Toward an Enhancead Dynamic VM Consolidation Approach for Cloud Datacenters Using Continuous Time Markov Chain

Monireh Hosseini Sayadnavard
Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran
mo.sayadnavard@gmail.com
Abolfazl Toroghi Haghighat*
Department of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran
haghighat@qiau.ac.ir
Amir Masoud Rahmani
Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran
rahmani@srbiau.ac.ir

Received: 08/Jan/2019 Revised: 18/Mar/2019 Accepted: 23/Apr/2019

Abstract

Dynamic Virtual Machine (VM) consolidation is an effective manner to reduce energy consumption and balance the resource load of physical machines (PMs) in cloud data centers that guarantees efficient power consumption while maintaining quality of service requirements. Reducing the number of active PMs using VM live migration leads to prevent inefficient usage of resources. However, high frequency of VM consolidation has the negative effect on the system reliability and we need to deal with the trade-off between energy consumption and system reliability. In recent years many research work has been done to optimize energy management using power management techniques. Although these methods are very efficient from the point of view of energy management, but they ignore the negative impact on the system reliability. In this paper a novel approach is proposed to achieve a reliable VM consolidation method. In this way, a Markov chain model is designed to determine the reliability of PMs and then it has been prioritized PMs based on their CPU utilization level and reliability status. Two algorithms are presented to determining source and destination servers. The efficiency of our proposed approach is validated by conducting extensive simulations. The results of the evaluation clearly show that the proposed approach significantly improve energy consumption while avoiding the inefficient VM migrations.

Keywords: Cloud Computing; VM Consolidation; Energy Efficiency; Reliability; Markov Chain.

1. Introduction

Cloud computing as one of the most interesting developments in technology is an on-demand computing model that provides services to users through Internet [1]. The ever increasing demand for computing resources has resulted in establishment of large-scale data centers, which require enormous amount of power and hence consume a lot of energy. Statistics of the worldwide data center electricity consumption show non-linear growth during the last decade and a similar trend is expected for the upcoming years [2]. The amount of computing resources and the inefficient use of these resources could lead to huge energy wastage. An effective way to improve the resource utilization and energy efficiency in cloud data centers is VM consolidation that has been widely studied in recent years [3,4]. During consolidation, VMs are periodically reallocated using live migration according to their current resource demand to minimize the number of active physical servers and the idle PMs are switched to low power modes to reduce the energy consumption [5]. Since most modern applications experience dynamic patterns of resource consumption because of highly variable workloads, VM consolidation in clouds is a complicated operation. Unconstrained VM consolidation may lead to degraded performance when an application is faced with increasing demand and resource usage. If the resources requirements of an application are not met, the response time will increase. While the QOS guarantee defined in the Service Level Agreements (SLAs) between the Cloud provider and their users is essential. Hence, the Cloud providers must consider the trade-off between performance and energy consumption in order to fulfil QOS requirements [6].

On the other hand, high VM consolidation has the negative effect on the reliability of the system [4,7,8]. most existing research on consolidation has focused on the performance-energy trade-off. While there are some works that consider the relationship of system reliability and energy efficiency in Cloud environment [8,9] and still there is a distinct need for more research on the mentioned challenge.

Server consolidation may increase the probability of server failure and compromise the reliability of the system by increasing the load on some servers and shutting down some of them. Hence, we need to consolidate servers in flexible manner with considering both energy efficiency and reliability to cover different operating conditions and scenarios.

In this paper we present a novel approach to dynamic VM consolidation by considering both reliability and

* Corresponding Author

energy efficiency simultaneously. We try to manage energy consumption along with considering the reliability of each PM in every phase of consolidation to reach equilibrium between these two metrics. we have calculated the reliability under the probability of failures occurrence in a heterogeneous environment. A computational model for PMs reliability prediction is presented and based on the results of this phase and the CPU utilization level of each PM, they are divided into different categories. Then, we can make proper decision to VMs migration to realization of our purpose. Specifically, the major contributions of this paper can be summarized as below:

- Designing a Markov chain model for predicting and analysing PMs reliability for VM consolidation implementation to optimize the relationship between energy efficiency and reliability.
- Propose a new policy to target PMs selection in dynamic VMs consolidation process to improve energy efficiency while considering the reliability factor of PMs.

The remainder of the paper is organized as follows: The current and past research on VM consolidation are reviewed in Section 2. The system model and problem statement are explained in Section 3. Our Markov Chain based reliability model and the proposed approach for dynamic VM consolidation are described in detail in Section 4. The experimental setup and results are shown and discussed in Section 5. Finally, conclusions are presented in the last section.

2. Related Works

There are several research works that addresses the VM consolidation. In this section we review relevant approaches in the literature related to the similar issues.

Many studies formulated the VM consolidation as a well-known NP-hard bin packing problem [6, 10-13]. Various heuristics like greedy algorithms are utilized to approximate the optimal solution of this NP-hard problem. These include worst fit and best fit in [10], first fit decreasing(FFD) and best fit decreasing (BFD) [11]. The authors in [12] have divided VMs consolidation into the four following phases: host overload detection, selection of VMs should be migrated, VM placement, and running PMs shrinking. Due to the complexity of VMs consolidation, the VMs consolidation issues in [12] were separated into several sub-problems, and then they have proposed novel adaptive heuristics for each sub-problem. They proposed MBFD algorithm to VM placement by considering power consumption and SLA violation. In this algorithm, the VMs are first sorted in decreasing order based on their utilizations. Then, these VMs are allocated to the hosts having minimum increase of energy consumption.

