are evaluated and also optimizing overheads of load balancing algorithms considering data center load parameter is investigated. Heterogeneity in physical machines power and performance characteristics along with data center load conditions are denoted in our proposed work as data center context parameters. Paper presents a context-aware VM placement optimization technique to reduce the cost of data center management by optimizing power consumption and enhancing performance without affecting the response times of applications.

Summary of contribution by proposed work is listed as follows;

i Investigation of the problem of minimizing the power consumption of the overall data center by taking into consideration varying power consumption and performance profile of heterogeneous physical servers.

ii Work proposes an efficient context adaptive VM placement optimization technique by considering power and performance characteristics of physical machines at different load levels in a heterogeneous data center.

iii Paper also proposes a data center load condition aware load balancing overhead avoidance algorithm to reduce VM placement time.

iv Finally, the proposed techniques are evaluated using the real world planet lab datasets and also synthetic workloads; the results are compared with existing well known adaptive heuristics based method to prove the effectiveness of the proposed solution.

The rest of the paper is organized as follows. The next section mentions some of the noted past contributions of researchers in solving a similar problem; section 3 describes the problem our work attempted to solve followed by the proposed system architecture and components presented in section 4.The section 5 explains the implementation of proposed context-adaptive VM placement optimization. Section 6 summarizes results obtained during the experimental evaluation of the proposed scheme with various performance metrics, and the paper presents our conclusion at the end.

II. RELATED WORKS

The power and cost optimization in cloud datacenters is an active research area in cloud computing domain, and it may remain so for years to come given the stochastic nature of the problem that it is dealing with. Many important algorithms have been proposed to reduce the cost of data center management by optimizing resource utilization and exploiting power saving opportunities. The following section discusses some of the works that motivated our research and are in line with our research direction.

Anton and Rajkumar proposed an adaptive heuristics based performance efficient and energy saving technique [4] for dynamic consolidation of VMs in cloud data centers. They presented competitive analysis and proved competitive ratios of optimal online deterministic algorithms. Authors addressed problems of VM migration and dynamic VM consolidation. Paper proposed a novel solution for dynamic consolidation of VMs based on analysis of historical data from the resource usage by VMs and power usage statistics of host machines to arrive at the VM placement decisions. Our proposed solution is evaluated against the adaptive heuristics based technique [4] proposed by the authors.

Yi-Ju Chiang *et al.* proposed a novel technique to utilize server idle power in the data center to minimize operational costs. The paper first studied the problem of controlling service rates and optimizing the operational cost of data centers. Authors formulated a three parameter cost function that takes into account costs of power consumption, system congestion, and server startup. A green control algorithm [6] was proposed to solve the constrained optimization problem of cost saving and to make costs versus performances tradeoffs in physical machines with different power-saving policies without violating the performance SLAs promised to users.

Adel Nadjaran Toosi *et al.* investigated a profit-maximizing technique for cloud service providers by optimizing the allocation of data center capacity to each pricing plan utilizing admission control for resource reservations. Authors proposed an optimization technique based on the formulation of stochastic dynamic programming [7] and two heuristics that consider trade-offs between computational complexity and optimality. The proposed technique is evaluated using real workload traces of Google to prove the effectiveness of the solution.

Ismael Solis Moreno *et al.* proposed a performance interference aware virtual machine placement strategy [16] to avoid performance bottlenecks caused by non-compatible VMs co-hosted on the same servers in data centers because of resource contentions. The paper proposes a novel technique for workload allocation for energy efficiency by considering the VM workload characteristics and host internal interference levels to select the suitable physical host for the given workload.

Yuanxiong Guo *et al.* proposed a technique to utilize energy storage available in data centers to reduce the overall electricity costs in the wholesale electricity markets, where the price of electricity varies both spatially and temporally.

The technique proposed integrates center-level load balancing with the server level configuration, and battery management and also at the same time ensures the quality-of-service for users. The paper utilizes Lyapunov optimization [1] to achieve a tradeoff between energy storage and cost saving.

Rahman *et al.* studied the power management problem of data center operations and summarized motivation, various aspects that influence the power costs, the current state of art technologies and methods proposed to improve the power management in the datacenter. The paper also proposes to utilize smart grid [2] environment to ensure efficient and dynamic power management solution for data centers.

Abdelkhalik *et al.* developed an energy and SLA-aware VM placement strategy, which dynamically assigns virtual machines to physical servers in a cloud environment. Abdelkhalik *et al.* formulated the VM placement problem using utility functions [3]. Paper proposes a genetic algorithm to search VM-PM assignments that maximize the utility function formulated for VM placement problem. The technique proposed co-optimizes SLA violations and power consumption. The performance results presented performs better than the well-known heuristics based approach.

Our previous work [8] has investigated the problem of cost saving in the geo-distributed data center scenario by exploiting the nature of varying power tariffs across geographies. The work aimed to reduce the electricity cost of data centers by proposing an electricity cost aware service broker technique that can route user requests to the cost-effective data center without compromising on the response times of applications.

The objective of minimizing power consumption in the data center has also been addressed using various techniques/algorithms such as PSO based [18], heuristics based [4], [17], best fit decreasing [19] and graph theory algorithms [20]. Some of the past works also attempted to solve VM placement optimization for network traffic minimization in the data center using techniques such as Ant colony optimization [21] and greedy based schemes [22]. The VM placement optimization problem is also addressed for ensuring QoS for users at all times by using Integer programming [5] technique and to also meet hybrid objectives such as maximizing resource utilization and reduce communication traffic using automata based schemes [23].

The problem of VM placement optimization has been addressed in the past using different approaches/algorithms to achieve different desired objectives as discussed above. Table 1 summarizes these important related works with their primary mechanism and goals achieved by each one of them.

III. PROBLEM DEFINITION

In this paper, proposed work investigated various parameters that constitute the context of the data center and proposed an efficient load balancing scheme to optimize power consumption and cost for the benefit of the data center owners without affecting the response times of applications.

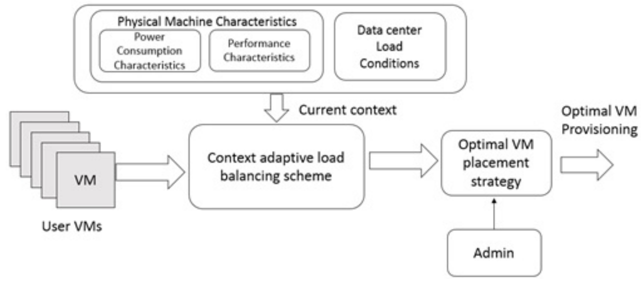

FIGURE 1. System block diagram of proposed solution.