In [13], a VM consolidation framework is proposed to minimize the performance-energy trade-off. The VM placement problem is resolved using semi-online multidimensional bin packing. The authors in [14] have considered rack, cooling structure and network topology when consolidating VMs. In this paper the MBFD algorithm is improved and then three structure-aware VM placement methods are proposed to consolidate VMs in the servers to minimize the number of active racks that results in turning off idle routing and cooling equipment in order to reduce the energy consumption. In [15] a burstiness-aware server consolidation algorithm, QUEUE, is proposed. First, the burstiness of workload is captured using a two-state Markov chain, then some extra resources on each PM is reserved to avoid live migrations. Shen et al. in [16] proposed a mechanism that predicts the VM resource utilization patterns and consolidates complementary VMs with spatial/ temporal awareness into one PM to reduce the number of PMs, maximize resource utilization and reduce the number of VM migrations. Complementary VMs are the VMs whose total demand of each resource dimension in the spatial space nearly reaches their host PM’s capacity during VM lifetime period in the temporal space.

In [17] DVFS-aware consolidation procedure is presented to eliminate the inconsistencies between consolidation and DVFS techniques. this paper also has proposed PSFWT as a fuzzy DVFS-aware multi criteria and objective resource allocation solution for VM placement in Cloud data centers that simultaneously optimizes important objectives including energy consumption, SLA violation, and number of VM migrations. different criteria of the system including CPU, RAM, and network bandwidth in decision making process is considered.

Beloglazov and Buyya [5] investigated the problem of overloaded hosts detection using a Markov chain model. A specified QoS goal is defined to maximizing the mean time between VM migrations for any known stationary workload. The unknown nonstationary workloads are also handled using a multi size Sliding Window workload estimation. In [18] a heuristics based multi-phase approach for server consolidation is proposed which effectively reduces residual resource fragmentation along with reducing the number of active PMs. Residual Resource Fragmentation refers to the state where sufficient amount of residual resources is available but are fragmented and distributed across multiple active PMs. In [19] a VMs placement algorithm is proposed that considers computation resources, Quality of Service (QoS) metrics, virtual machine status and I/O data with priority based probability queuing model. Data location during Virtual machines placement is considered to avoid unnecessary migration to gain high performance for applications. The authors in [20] studied the influence of four aspects on energy consumption and QoS, namely, the dynamic workload, CPU utilization, times of VM migrations, and opportunity of VM migration from nine related factors. They created a Bayesian Network based estimation model (BNEM) for dynamic VM migration using these factors that each node represents one aspect of VM migration. Khani et al. [21] proposed a distributed mechanism for dynamic consolidation of virtual machines using a non-cooperative game for reducing power consumption in data centers with heterogeneous PMs. In [22] a prediction-based consolidation approach is proposed that considers both an estimation of future requested resources using Kernel Density Estimation technique, and future migration traffic to decrease the number of migrations.

There are also various metaheuristic algorithms that have been proposed to solve the VM consolidation problem in cloud computing environments. These algorithms rely on a probabilistic approach to find near optimal solutions to the problems. In [23], ant colony optimization method (ACO) is used to pack the VMs into the least number of physical machines while preserving Quality of Service requirements. A multi-objective function is defined that considers both the number of dormant PMs and the number of migrations. The GABA approach [24] is a genetic algorithm (GA) based algorithm that dynamically finds the optimum reconfiguration for a set of VMs according to the predicted future demand of the running workload. The algorithm decreases the number of PM significantly and converges within reasonable time. In [25], a VM consolidation approach is proposed based on the particle swarm optimization (PSO) algorithm, which considered reducing energy consumption and improving resource utilization in the data center as the optimization objective. In [26], a nonlinear model is introduced to quantify PM power consumption and then VM placement is formulated as a bi-objective optimization problem, which is solved using an ACO based algorithm.

Deng et al. in [8] presented a Reliability-Aware server Consolidation stratEgy (RACE) to address a multi-objective problem with considering hardware reliability and energy efficiency. A utility model has been formulated that uses three parameters U_{SLA}, U_r, and U_e to determine the best VM-to-PM mapping. U_{SLA} ensures that there are enough resources to support the SLA, U_r value shows the impacts of turning servers on and off and temperature variation on reliability and lifetime, and U_e estimates the amount of power usagereduction. Finally, the mapping that has the maximum value of the sum of these three parameters is chosen to provide an optimized solution of the problem.

There are also some works that considered energy efficiency and reliability in cloud computing at the same time that a review of them has been provided in [7]. But most of these works have not specifically addressed the issue of consolidation and focus on resource allocation in a reliable and energy efficient manner. However, our proposed approach provides a reliability model of PMs to use in consolidation process with the aim of saving unnecessary wastage of energy that will be required to restart all the running process that were interrupted during the failure.

3. System Model And Problem Definition

We consider a system consist of a single data center with heterogeneous resources as the scope of our work is restricted to migrations within a data center. Let PM = $\{pm_1, pm_2, \ldots, pm_i, \ldots, pm_m\}$ be the set of active PMs in the current state of the data center and $VM_i=\{vm_1, vm_2, \ldots, vm_j, \ldots, vm_n\}$ be the set of deployed VMs that in PM_i. Each PM is characterized by the CPU performance defined in Millions of Instructions Per Second (MIPS), amount of RAM, network bandwidth, and disk storage. But the disk storage space in any PM is usually large and dynamic variations in disk space requirements are usually not observed. Hence, it can be safely neglected. The other three resources CPU, Memory and Network Bandwidth are considered for the consolidation process.

At any given time, a cloud data center usually serves many simultaneous users. Users submit their requests for provisioning *n* heterogeneous VMs, which are allocated to the PMs and characterized by requirements of resources. The length of each request is specified in millions of instructions (MI). It is assumed that each of the *n* VMs is already placed in some PM in the data center. The problem is to minimize the number of PMs used, by maximizing the resource utilization in each PM using live migration of VMs so that the freed PMs can be set to a power saving state.

3.1 Reliability Model

Reliability is defined as an evaluation parameter to measure the system's ability to functioning correctly under certain conditions over a specified interval of time. In cloud computing system there are two general aspects of reliability in server consolidation approaches [4], service reliability, and hardware reliability. In this study, the second category is considered. We formulated the reliability of PMs by using the reliability of two important components of a PM, hardware, and hypervisor (VMM) that is explained in the next section and it can be expressed as [27]:

$$R_{PM_i}(t) = R_{HW_i}(t)R_{VMM_i}(t) = e^{-\left(\lambda_{HW_i}+\lambda_{VMM_i}\right)t} \quad (1)$$

4. Proposed Approach

The proposed approach to consists of the following two main components.

- Prediction module: observes energy consumption caused by VMs and PMs and collect historical data of past failures that can be utilized in Markov chain based prediction model. The module is executed on each PM locally.
- Decision making unit: manages VM placement on PMs in the data center. According to the received PMs messages and states analysis, this unit determines each PM belongs to which category. Then, VM selection and target PM selection algorithms are carried out and appropriate decisions are made to solve the consolidation problem.

4.1 Markov Chain Based Prediction Model

Markov chain model is the most fundamental and general state-based stochastic method that concerns about a sequence of random variables, which correspond to the states of a system, in such a way that the state at one time epoch depends only on the one in the previous time epoch [28].

Markov chains are usually classified into two categories: Discrete Time Markov Chains (DTMC) and Continuous Time Markov Chains (CTMC). CTMC, semi-Markov process and Stochastic Petri Net (SPN) have been used widely for evaluating the performance [29], reliability/ availability [30], and performability [31] of computer systems. in this paper, we choose CTMC model to develop a prediction mechanism to analysis PMs reliability. Since, exponential random variable is the only continues random variable with Markov property and hardware and software fault are commonly modelled as exponential distribution, we assume that the time to transit from a system state to another due to failures and recovery follows an exponential distribution. Fig.1 shows the CTMC model state transition diagram for the probabilistic reliability behavior of each PM in data centre. Although in many works only two active and failed states are considered for a host, but there are some factors that result in performance degradation.

In this study we consider that hypervisor (VMM) is affected by software aging. One of the common ways to deal with this problem is software rejuvenation as a proactive fault management technique to prevent or postpone failures in VMMs and VMs. Migrate-VM

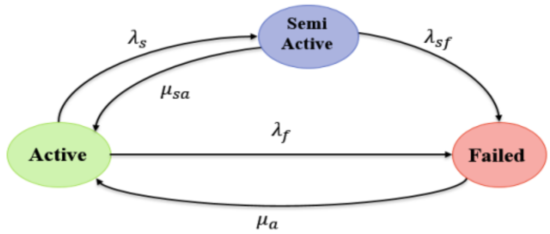

Fig. 1. State transition diagram

rejuvenation [32] is an effective technique for VMM rejuvenation. In this technique, before triggering the VMM rejuvenation, running VMs are migrated to another host and then VMM rejuvenation starts. If we choose a PM with aged VMM as a VMs migration destination in consolidation process, leads to increase the number of migrations and waste energy. Therefore, in order to model the reliability of PMs, in addition to hardware failures, VMM failures are also considered which can be extended to other software failures.

As depicted in fig.1, the model consists of three states including Active, Semi active and Failed. Let X(t) with discrete state space S = {A, SA, F} represents the state of PM at time t.

$$X(t) = \{X_A(t), X_{SA}(t), X_F(t)\} \qquad (2)$$

if X(t)= $X_A(t)$, PM is in proper condition and active state. In other two states that are inactive states we consider hypervisor failure and hardware failures. Hardware failures are critical and lead PMs to the failed state. Semi active state is about VMM rejuvenation process during which the PM is not available.