The proposed system considers physical machine characteristics such as performance to power ratio, which signifies the physical machine's power efficiency and data center load conditions at that point of time to arrive at the VM provisioning decisions. Our power consumption optimization problem is formulated as below,

$$Ptotal(t) = \sum_{i=0}^{N} Po(t)(i)(l) \tag{1}$$

Ptotal denotes the total power consumption of cloud infrastructure at time t, N denotes the number of physical machines at each data center. The $Po(t)(i)(l)$ corresponds to power consumption of i^{th} machine having CPU load of l percentage at time t. The objective of our proposed work is to optimize the value of Ptotal by considering current data center context parameters for making VM placement and reallocation decisions without compromising on the response times of user applications.

IV. PROPOSED SYSTEM MODEL

The targeted environment for our proposed model is the cloud IaaS system in a large datacenter with N heterogeneous machines. Each node is characterized by major resources such as CPU, main memory, network, and NAS (network attached storage) for storage requirements. The proposed system has no prior knowledge of application workloads and VM placement details. The geo-distributed users of such a cloud system submit their VM placements requests which correspond to a dynamic mix of different application workloads that may be co-hosted on a single physical server in a cloud data center. The software layer for context adaptive load balancer consists of two distributed modules. These modules help central load balancer to capture context details at the physical machine level and datacenter level for efficient VM provisioning and re-provisioning. These modules are explained in the following subsections.

A. LOCAL CONTEXT MANAGER

The local context manager shown in figure 2 works at every physical server in the cloud and the same level as a hypervisor.

TABLE 1. Summary of related past works in VM placement and optimization.

Serial No	Primary Mechanism	Authors,Publication year	Goal of proposed work
1	Adaptive Heuristics based	Anton and Rajkumar,2012	Minimizing total energy consumption of datacenter.
2	N-Policy Green control algorithm	Yi-Ju Chiang et. al,2015	Optimizes operational cost of datacenter and ensures SLA guarantee.
3	Dynamic programming and heuristics based	Adel Nadjaran Toosi et.al,2015	Maximizing profit for datacenter owners.
4	PSO based	Seyed Ebrahim Dashti and Amir Masoud Rahmani,2015	Minimizing energy consumption and ensures QoS for users.
5	Heuristics based	Li, X.et. al,2013	Minimizing total energy consumption.
6	Best fit decreasing	Noumankhan Sayeedkhan, P. and S. Balaji,2014	Minimizing total energy consumption.
7	Graph theory based	Xiao, Z., et al., 2015	Minimizing total energy consumption.
8	ACO based	Dong, J.-k., et al.,2014	Reduce total communication traffic in the data center network.
9	Greedy algorithm based	Kanagavelu, R., et al.,2014	Reduces amount of inter-VM traffic and network load.
10	Integer programming	Li, W., J. Tordsson, and E. Elmroth,2012	Ensures QoS for users.
11	Automata based	Liu, C., et al,2014	Maximize resource utilization and minimize communication traffic.
12	Lyapunov Optimization	Yuanxiong Guo and Yuguang Fang,2013	Minimizing overall electricity costs in variable pricing electricity market.
13	Genetic algorithm based	Abdelkhalik et.al,2015	Minimizing power consumption and SLA violations.
14	Interference aware algorithm	Ismail Solis Moreno et.al,2013	Minimizing energy consumption and performance aberrations.
15	Power cost aware algorithm	Ashwin kumar and Annappa,2017	Minimizing power cost and ensuring response time guarantee.
16	Context aware adaptive Heuristics based	Ashwin kumar and Annappa,2019 (This article)	Minimizing total energy consumption of datacenter.

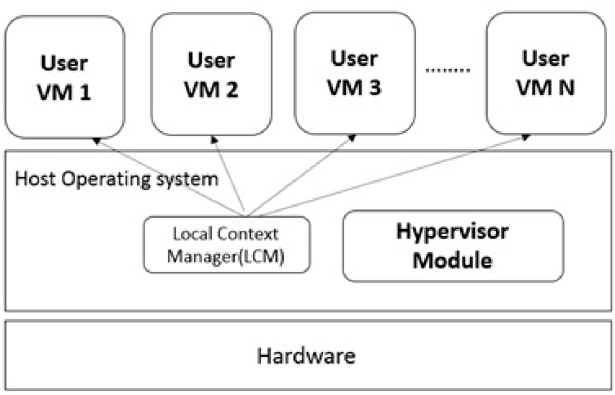

FIGURE 2. Local context manager architecture.

The local context manager collects information about all co-located VMs running on its physical machines such as resource utilization statistics of each VM, machine resource availability statistics, and its physical machine characteristics and runtime power consumption details. The local context manager also holds the resource utilization history of its host. The global workload scheduler accesses these details for effective VM placement decisions.

B. GLOBAL WORKLOAD SCHEDULER

The Global workload scheduler (GWS) shown in figure 3 works at the central load balancer server and it works in tandem with local context manager to detect the current local and global context to enable dynamic load balancing in data centers. The GWS is responsible for detecting the current load conditions (peak and nonpeak) using host utilization details preserved with LCM. It is also responsible for issuing commands for context adaptive VM placement optimizations at regular interval. The communication and sharing between

TABLE 2. Power and performance metrics from SPECPower benchmark.

Target Load %	10%		20%		30%		40%		50%		60%		70%		80%		90%		100%		Avg
PM Type	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P	P2P	P2P
HP ProLiant ML 110 G4	89.4	63.9	92.6	116	96	170	99.5	222	102	262	106	313	108	350	112	394	114	430	117	467	268
HP ProLiant ML 110 G5	97	102	101	195	105	282	110	354	116	426	121	494	125	554	129	618	133	679	135	731	431
HP ProLiant ML 110 G3	112	47.9	118	89.4	125	128	131	160	137	191	147	218	153	241	157	268	164	285	169	309	190
IBM Server x3250	46.7	665	52.3	1205	57.9	1621	65.4	1930	73	2143	80.7	2361	89.5	2466	99.6	2539	105	2685	113	2767	2098
IBM Server x3550 XeonX5675	98	917	109	1651	118	2274	128	2793	140	3201	153	3497	170	3680	189	3805	205	3929	222	4009	3093
IBM Server x3550 XeonX5670	107	861	120	1528	131	2094	143	2568	156	2933	173	3200	191	3363	211	3491	229	3603	247	3694	2843

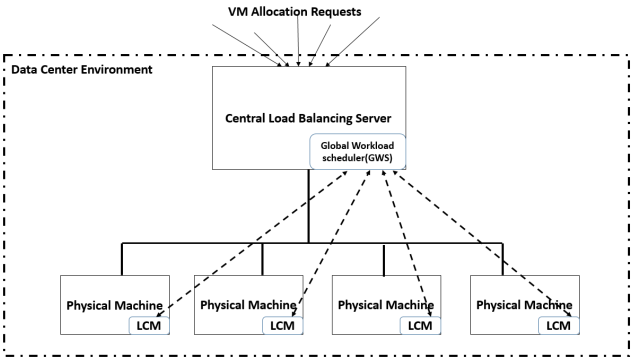

FIGURE 3. Global workload scheduler architecture.