We define μ and λ as the repair rate and the failure rate of a PM, respectively. With this assumptions, the transient process X(t) can be modelled mathematically as a homogeneous CTMC on the state space S. for each time t>0, the probability of a PM in state i is given by $X_i(t)=\Pr\{X(t) = i\}$, $i \in S$. The Markov process is defined by generator Q whose is given by:

$$Q = \begin{bmatrix} -\lambda_s - \lambda_f & \lambda_s & \lambda_f \\ \mu_{sa} & -\mu_{sa} - \lambda_{sf} & \lambda_{sf} \\ \mu_a & 0 & -\mu_a \end{bmatrix} \qquad (3)$$

First Passage Time: Let τ_j be the expected value of random time to reach state j (for the first time), given that it started in state i. These are sometimes referred to as mean first-passage time. The first passage time into state N is defined to be

$$T = min\{n \geq 0 : X(t) = N\} \qquad (4)$$

Where {1, …, N} represent the state space. the expected value E(T) is defined as

$$\tau_j = E\big((T|X_0 = i)\big) \qquad (5)$$

According to a theorem defined in [33], the expected first passage time satisfy the following relation,

$$r_i\tau_i = 1 + \sum_{j=1}^{N-1} r_{i,j}\tau_j \quad . \ 1 \leq i \leq N-1 \qquad 1 \qquad (6)$$

Where $r_i = \sum_{j=1}^{N} r_{i,j}$ and $r_{i,j}$ is the entry of rate matrix R,

$$R = \begin{bmatrix} 0 & \lambda_s & \lambda_f \\ \mu_{sa} & 0 & \lambda_{sf} \\ \mu_a & 0 & 0 \end{bmatrix} \qquad (7)$$

historical data and past failures can be utilized to estimate the λ and μ. Then, the generator matrix is constructed based on the estimated rates and the CTMC transition diagram. In the next step, transient state analysis performs to predict the state of PM and also compute the defined metrics values including expected time for the first occurrence of failure. The difference between the predicted and actual values can be used to train and modify the transition rates. Then, the obtained results are sent to decision making unit and used to classify the PMs for consolidation process.

4.2 VM Consolidation Process

After performing the prediction phase, PMs status is sent to the decision making unit. To determine whether the host is overloaded, we apply LR method proposed by beloglazov et al. [34]. this method utilizes local regression to fit a trend polynomial to the last k observations of the CPU utilization. In LR method for each new observation a new trend line is found. This trend line is used to estimate the next observation. Then the algorithm decides that the host is considered overloaded and some VMs should be migrated from it. Underloaded PMs can be found by comparing the CPU utilization with a low threshold. Other PMs are considered as well-utilized. According to the obtained results from previous steps, each PM will be in one of the six sets WR, OR, UR, WU, OU and UU. These sets represent the well-utilized and reliable, overloaded and reliable, under-loaded and reliable, well-utilized and unreliable, overloaded and unreliable, and under-loaded and unreliable PMs, respectively. Unreliable state is related to semi active and fail states in Markov model. Then these sets are divided into critical, optimal and sub optimal categories.

To select the migration source, the categories whose PMs are in critical situation are candidate. PMs in OU set have the highest priority. PMs of critical, optimal, and sub optimal categories are sorted based on MFPT in ascending order. The pseudocode of the PMs categorization algorithm is given as algorithm 1. At the end of this phase, the potential source PMs are determined. It should be noted that if all the PMs are in WR set, no migration will be done.

4.2.1 VM Selection and Placement

When finding a set of PMs in critical category, some VMs in the hosts are migrated to guarantee QoS for the users. Therefore, a VM selection policy is needed in the dynamic VM consolidation. Here, VMs are selected based on the Minimum Migration Time (MMT) policy [34]. MMT migrates a VM *v* that requires the minimum time to complete a migration relatively to the other VMs allocated to the PM. The migration time is estimated as the amount of RAM utilized by the VM divided by the spare network bandwidth available for the PM *j*.

After the VMs to be migrated are acquired, we need a policy to select appropriate target for migrations. When a PM is selected as the destination of VMs migration, its state likely change due to increasing workload and resource usage. Therefore, the proposed algorithm in this phase tries to find a proper host with sufficient residual capacity and considering energy consumption and reliability. In this way PMs in WR list of the optimal category is explored at first.

If the algorithm fails to find adequate PM, the search process continues in underutilized and reliable PM within the sub optimal category. In our proposed policy, a VM will be migrated to a PM with the highest score that is estimated according to Eq. (8). To specify each PM score, energy cost and reliability are considered. Then, scores are determined using weight assignment to each criterion. α is an adjustable weight to obtain different trade-off points, since each cloud provider will pursue various objectives and business requirements.

According to the proposed scoring method, VMs assigned to the PMs that their mean first passage time to an unreliable state is longer than the others to prevent additional migrations. The idea behind this is that such PMs will probably stay in reliable state for a longer period of time. Therefore, the migrated VMs can complete their works on the same server without any interruption or wasting time because of forced migration.

$$Score_i = \alpha \times \left(R_{PM_i} + MFP\_time_{PM_i}\right) + (1 - \alpha) \times \left(EC_i^{curr} - EC_i^{after}\right) \quad (8)$$

Where R_{pm} is the reliability of target server and $MFP\_time_{PM_i}$ is the expected value of random time to reach the unreliable state starting from the reliable state. EC_i^{after} and EC_i^{curr} are the energy cost after and before the VM_i placement, respectively. In order to allocate VMs to PMs, VM migration list will be sorted according to their CPU capacity requirements in decreasing order. Then the score of each PM in WR list is computed (see Algorithm 2).

After performing the aforementioned steps, we can safely shut down remaining underloaded PMs in UR and UU sets of suboptimal category. First we attempt to migrate all VMs on the PMs of UU set because of unreliability. If the proper destination PMs was found to hosting the migrated VMs, then source PM is switched to sleep mode. Finally, UR set is explored to reduce the number of active PMs as much as possible.

5. Performance Evaluation

This section describes our experimental results for the proposed approach. In this study we have chosen CloudSim toolkit [35] as the simulation platform that is a modern simulation framework for cloud computing environments. The experiments simulate a data center comprised of 800 heterogeneous PMs, half of which are HP ProLiant ML110 G4 (Intel Xeon 3040 2 Cores 1860 MHz, 4 GB) servers, and the other half are HP ProLiant ML110 G5 (Intel Xeon 3075 2 Cores 2260 MHz, 4 GB). VMs are supposed to correspond to Amazon EC2 instance types with the only exception that all the VMs are single-core, because of the fact that the workload data used for the simulations come from single-core VMs.

There are four types of VMs in the experiments: High-CPU Medium Instance, Extra Large Instance, Small Instance, and Micro Instance. After creating PM and VM instances on the CloudSim platform, the VMs are deployed to random PMs based on their resource requirements. After each round of VMs consolidation, VMs resource demands changes according to workload data. We assume HighTR and LowTR thresholds equal to 0.8 and 0.4, respectively. The parameter α is set to 0.5 in our experiments. To estimate the reliability of PMs, reliability of hardware is computed based on the decrease of mean time to failure(MTTF) models presented in [8] with considering CPU and disk reliability degradation.

In order to make the results of simulation more realistic, it is important to conduct experiments using workload traces from a real system. We have used data that provided as a part of the CoMon project, a monitoring infrastructure for Planet Lab [36]. In this project, the data on the CPU utilization is obtained every five minutes by more than a thousand virtual machines from servers located at more than 500 places around the world. We have chosen 10 different days from the workload traces gathered during March and April 2011, randomly.

In order to reasonably evaluate the efficiency of our proposed approach, we adopt several metrics that were presented by beloglazov et al. [34]. There are many metrics to measure the efficiency and superiority of various algorithms for VM consolidation problem. The main targets of VM consolidation in the cloud data center is to reduce energy consumption and SLA violations. So, we have chosen the related metrics to these objectives. One metric is the energy consumption consumed by the data center and the metrics used for quantifying SLA violations are based on the model provided in the CloudSim simulator (SLATH, PDM and SLAV).

The SLATAH is defined as Eq. (9), measures the percentage of time during which active hosts have experienced CPU utilization of 100%.

$$SLATAH = \frac{1}{n}\sum_{i=1}^{n}\frac{T_i^s}{T_i^a} \quad (9)$$

Algorithm 1: Categorization

```
Input: PMs states
Output: Categories
1:  foreach host in PM set do
2:     if PM_state = OU | OR | WU add to Critical_cat
3:        elseif PM_state = WR add to Optimal_cat
4:        elseif PM_state= UR | UU add to SubOptimal_cat
5:     end if
6:   end for
7:  Sort lists in Critical_cat based on mfpt in ascending order.
8:  Sort list in Optimal_cat based on mfpt in ascending order.
9:  Sort lists SubOptimal_cat based on mfpt in ascending order.
10: return categories
```

Algorithm 2: Target PM selection Algorithm

```
Input: MigrationList, WR_list, UR_list
Output: MigrationSchedule
1: Sort Migration_list by resource requirements in descending order
2: foreach VM_j in MigrationList do
3:   best_score = Min
4:   target_PM = Null
5:    foreach PM_i in WR_list do
6:       Calculate Score_i using eq. (8)
7:       if Score_i  >  best_score then
8:          best_score = Score_i
9:          target_PM = PM_i
10:     endif
11:   end for
12:  if target_PM= Null then
13:    foreach PM_i in UR_list do
13:       repeat steps 7-10
14:      calculate new PM_i_load
15:      if PM_i_load> LowTR then
16:          Add PM_i to WR_list
17:      endif
18:    endfor
19:   endif
20:   MigrationSchedule.put(VM_j, target_PM)
21:  end for
22: return  MigrationSchedule
```

Where n is the total number of physical machines, T_i^s is the total time of SLAV caused by the CPU resource overload of PM_i, T_i^a is the running time of PM_i. Another metric, PDM is calculated as follow:

$$PDM = \frac{1}{m}\sum_{j=1}^{m}\frac{C_j^d}{C_j^r} \quad (10)$$

Where m is the number of VMs, C_j^d is the unsatisfied CPU required capacity caused by the migration of VM_j, and C_j^r is the CPU capacity requested by VM_j. SLAV is a combined metric of two aforementioned metrics that evaluates a single-day QoS of the data center and is defined as:

$$SLAV = SLATAH \times PDM \quad (11)$$

Table 1. Simulation results of different algorithms

Method	EC(KWh)	SLAV	ESV(%)	VM Migrations
LR-MMT	160.21	0.49355	78.3394	32095
LR-MC	147.35	0.77112	111.2479	27350
LR-RS	146.01	0.78238	115.9127	26367
R-VMC	122.47	0.14171	19.6698	9457

ESV as described in EQ. (12), is a metric consist of energy consumption of a data center per day(EC) and the level of SLA violations. A lower estimation of ESV indicates that energy saving is higher than the SLA violations.

$$ESV = EC \times SLAV \quad (12)$$

5.1 Simulation Results

In this section the result of our experiments are discussed. Since we use LR method to host overload detection, three traditional combination methods, LR-MMT, LR-MC, and LR-RS [34], are selected to compare and evaluate our proposed approach. These methods apply PABFD algorithm [34] to target selection for migrated VMs. The safety parameter is set to 1.2 in experiments. The CTMC model parameter default values are found in the literature [27,32,37].