LCM and GWS are enabled by the sharing of common data structures and by using regular information polling.

C. POWER EFFICIENCY OF PHYSICAL MACHINES

The power consumption of physical machines is collective sum of power consumption of CPU, RAM, Disk, Power supply unit and cooling unit but past studies [9], [10] have observed that there exists a linear relationship between power consumption and CPU utilization. However, because of evolving modern multi-core CPUs and virtualization, the modern servers contain large RAMs which starts to consume significant share in power consumption by its server. Also, coupled with the difficulty of modeling power consumption of multi-core CPUs make building an accurate analytical model for power consumption, a complex research problem [4]. Therefore, instead of building an analytical model for power consumption and performance efficiency of physical machines, proposed work uses real data on energy consumption and performance metrics provided by the published results of the SPECpower benchmark [11]. The data center consists of a heterogeneous set of machines which vary in terms of power consumption and throughput. We measure the power efficiency of each physical machine by using the ratio of throughput(NumOps) to the power consumed(Pc) at different load levels, and an average of these values is considered as performance to power ratio of the physical machine.

$$PerfToPowerR(L\%) = NumOps(L\%)/P_c(L\%) \qquad (2)$$

The equation (2) shows the calculation of performance to power ratio of the physical machine at a specific load level on CPU. Then using different load levels data of *PerfToPowerR* ratio, average *PerfToPowerR* can be calculated as in (3) where N indicates total number of different load levels considered and *PerfToPowerR(Li)* indicates performance to power ratio of physical machine at specific load level on CPU at instance i calculated in (2).

$$AverPerf2Pow = 1/N \sum_{i=0}^{N} PerfToPowerR(Li) \qquad (3)$$

The *AverPerf2Pow* is used as an indicator of power and performance efficiency for the physical machine; a higher value of *AverPerf2Pow* indicate higher efficiency of the physical machine.

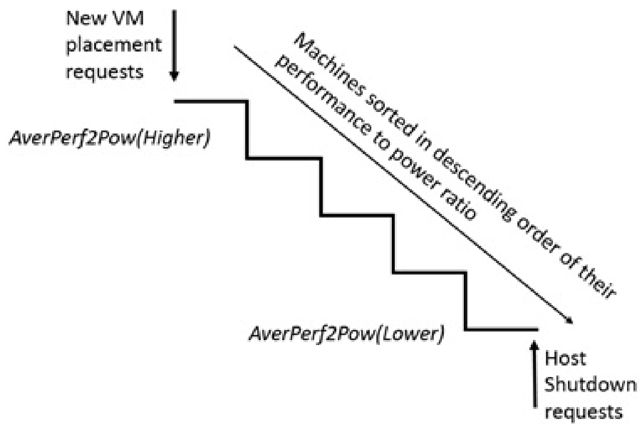

FIGURE 4. Proposed VM placement and host shutdown process.

Table 2 lists the power and performance metrics for commonly used physical machine (server) types and proposed work utilizes these server configurations in performance evaluation presented at the end. The P indicates power consumption and P2P indicate performance to power ratios at different load levels. The Avg P2P indicates average performance to power ratio of physical machines. Proposed technique prioritizes physical machines with higher *AverPerf2Pow* for provisioning during VM allocation/re-allocation requests. During non-peak hours, physical machines with lesser *AverPerf2Pow* are prioritized for power-off to ensure power efficient machines are always used to save power. Figure 4 shows the process of prioritizing hosts based on their power efficiency during new VM placement requests and when host shutdown requests are received. The power efficiency of the physical servers is modeled using the Spec Benchmark [11] data published for several types of servers available in the market.

D. LOAD CONTEXT BASED ALGORITHM ADAPTATIONS

The VM placement optimization algorithm has to check each physical machine state at regular intervals for overload condition and underload condition for re-assigning the VMs to meet performance SLAs guarantee and also to save power. In the process of achieving the objective, the VM placement optimizer algorithm also utilizes significant compute power and CPU time, so it becomes essential to optimize the VM placement(load balancer) algorithm to eliminate the checks that are redundant when we consider the current context of the data center. Proposed work investigates the modification of VM placement optimization algorithm to avoid underload detection of each host when peak traffic(load) situation is detected in the data center to avoid unnecessary host switch offs and VM migrations thereby attempting to save significant compute power and compute time. Figure 5 illustrates the algorithm modification proposed for optimizing VM placements in the data center. The host underload detection step is skipped when datacenter is experiencing heavy traffic of computing/service requests.

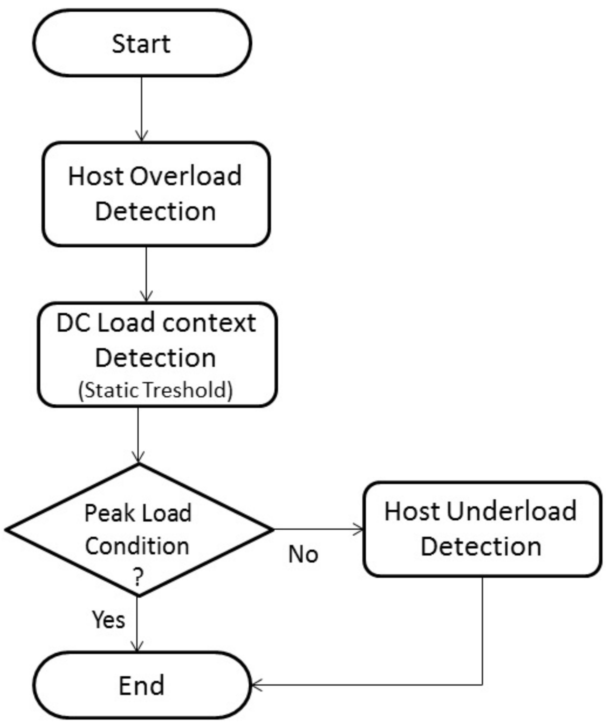

FIGURE 5. Load context aware VM placement optimization process.