Comparison between other methods and our proposed algorithm (R-VMC) is shown in Table 1. The obtained results indicate that energy consumption is reduced by R-VMC algorithm compared to LR-MMT, LR-MC, and LR-RS, due to decreasing number of migrations and switching the underload and unreliable PMs to sleep mode which leads to energy saving. In terms of SLAV, R-VMC has optimal SLAV compared to others and LR-RS has the highest SLAV. According to the results, R-VMC's SLAV is only 18% of LR-RS's SLAV. These results reveal that R-VMC is better than other algorithms in guaranteeing QoS. The ESV index in Table 1 indicates that the comprehensive performance of R-VMC is considerably higher than others. The ESV of R-VMC is only 25% of LR-MMT which has the closest value to R-VMC, 17.6% of LR-MC and 16% of LR-RS. Eventually, the methods are compared in terms of the number of VM migration based on the experimental results. R-VMC has the lowest number of VM migrations because it avoids additional migration by selecting proper and reliable destination PMs. Fig. 2 shows the energy consumption of our proposed algorithm and the other algorithms. As can be seen, R-VMC is better than other algorithms in terms of energy consumption.

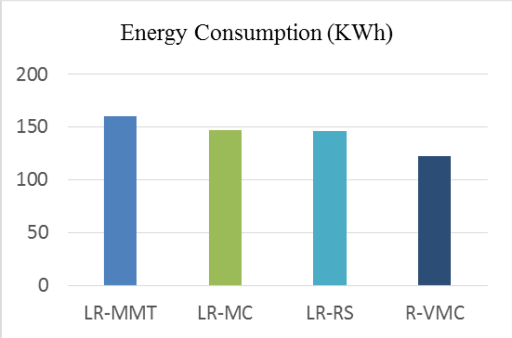

Fig. 2. The energy consumption of algorithms

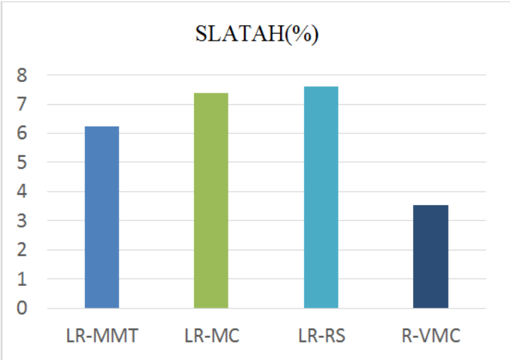

Fig. 3. Comparison of SLATAH

The reason is that R-VMC can avoid inefficient and extra migration due to selecting more reliable target PMs. So, minimizing the number of active physical servers and reducing the VM migrations leads to decreasing energy consumption.

Fig. 3 compares the results of SLATAH with other algorithms. It is completely obvious that R-VMC

outperforms the other methods and reduces PMs overload risk. The reason is that proposed approach can proactively migrate VMs from a host before the

Host become overloaded. On the other hand, R-VMC considers reliability which effectively leads to proper target PM selection. So, the QOS of running PMs is maintained.

Fig. 4 compares the R-VMC and the other algorithms in terms of PDM. As depicted in this figure, R-VMC has better performance. Prevention of extra migrations effects on this parameter directly and migrating VMs to the safer PMs with considering failures and VMM rejuvenation reduce the number of VM migration. Indeed, according to obtained results in experiments, we can conclude that one of our objectives about decreasing VM migration, has been achieved.

Fig. 5 shows the number of migrations of proposed algorithm and other algorithms. According to the results, the R-VMC has a smaller number of migrations and outperforms LR-MMT, LR-MC and LR-RS significantly.

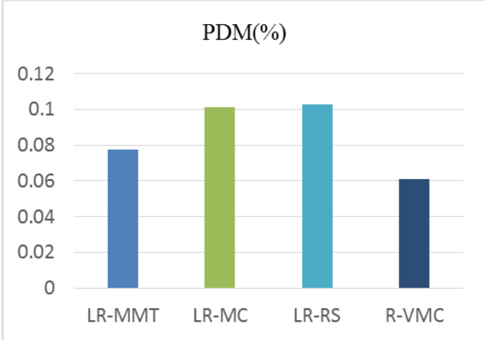

Fig. 4. Comparison of PDM

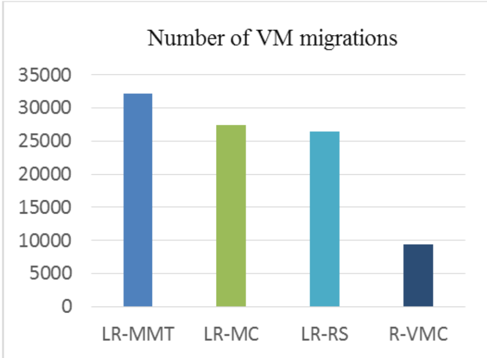

Fig. 5. Comparison of the number of VM migrations

The reason is that our proposed approach properly selects reliable servers as destination of migrations and prevents unnecessary migrations by avoiding unprofitable and aggressive reconfigurations. When we consider reliability, the probability of VMs migration, because of failure occurrence, is reduced. Therefore, while consuming a lower amount of energy, R-VMC has a fewer number of migrations.

6. Conclusion

In this paper, we have proposed a novel dynamic VM consolidation method in cloud data centers considering the reliability of each PM along with reducing the number of active PMs simultaneously. Most of the existing works on VM consolidation have been focused only on reducing the number of active PMs using VM live migration to prevent inefficient usage of resources. But on the other hand, high frequency of VM consolidation has the negative effect on the system reliability. Also, frequent turning on or off resources or putting them in sleep mode tends to make them more susceptible to failure and result in increasing the overall response time, service delays, and the SLA violation. Therefore, in this paper we tried to consolidate servers in flexible manner with considering both energy efficiency and reliability.

First, we have introduced a Markov model for reliability estimation of PMs. Then PMs are categorized based on the obtained results from the model and CPU overload detection algorithm (LR). Finally, we consider utilization along with the reliability in consolidation steps to select source and target PMs that leads to proper decision making and reduce migrations number, energy consumption, and consequently SLA violation. To evaluate the proposed VM consolidation method, CloudSim was chosen as the simulation platform and the simulation results have shown the effectiveness of R-VMC compared to other algorithms in terms of SLATAH, PDM, SLAV, EC, ESV and the number of VM migrations.

References

[1] M. Armbrust et al., "A view of cloud computing," Communications of the ACM, vol. 53, no. 4, pp. 50-58, 2010.

[2] M. Mills, "The cloud begins with coal. big data, big networks, big infrastructure, and big power. an overview of the electricity used by the digital ecosystem," ed, 2013.

[3] R. W. Ahmad, A. Gani, S. H. A. Hamid, M. Shiraz, A. Yousafzai, and F. Xia, "A survey on virtual machine migration and server consolidation frameworks for cloud data centers," Journal of Network and Computer Applications, vol. 52, pp. 11-25, 2015.

[4] A. Varasteh and M. Goudarzi, "Server consolidation techniques in virtualized data centers: A survey," IEEE Systems Journal, vol. 11, no. 2, pp. 772-783, 2017.

[5] A. Beloglazov and R. Buyya, "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, vol. 24, no. 7, pp. 1366-1379, 2013.

[6] A. Beloglazov, "Energy-efficient management of virtual machines in data centers for cloud computing," 2013.

[7] Y. Sharma, B. Javadi, W. Si, and D. Sun, "Reliability and energy efficiency in cloud computing systems: Survey and taxonomy," Journal of Network and Computer Applications, vol. 74, pp. 66-85, 2016.

[8] W. Deng, F. Liu, H. Jin, X. Liao, and H. Liu, "Reliability-aware server consolidation for balancing energy- lifetime tradeoff in virtualized cloud datacenters," International Journal of Communication Systems, vol. 27, no. 4, pp. 623-642, 2014.

[9] A. Varasteh, F. Tashtarian, and M. Goudarzi, "On Reliability-Aware Server Consolidation in Cloud Datacenters," arXiv preprint arXiv:1709.00411, 2017.

[10] L. Grit, D. Irwin, A. Yumerefendi, and J. Chase, "Virtual machine hosting for networked clusters: Building the foundations for autonomic orchestration," in Proceedings of the 2nd International Workshop on Virtualization Technology in Distributed Computing, 2006, p. 7: IEEE Computer Society.

[11] B. Speitkamp and M. Bichler, "A mathematical programming approach for server consolidation problems in virtualized data centers," IEEE Transactions on services computing, vol. 3, no. 4, pp. 266-278, 2010.

[12] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," Future generation computer systems, vol. 28, no. 5, pp. 755-768, 2012.

[13] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," in Proceedings of the 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing, 2010, pp. 826-831: IEEE Computer Society.

[14] S. Esfandiarpoor, A. Pahlavan, and M. Goudarzi, "Structure-aware online virtual machine consolidation for datacenter energy improvement in cloud computing," Computers & Electrical Engineering, vol. 42, pp. 74-89, 2015.

[15] S. Zhang, Z. Qian, Z. Luo, J. Wu, and S. Lu, "Burstiness-aware resource reservation for server consolidation in computing clouds," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 4, pp. 964-977, 2016.

[16] H. Shen and L. Chen, "Compvm: A complementary vm allocation mechanism for cloud systems," IEEE/ACM Transactions on Networking (TON), vol. 26, no. 3, pp. 1348-1361, 2018.

[17] E. Arianyan, H. Taheri, and V. Khoshdel, "Novel fuzzy multi objective DVFS-aware consolidation heuristics for energy and SLA efficient resource management in cloud data centers," Journal of Network and Computer Applications, vol. 78, pp. 43-61, 2017.

[18] K. S. Rao and P. S. Thilagam, "Heuristics based server consolidation with residual resource defragmentation in cloud data centers," Future Generation Computer Systems, vol. 50, pp. 87-98, 2015.