V. CONTEXT AWARE VM PLACEMENT OPTIMIZATION

In this section, we propose a context-aware VM placement optimization framework for dynamic VM placement with the sole aim of cost optimization for datacenter by reducing power consumption without any performance penalty for user applications. The problem of dynamic VM placement optimization can be generally split into 4 subproblems,

1) Host Overload Detection: determining when the host is considered over utilized and is affecting the performance of one or more VMs residing on it, hence requiring migration of some VM out of it.
2) Host Underload Detection: determining when the host is underutilized with power wastage because of idling of resources, hence all VMs residing on it can be migrated so that host can be switched off to save power.
3) VM Selection: the process of selection of VM that can be migrated out of the over-utilized(overloaded) host.
4) VM Placement: the process of searching for a new host to migrate the VMs selected from the overloaded or under-loaded hosts.

The global workload scheduler(GWS) is responsible for carrying out VM placement optimization in our proposed framework, and it regularly checks each host for overload and underload conditions by communicating with local context managers(LCM) of each host. The proposed work defines the VM optimization scheduling interval as 5 minutes, which is a similar interval used in distributed resource scheduler (DRS) of VMware [3].

A. VM PLACEMENT OPTIMIZATION

The proposed algorithm for VM placement optimization invoked by GWS at regular time intervals can be specified as in algorithm 1.

The proposed algorithm first checks each host(PM) for host overload detection and then collects the VM migration list to move out VMs from overloaded hosts. The VMs to be migrated are found a suitable host for migration from overloaded hosts. Then proposed algorithm checks the current context of the datacenter for peak load situation, if it is peak load situation, then algorithm skips the underload detection of hosts to avoid unnecessary host switch offs and VM migrations. If it is a normal load condition at the data center, the algorithm performs the host underload detection for all hosts and prepares the VM migration lists from underloaded machines that are going to be switched off.

B. PROPOSED VM PLACEMENT ALGORITHM

The algorithm for VM placement called power and performance-aware best fit decreasing VM placement technique, a modified version of the Power Aware Best Fit Decreasing (PABFD) algorithm [4] is proposed as in algorithm 2.

The algorithm ensures that the host machine(PM) with higher performance to power ratio is checked for VM allocation at first to aid power efficient processing in DC. The algorithm 2 returns with VM to PM allocations, which are efficient in terms of power and performance efficiency.

C. HOST UNDERLOAD DETECTION

The under-loaded host detection and switch off process is essential in the data center to save power when the load on the data center is not high. However, the host power off algorithm should take care of the performance efficiency of

Algorithm 1 Proposed VM Placement Optimization

```
Result: Prepares new VM re-placement request list
Input  : pmList
Output: migrationList
vmsForMigration = 0;//Temp List of migrating VMs
foreach pm ∈ pmList do
    /* Identify VMs to be migrated from
       Overloaded Hosts                       */
    if isHostInOverloadedCondition(pm) then
        vmsForMigration.add(
        getVmsToMigrateFromOverloadedPM(pm) );
        migrationList.add(
        getNewVmPlacements(vmsForMigration) );
        vmsForMigration.clear();
    end
    /* Query current DC load context */
    if isNonPeakSituationInDc() then
        /* select VMs from underloaded
           Hosts for migration                */
        foreach pm ∈ pmList do
            if isHostInUnderloadedCondition(pm) then
                vmsForMigration.add(pm.getVmList());
                migrationList.add( getNewVmPlace-
                ments(vmsForMigration));
            end
        end
    end
end
return migrationList;
```

the host machine also in consideration to maximize the power saving. When datacenter is experiencing lesser workload requests, VM placement optimization algorithm should consider switching off host machines with low power efficiency to maximize power saving benefit. The algorithm 3 explains our proposed technique for host underload detection and host selection for power off that takes into account the performance to power ratio of host machines.

Algorithm 3 ensures that the host with utilization lesser than minUtilization and which is not performance and power efficient is switched off, thereby saving power in a non-peak situation in the data center. The switching of the host is performed only after all the VMs running on the said underutilized host are migrated successfully.

D. LOAD CONTEXT DETECTION IN DATACENTER

The load context detection in the data center is vital for optimizing resource management decisions. The algorithm 4 describes the data center load condition detection used in our proposed work. The algorithm utilizes data stored by a local context manager (LCM) for each host such as VMs running on hosts and their MIPS utilization to arrive at the overall host

Algorithm 2 Power and Performance Aware BFD

```
Result: Power and performance efficient VM-PM
        mapping list
Input  : pmList,vmList
Output: VMAllocationList
vmList = SortByCPUUtilizationDecreasing(vmList);
pmList =
SortByAvrPerf2PowerRatioDecreasing(pmList);
foreach vm ∈ vmList do
    minPower = MAX_VALUE; PMAssigned = NULL;
    foreach pm ∈ pmList do
        if isSuitablePM(pm,vm) then
            power = estimatedPower(pm,vm); if power
            < minPower then
                PMAssigned = pm; minPower = power;
            end
        end
    end
    if PMAssigned ! = NULL then
        VMAllocationList.add(vm, PMAssigned);
    end
end
return VMAllocationList;
```

Algorithm 3 Underloaded Host Detection Algorithm

```
Result: Selects underutilized and power inefficient Host
        for shutdown
Input  : pmList
Output: underUtilizedHost
/* Initialize to static threshold
   value for under utilization check
   */
minP2PRatio = MAX_VAL;
minUtilization = LOWER_TRESHOLD;
/* Select power inefficient Host with
   lower than threshold CPU
   utilization                              */
foreach pm ∈ pmList do
    utilization = pm.getUtilizationOfCpu();
    if (utilization > 0) && (utilization < minUtilization)
    then
        pwModel = pm.getPowerModel();
        power2PerfRatio =
        pwModel.getPerf2PowerData();
        if power2PerfRatio < minP2PRatio then
            underUtilizedHost = pm;
            minP2PRatio = power2PerfRatio;
        end
    end
end
return underUtilizedHost;
```

Algorithm 4 DC Load Context Detection Algorithm

```
Result: Detects load context of data center
Input : pmList
Output: isPeakSituationFlag
TotalHostUtilizationsInDc = 0;
AverageHostUtilizationsInDc = 0;
isPeakSituationFlag = FALSE;
/* Measure total DC MIPS utilization
   */
foreach pm ∈ pmList do
    utilization = 0;
    foreach vm ∈ pm.getVmList() do
        utilization = utilization +
        vm.getMipsUtilization();
    end
    TotalHostUtilization = TotalHostUtilization +
    utilization;
end
AverageHostUtilizationsInDc =
TotalHostUtilization / NumHostsInDc;
if AverageHostUtilizationsInDc >
MAX_UTIL_THR_DC then
    isPeakSituationFlag = TRUE;
end
return isPeakSituationFlag;
```

CPU utilization. Once host utilization data is summed up for all hosts in the data center, the proposed solution calculates the average utilization of data center servers, if the data center has an average utilization of over MAX_UTIL_THR_DC then algorithm designates current context as peak load situation else it is considered the normal/non-peak situation in the data center. The TotalHostUtilization is calculated by the local context manager(LCM) at each host and provided to global workload scheduler(GWS) to measure AverageHostUtilizationsInDc and to set isPeakSituationFlag.The algorithm 4 is utilized in algorithm 1 to optimize VM placements.