[19] A. Ponraj, "Optimistic virtual machine placement in cloud data centers using queuing approach," Future Generation Computer Systems, vol. 93, pp. 338-344, 2019.

[20] Z. Li, C. Yan, X. Yu, and N. Yu, "Bayesian network-based virtual machines consolidation method," Future Generation Computer Systems, vol. 69, pp. 75-87, 2017.

[21] H. Khani, A. Latifi, N. Yazdani, and S. Mohammadi, "Distributed consolidation of virtual machines for power efficiency in heterogeneous cloud data centers," Computers & Electrical Engineering, vol. 47, pp. 173-185, 2015.

[22] T. Mahdhi and H. Mezni, "A prediction-Based VM consolidation approach in IaaS Cloud Data Centers," Journal of Systems and Software, vol. 146, pp. 263-285, 2018.

[23] F. Farahnakian et al., "Using ant colony system to consolidate VMs for green cloud computing," IEEE Transactions on Services Computing, vol. 8, no. 2, pp. 187-198, 2015.

[24] H. Mi, H. Wang, G. Yin, Y. Zhou, D. Shi, and L. Yuan, "Online self-reconfiguration with performance guarantee for energy-efficient large-scale cloud computing data centers," in Services Computing (SCC), 2010 IEEE International Conference on, 2010, pp. 514-521: IEEE.

[25] H. Li, G. Zhu, C. Cui, H. Tang, Y. Dou, and C. He, "Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing," Computing, vol. 98, no. 3, pp. 303-317, 2016.

[26] H. Zhao, J. Wang, F. Liu, Q. Wang, W. Zhang, and Q. Zheng, "Power-aware and performance-guaranteed virtual machine placement in the cloud," IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 6, pp. 1385-1400, 2018.

[27] B. Wei, C. Lin, and X. Kong, "Dependability modeling and analysis for the virtual data center of cloud computing," in High Performance Computing and Communications (HPCC), 2011 IEEE 13th International Conference on, 2011, pp. 784-789: IEEE.

[28] N. B. Fuqua, "The applicability of markov analysis methods to reliability, maintainability, and safety," Selected Topic in Assurance Related Technologies (START), vol. 2, no. 10, pp. 1-8, 2003.

[29] K. S. Trivedi, Probability & statistics with reliability, queuing and computer science applications. John Wiley & Sons, 2008.

[30] A. Goyal, S. S. Lavenberg, and K. S. Trivedi, "Probabilistic modeling of computer system availability," Annals of Operations Research, vol. 8, no. 1, pp. 285-306, 1987.

[31] J. F. Meyer, "Closed-form solutions of performability," IEEE Transactions on Computers, no. 7, pp. 648-657, 1982.

[32] F. Machida, D. S. Kim, and K. S. Trivedi, "Modeling and analysis of software rejuvenation in a server virtualized system with live VM migration," Performance Evaluation, vol. 70, no. 3, pp. 212-230, 2013.

[33] J. K. Ghosh, "Introduction to Modeling and Analysis of Stochastic Systems, by VG Kulkarni," International Statistical Review, vol. 80, no. 3, pp. 487-487, 2012.

[34] 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 and Computation: Practice and Experience, vol. 24, no. 13, pp. 1397-1420, 2012.

[35] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software: Practice and experience, vol. 41, no. 1, pp. 23-50, 2011.

[36] K. Park and V. S. Pai, "CoMon: a mostly-scalable monitoring system for PlanetLab," ACM SIGOPS Operating Systems Review, vol. 40, no. 1, pp. 65-74, 2006.

[37] R. d. S. Matos, P. R. Maciel, F. Machida, D. S. Kim, and K. S. Trivedi, "Sensitivity analysis of server virtualized system availability," IEEE Transactions on Reliability, vol. 61, no. 4, pp. 994-1006, 2012.

Monireh Hosseini Sayadnavard received the B.S. degree in computer engineering from Islamic Azad University, Lahijan, Iran, in 2005, the M.S. degree from Islamic Azad University, Qazvin, Iran, in 2010. She is now a Ph.D. candidate at Islamic Azad University (Tehran Science and Research branch). Her research interests include cloud computing, distributed systems and evolutionary computing.

Abolfazl Toroghi Haghighat received his B.S. and M.S. degrees in Electrical Engineering from Sharif University of Technology, Tehran, Iran, in 1993 and 1996, respectively. His Ph.D. degree from Amirkabir University of Technology (AUT), Tehran, Iran, in 2003. He is an Assistant Professor of Computer Engineering and Information Technology at Islamic Azad University of Qazvin. His research interests are in wireless networks, pattern recognition, fault-tolerant computing, and distributed systems.

Amir Masoud Rahmani received his B.S. in computer engineering from Amir Kabir University, Tehran, in 1996, the M.S. in computer engineering from Sharif University of technology, Tehran, in 1998 and the Ph.D. degree in computer engineering from IAU University, Tehran, in 2005. He is a professor in the Department of Computer Engineering at the IAU University. He is the author/co-author of more than 150 publications in technical journals and conferences. His research interests are in the areas of distributed systems, ad hoc and sensor wireless networks, scheduling algorithms and evolutionary computing.