VI. PERFORMANCE EVALUATION

Evaluation of our proposed context adaptive solution for power and cost saving has been carried out against another adaptive heuristics based technique [4] proposed by Beloglazov *et al*.

A. PERFORMANCE METRICS

Table 3 lists the various performance metrics used in our evaluation to prove the performance improvement that our proposed solution brings in.

B. EXPERIMENTAL SETUP

The Cloudsim toolkit [13] is utilized for the evaluation of our context adaptive scheme for power and cost saving. Clousim toolkit is a widely used simulation tool among cloud researchers, and it provides a scalable and layered simulation framework that helps modeling, simulation, and evaluation of emerging cloud computing architectures and applications before deployment. The simulation of the cloud data center is carried out using a composition of six different types of physical machines configurations shown in table 4.

All the experiments are conducted on an HP Probook laptop (with Core i5 CPU with 8 GB RAM) running Windows 7 OS. The simulation duration is set to one day, which is a similar duration chosen to evaluate heuristics based solution [4]. Our proposed algorithm is invoked every 5 minutes once, which is a similar duration chosen for VMWare distributed resource scheduler [3] (DRS) to check and re-adjust resource mappings. Two Experiments were conducted with two different nature of workloads and varying datacenter resource compositions to evaluate our proposed context adaptive solution. The objective of our first experiment is to validate the effectiveness of our proposed solution against synthetic workloads with a variable number of VMs ranging from 100-400 VMs to create a lightly loaded scenario to heavily loaded scenario in a data center with 100 PMs of two different server(PM) types. The experiment composition chosen is to ensure that our proposed solution is useful in all

TABLE 3. Performance metrics used for evaluation.

Serial No	Performance Metric	Description
1	Energy Consumption	The metric indicates the total power consumed by all physical machines in the data centers. Lower power consumption leads to cost savings for data center owners.
2	Overall SLA Violations	SLA Violations arise from the non-optimal mapping of VMs to PMs and are usually resulted from overutilization of servers and many migrations involving the same VM.
3	Total VM Migrations	VM migrations indicate a number of re-mappings of VMs done among available PMs during the given time. A higher number of migrations may cause performance degradation and a low number of VM migration may mean non-adaptive assignment with respect to data center situations.
4	Total Host(PM) Shutdowns	The metric shows a number of times the hosts(PMs) were shut down. PM shutdowns save power for data center but frequent shutdowns may mean higher energy consumption because of PM start-up procedures and also will lead to failure of the machine in the long run.
5	Mean VM Allocation time and Mean Host selection time	The metric helps to evaluate the running time of our proposed solution against related past work.

TABLE 4. Physical machine configurations in data center.

Serial No	Machine Model Name	MIPS	Num cores	RAM Size(in MBs)	Network BW(in GBs)	Type
1	HP Proliant ML 110 G4	1860	2	4096	1	Small
2	HP Proliant ML 110 G5	2660	2	4096	1	Small
3	HP Proliant ML 110 G3	3000	2	4096	1	Medium
4	IBM server x3250	3067	4	8192	1	Medium
5	IBM server x3550[Xeon-X5675]	3067	6	16384	1	Big
6	IBM server x3550[Xeon-X5670]	2933	6	12288	1	Big

TABLE 5. Virtual Machines(VM) configurations used in DC.

Serial No	VM Type	[CPU_MIPS,num_cores, RAM_in_MBs, VM_Size_in_GBs]
1	Type 1 [Extra Big]	[2500,1,870,2.5]
2	Type 2 [Big]	[2000,1,1740,2.5]
3	Type 3 [Small]	[1000,1,1740,2.5]
4	Type 4 [Extra-Small]	[500,1,613,2.5]

load cases. The goal of our second experiment is to appraise the effectiveness of our proposed context adaptive solution against a real system PlanetLab workload traces [12], [15] containing CPU utilization data of 1033 VMs. The real workload is evaluated using a datacenter composing 400 PMs of up to six different host types.

C. EXPERIMENT 1: SYNTHETIC WORKLOAD WITH VARYING NUMBER OF VMS

The aim of experiment 1 is to evaluate the effectiveness of our solution with different load conditions (lightly to heavily loaded) in a data center. Five different configurations are chosen for testifying different load conditions.

- *Configuration 1.1: 100 VMs to be allocated to 100 PMs*
- *Configuration 1.2: 200 VMs to be allocated to 100 PMs*
- *Configuration 1.3: 250 VMs to be allocated to 100 PMs*
- *Configuration 1.4: 300 VMs to be allocated to 100 PMs*
- *Configuration 1.5: 400 VMs to be allocated to 100 PMs*

The simulation of the cloud data center is done using two types of host machines of type HP ProLiant ML110 G4 and IBM server x3250 with configurations shown in Table 3 and all VM types shown in Table 4 are used to create VMs. Cloudlets are programmed to generate utilization data every five minutes based on the stochastic model [13]. The energy consumption results of the proposed solution, along with a heuristic-based solution [4] proposed by Beloglazov *et al.* are shown in Figure 6.

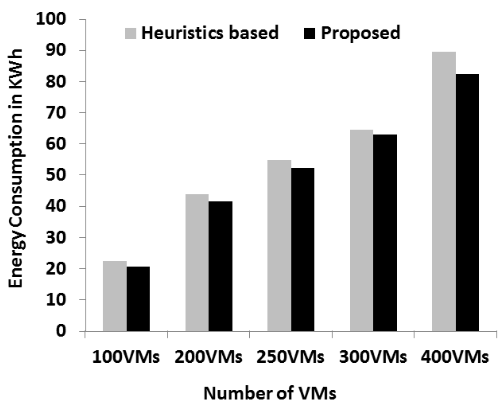

FIGURE 6. Power consumption results of experiment 1.

Figure 6 shows the energy consumption profile of heuristic-based solution and our proposed solution for 5 configurations of workload. Results indicate that our proposed solution saves approximately 8-10% energy during lightly and heavily loaded cases and 2-6% during moderately loaded cases. The results of all performance metrics are tabulated in Table 6.

The graphs plotted for total VM migrations in Figure 7 shows that the VM migrations increase with the increase in VM load conditions in case of heuristic-based method whereas with the proposed solution, observed VM migrations are almost same and much smaller in number.

The figure 8 and figure 9 shows that overall SLA violations observed and total host shutdowns during

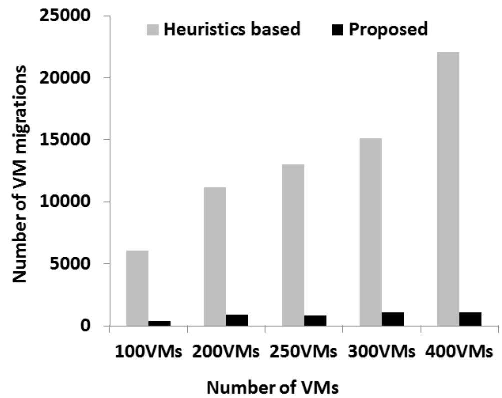

FIGURE 7. VM migrations results of experiment 1.

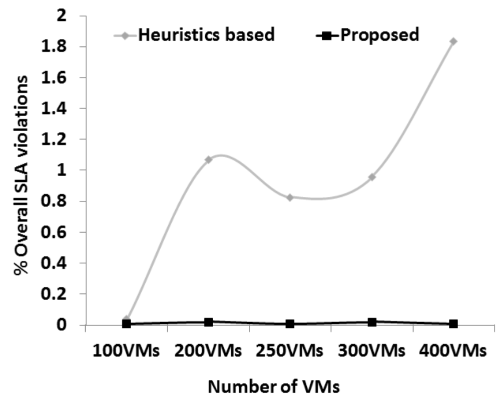

FIGURE 8. Overall SLA violations results of experiment 1.

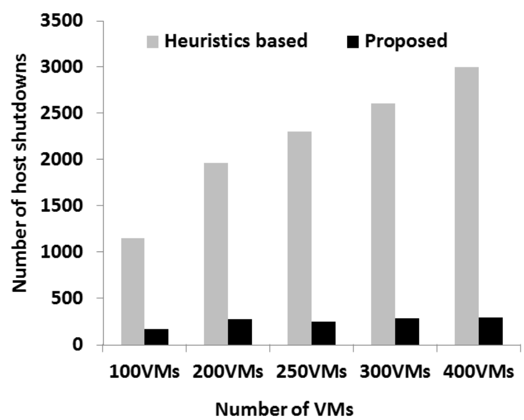

FIGURE 9. Number of host shutdowns during experiment 1.

experiment 1 are much smaller in case of our proposed solution when compared with a heuristic based solution for all the configurations.

The mean time for VM placement and host selection process are smaller in case of the proposed solution than heuristics based method, as shown in figure 10 and figure 11.

D. EXPERIMENT 2: PLANETLAB WORKLOAD WITH MULTIPLE HOST MACHINE(PM) TYPES

The goal of experiment 2 is to evaluate the effectiveness of our solution using a real-world workload in a data center.

TABLE 6. Evaluation results for performance metrics for experiment 1.

Number of VMs	Energy consumption (in KWh)		Number of VM migrations		SLA perf degradation due to migration		Overall SLA violation		Number of host shutdowns	
	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution
100	22.51	20.69	6102	412	0.15%	0.01%	0.04%	0.01%	1155	175
200	43.96	41.57	11156	921	0.13%	0.01%	1.07%	0.02%	1969	277
250	54.79	52.24	13001	866	0.12%	0.01%	0.83%	0.01%	2304	249
300	64.72	63.14	15107	1070	0.12%	0.01%	0.96%	0.02%	2607	289
400	89.55	82.50	22068	1100	0.14%	0.01%	1.84%	0.01%	3003	292

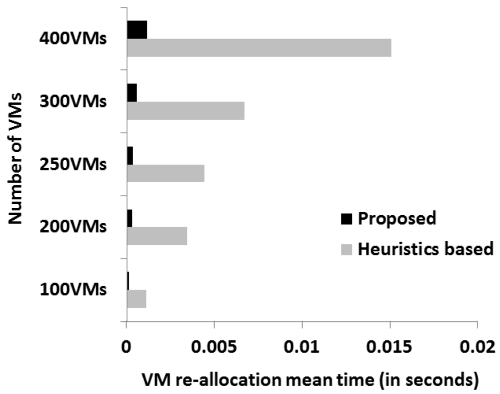

FIGURE 10. VM re-allocation mean time in experiment 1.

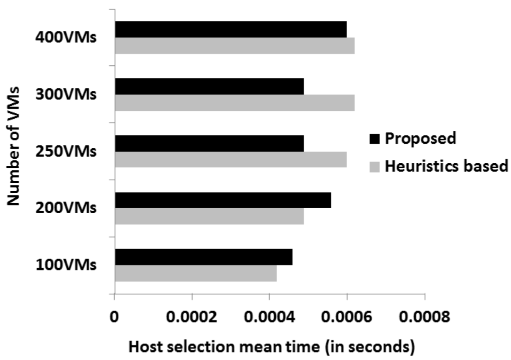

FIGURE 11. Host selection mean time in experiment 1.

Three different configurations have been chosen for testifying different host machine compositions with varying power and performance characteristics.

- *Configuration 2.1: DC with 2 host types of 400 PMs*
- *Configuration 2.2: DC with 4 host types of 400 PMs*
- *Configuration 2.3: DC with 6 host types of 400 PMs*

The simulation of the cloud data center is done using six types of host machines with configurations shown in Table 3. The experiment uses a real workload consisting of 1033 VMs resource utilization data [15] captured in PlanetLab servers. The physical machine(PM) types used in all 3 configurations is listed in Table 7, and their configurations are specified in Table 4.

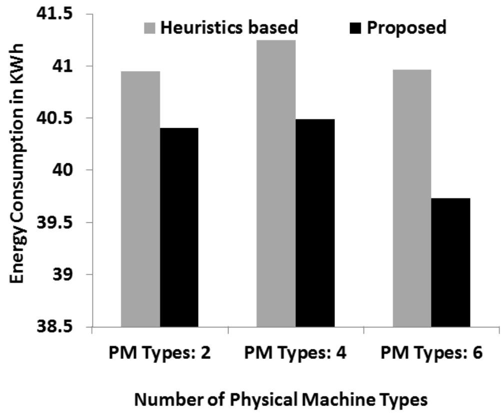

FIGURE 12. Energy consumption results of experiment 2.

TABLE 7. Physical machines types used in experiment 2.

Configuration name	Host(PM) types
Configuration 2.1	HP ProLiant ML110 G4 IBM server x3250
Configuration 2.2	HP ProLiant ML110 G4 IBM server x3250 HP ProLiant ML110 G5 IBM server x3550 [XeonX5675]
Configuration 2.3	HP ProLiant ML110 G4 IBM server x3250 HP ProLiant ML110 G5 IBM server x3550 [XeonX5675] HP ProLiant ML110 G3 IBM server x3550 [Xeon X5670]

Figure 12 shows the energy consumption profile of heuristic-based solution and our proposed solution for 3 configurations of host types with the real workload. Results indicate that our proposed solution saves approximately 1-3% energy in comparison with heuristics based method. The results of experiment 2 are tabulated in Table 8.

The graphs plotted for total VM migrations in figure 13 shows that the VM migrations are almost the same for all configurations and much smaller in number compared to heuristics based method. Figure 14 and figure 15 shows that overall SLA violations observed and total host shutdowns

TABLE 8. Evaluation results of performance metrics for experiment 2.

Types of Physical Servers	Energy consumption (in KWh)		Number of VM migrations		SLA perf degradation due to migration		Overall SLA violation		Number of host shutdowns	
	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution	Heuristics based	Proposed Solution
2	40.95	40.41	16102	1875	0.07%	0.00%	0.08%	0.00%	2211	377
4	41.25	40.49	15855	1828	0.07%	0.00%	0.08%	0.00%	2228	382
6	40.97	39.73	15941	1716	0.07%	0.00%	0.08%	0.00%	2217	378

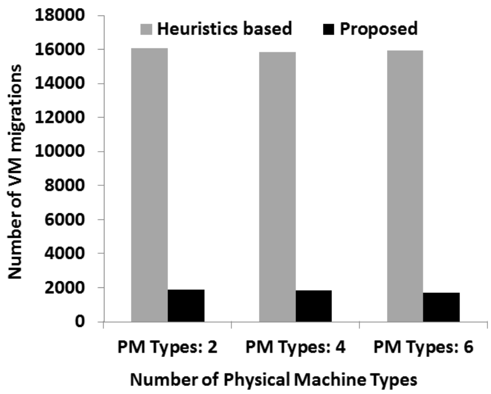

FIGURE 13. VM migrations results of experiment 2.

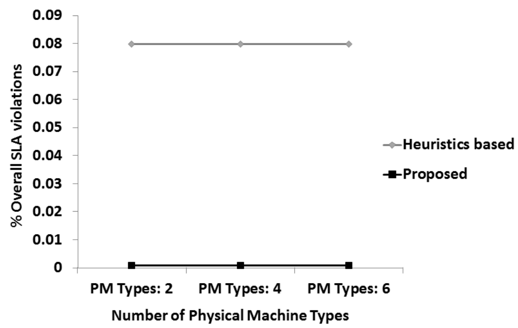

FIGURE 14. Overall SLA violations for experiment 2.

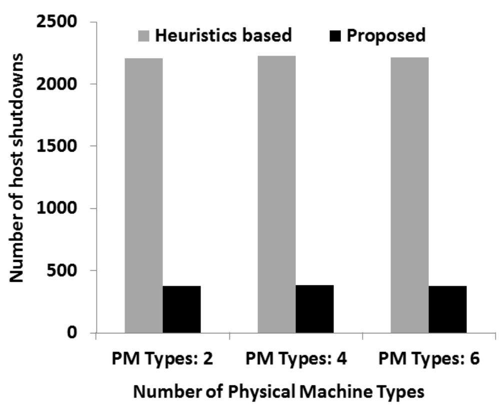

FIGURE 15. Number of host shutdowns reported during experiment 2.

during experiment 2 also are much smaller in case of our proposed solution when compared with a heuristic based solution for all the configurations. It can be observed from

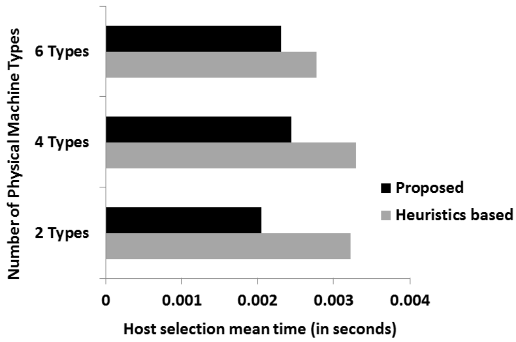

FIGURE 16. Mean host selection time in experiment 2.

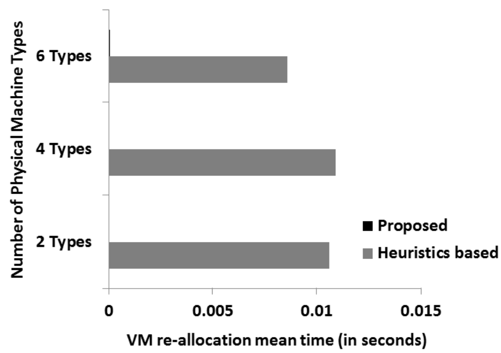

FIGURE 17. Mean VM re-allocation time in experiment 2.

figure 16 and 17 that mean time for VM placement and host selection process are smaller in case of the proposed solution than heuristics based method in case of experiment 2 also.

VII. CONCLUSIONS

The paper presents a context adaptive self-managing VM load balancing scheme for virtualized cloud data centers for dynamically assigning VMs to available PMs based on the data center load context and physical machine characteristics (power and performance). The main objective of the proposed scheme is to reduce power consumption and improve the efficiency of data center operations by minimizing SLA Violations. Experimental evaluations were conducted for comparing the proposed context adaptive solution against an existing heuristics based approach using both synthetic workload and real workload traces for various combination of VM and PM types.

Performance evaluation results showed that context adaptive solution performs better than heuristics based technique for power consumption minimization and improves the efficiency of the operation by reducing VM migrations, host shutdowns and SLA violations in both the experiments conducted. The key differentiating factors between proposed context adaptive solution and heuristics based technique are using performance and power characteristics of physical machines and using the global context of the data center to improve the decision making. Further research is planned to improve the context adaptive scheme by incorporating the resource consumption characteristics of VMs residing on the same PM to help reduce SLA violations and performance bottlenecks caused by conflicting resource demands by VMs.

REFERENCES

[1] Y. Guo and Y. Fang, "Electricity cost saving strategy in data centers by using energy storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1149–1160, Jun. 2013. doi: 10.1109/TPDS.2012.201.

[2] A. Rahman, X. Liu, and F. Kong, "A survey on geographic load balancing based data center power management in the smart grid environment," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 214–233, 1st Quart., 2014. doi: 10.1109/SURV.2013.070813.00183.

[3] A. Mosa and N. W. Paton, "Optimizing virtual machine placement for energy and SLA in clouds using utility functions," *J. Cloud Comput., Adv., Syst. Appl.*, vol. 5, p. 17, Oct. 2016. doi: 10.1186/s13677-016-0067-7.

[4] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurrency Comput., Pract. Exper.*, Vol. 24, no. 13, Sep. 2012. doi: 10.1002/cpe.1867.

[5] W. Li, J. Tordsson, and E. Elmroth, "Virtual machine placement for predictable and time-constrained peak loads," in *Economics of Grids, Clouds, Systems, and Services*. Berlin, Germany: Springer-Verlag, 2011, pp. 120–134. doi: 10.1007/978-3-642-28675-9_9.

[6] Y.-J. Chiang, Y.-C. Ouyang, and C.-H. Hsu, "An efficient green control algorithm in cloud computing for cost optimization," *IEEE Trans. Cloud Comput.*, vol. 3, no. 2, pp. 145–155, Apr./Jun. 2015. doi: 10.1109/TCC.2014.2350492.

[7] A. N. Toosi, K. Vanmechelen, K. Ramamohanarao and R. Buyya, "Revenue maximization with optimal capacity control in infrastructure as a service cloud markets," *IEEE Trans. Cloud Comput.*, vol. 3, no. 3, pp. 261–274, Jul./Sep. 2015. doi: 10.1109/TCC.2014.2382119.

[8] A. K. Kulkami and B. Annappa, "Cost aware service broker algorithm for load balancing geo-distrubuted data centers in cloud," in *Proc. IEEE Int. Conf. Signal Process., Inform., Commun. Energy Syst. (SPICES)*, Aug. 2017, pp. 1–5. doi: 10.1109/SPICES.2017.8091337.

[9] X. Fan, W.-D. Weber, and L. A. Barroso, "Power provisioning for a warehouse-sized computer," in *Proc. 34th Annu. Int. Symp. Comput. Archit.*, New York, NY, USA, Jun. 2007, pp. 13–23.

[10] D. Kusic, J. O. Kephart, J. E. Hanson, N. Kandasamy, and G. Jiang, "Power and performance management of virtualized computing environments via lookahead control," *Cluster Comput.*, vol. 12, no. 1, pp. 1–15, Mar. 2009. doi: 10.1007/s10586-008-0070-y.

[11] *The SPECpower Benchmark*. Accessed: Mar. 10, 2019. [Online]. Available: http://www.spec.org/power_ssj2008/

[12] K. Park and V. S. Pai, "CoMon: A mostly-scalable monitoring system for PlanetLab," *ACM SIGOPS Operating Syst. Rev.*, vol. 40, pp. 65–74, Jan. 2006.

[13] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Softw., Pract. Exper.*, vol. 41, no. 1, pp. 23–50, Jan. 2011. doi: 10.1002/spe.995.

[14] J. Hamilton. *Overall Data Center Costs*. Accessed: Mar. 10, 2019. [Online]. Available: https://perspectives.mvdirona.com/2010/09/overall-data-center-costs/

[15] *PlanetLab-Workload-Traces*. Accessed: Mar. 10, 2019. [Online]. Available: https://github.com/beloglazov/planetlab-workload-traces

[16] I. S. Moreno, R. Yang, J. Xu, and T. Wo, "Improved energy-efficiency in cloud datacenters with interference-aware virtual machine placement," in *Proc. IEEE 11th Int. Symp. Auton. Decentralized Syst. (ISADS)*, Mexico City, Mexico, Mar. 2013, pp. 1–8. doi: 10.1109/ISADS.2013.6513411.

[17] X. Li, Z. Qian, S. Lu, and J. Wu, "Energy efficient virtual machine placement algorithm with balanced and improved resource utilization in a data center," *Math. Comput. Model.*, vol. 58, nos. 5–6, pp. 1222–1235, Sep. 2013. doi: 10.1016/j.mcm.2013.02.003.

[18] S. E. Dashti and A. M. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," *J. Exp. Theor. Artif. Intell.*, vol. 28, nos. 1–2, pp. 97–112, 2016. doi: 10.1080/0952813X.2015.1020519.

[19] P. N. Sayeedkhan and S. B. S, "Virtual machine placement based on disk I/O load in cloud," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 4, pp. 5477–5479, 2014. doi: 10.1.1.432.5281.

[20] Z. Xiao, J. Jiang, Y. Zhu, Z. Ming, S. Zhong, and S. Cai, "A solution of dynamic VMs placement problem for energy consumption optimization based on evolutionary game theory," *J. Syst. Softw.*, vol. 101, pp. 260–272, Mar. 2015. doi: 10.1016/j.jss.2014.12.030.

[21] J.-K. Dong, H.-B. Wang, Y.-Y. Li, and S.-D. Cheng, "Virtual machine placement optimizing to improve network performance in cloud data centers," *J. China Univ. Posts Telecommun.*, vol. 21, no. 3, pp. 62–70, 2014. doi: 10.1016/S1005-8885(14)60302-2.

[22] R. Kanagavelu, B.-S. Lee, N. T. D. Le, L. N. Mingjie, and K. M. M. Aung, "Virtual machine placement with two-path traffic routing for reduced congestion in data center networks," *Comput. Commun.*, vol. 53, pp. 1–12, Nov. 2014. doi: 10.1016/j.comcom.2014.07.009.

[23] C. Liu, C. Shen, S. Li, and S. Wang, "A new evolutionary multi-objective algorithm to virtual machine placement in virtualized data center," in *Proc. IEEE 5th Int. Conf. Softw. Eng. Service Sci.*, Beijing, China, Jun. 2014, pp. 272–275. doi: 10.1109/ICSESS.2014.6933561.

ASHWIN KUMAR KULKARNI received the B.E. degree in computer science and engineering from the KBN College of Engineering, Gulbarga, India, and the M.Tech. degree from the Basaveshwar Engineering College, Bagalkot, India. He is currently a Ph.D. Research Scholar with the Computer Science and Engineering Department, National Institute of Technology Karnataka, Surathkal, India. He has published four research papers in international conferences/journals so far. His current research interests include cloud computing, parallel programming, and distributed computing.

B. ANNAPPA received the B.E. degree from the Govt. B.D.T. College of Engineering, Davangere affiliated Mysore University, Karnataka, and the M.Tech. and Ph.D. degrees in computer science and engineering from the National Institute of Technology Karnataka, Surathkal. He is currently a Professor with the Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, India. He has more than 25 years of experience in teaching and research. He has published more than 100 research papers in international conferences and journals. His research interests include cloud computing, big data analytics, distributed computing, software engineering, and process mining. He is a Fellow of the Institution of Engineers (INDIA). He is also a Life Member of the Computer Society of India, the Indian Society of Technical Education, the Cloud Computing Innovation Council of India and the Advanced Computing and Communications Society. He was the Organizing Chair of the International Conference ADCONS-2013. He was the Chair of IEEE Mangalore Subsection, in 2018. He is also the General Chair of DISCOVER 2010. He serves on the TPC of many international conferences and reviewer for several journals. He is also the Chair of the IEEE Computer Society Chapter, India Council. He was the Secretary of the IEI Mangaluru Local Center.