AKADÉMIAI KIADÓ

A survey on application of machine learning to manage the virtual machines in cloud computing

VARUN BARTHWAL*, M.M.S. RAUTHAN and ROHAN VARMA

Hemvati Nandan Bahuguna Garhwal University, Uttarakhand, India

International Review of Applied Sciences and Engineering

11 (2020) 3, 197–208

DOI:
10.1556/1848.2020.00065

Received: November 2, 2019 • Accepted: January 29, 2020
Published online: October 5, 2020

REVIEW PAPER

ABSTRACT

Virtual machine (VM) management is a fundamental challenge in the cloud datacenter, as it requires not only scheduling and placement, but also optimization of the method to maintain the energy cost and service quality. This paper reviews the different areas of literature that deal with the resource utilization prediction, VM migration, VM placement and the selection of physical machines (PMs) for hosting the VMs. The main features of VM management policies were also examined using a comparative analysis of the current policies. Many research works include Machine Learning (ML) for detecting the PM overloading, the selection of VMs from over-utilized PM and VM placement as the main activities. This article aims to identify and classify research done in the area of scheduling and placement of VMs using the ML with resource utilization history. Energy efficiency, VM migration counts and Service quality were the key performance parameters that were used to assess the performance of the cloud datacenter.

KEYWORDS

cloud computing, datacenter, dynamic consolidation, machine learning, prediction, virtualization, workload data

1. INTRODUCTION

Users request resources on-demand and execute the application using VM resources that fit according to application needs. In most of the Cloud systems, VM management was done according to their current resource utilization. The VM management process requires additional computation and memory resources that increase system cost and affects overall performance. In our survey, the learning of system behavior was found as an appropriate technique to develop more refined VMs management process. This may be proposed according to the past, overall, long-term utilization levels. It is a fundamental challenge to allocate VMs onto PMs, while maintaining the service level agreement (SLA) requirement, system performance, optimum utilization of resources, and reduction in energy consumption (EC). Overloading and under-loading of PMs is also the problem that occurred in the placement process, to avoid such problems the prediction of resource utilization was done in most of the research. The prediction of resource utilization is not only based upon current usage patterns but also past system behavior may be considered. In this review, it was found that machine learning (ML) models are suitable to predict resource utilization using historical data to achieve effective VM scheduling and placement.

This paper reviews different areas of literature that deal with the Virtual Machines (VMs) placement onto physical machines (PMs). This survey aims to identify and classify research done in the area of scheduling and placement of the VMs using a ML approach with resource utilization history. Most of the research works have been done to improve VM scheduling and placement in the cloud system using current and historical resource utilization. ML models are rarely used in literature; however, significant tasks have been done in regression-based solutions, Artificial Neural Networks (ANNs), and reinforcement learning (RL).

*Corresponding author.
E-mail: varuncsed1@gmail.com

1.1. Virtualization

The technique of virtualization has the aim of efficient management of extremely large datacenter by executing different operating systems in isolation on individual host or PM. It performs as a layer between computer hardware and the operating system. The hardware resources split into the logical unit known as VMs and virtualization enable the system to place more than one VM to share the resources of a single PM. Each VM hosts the operating system and has access to the hardware resources of PMs. This technique enables cloud datacenter with more flexibility and provides better support for on-demand resource allocation using VM migration. Migration is a mechanism in which a VM is relocated from one PM to another without any interruption [1]. However, it reduces the performance of running applications in a VM [2]. Virtualization is the key characteristic of cloud framework that differentiates cloud computing from the earlier computing paradigm (e.g., grid computing, distributed computing, parallel computing). It provides dynamic management of VMs, cost-effective utilization, and PM resources to different users with isolation. VMware, XEN hypervisor, and Kernel-based Virtual Machine (KVM) are widely used virtualization platforms to create a virtual environment in a cloud datacenter.

1.2. Workload data

Workload traces are the resource usage data at a certain period which are stored in text files, e.g., in PlanetLab [3]. PlanetLab data is provided as a part of the CoMon project and contains the CPU utilization of more than a thousand VMs from servers located at more than five hundred places around the world. The interval of utilization measurements is 5 min. Each file is associated with a VM and each line has a number that represents the CPU utilization of PMs. It is a load as a percentage of requested CPU resource capacity in some data samples. The simulator, e.g., CloudSim [4, 5], based experiments were performed for evaluating the various models and algorithms using workload traces from a real cloud system. Real test data are useful to obtain relevant results for the performance analysis of the cloud environment that is a good initiative to foster research in cloud computing. It contains data related to hardware machine resource utilization in a file and known as workload traces. The workload is also known as jobs that contain one or more than one task which is independently running software units. The Required CPU, memory, and disk capacity are mentioned for each task in workload and regularly measured in time (In Planet Lab, the sample data period is 5 min). Some workload traces contain resource consumption values as one file per VM and also describe the dynamic data of VM. It includes the used CPU, memory, disk and network Input-Output (I/O) values.

Bitbrains is a specialized service provider for hosting and managing business computing for enterprises. GWA-T-12 Bitbrains dataset consists of traces in the files where each file represents the performance metrics of 1750 VMs. These files are arranged as fastStrorge and Rnd traces. The first one, the fastSorage, consists of 1250 VMs that are connected to fast storage area network and Rnd traces are represented using 500 VMs that are connected to relatively slower storage devices. The format of each file is row-based and each row represents an observation of the performance metrics [6].

Many researchers also used TPC [7], RUBiS [8], and SPEC [9] benchmarks to test the system performance. These are the benchmarks that were used to generate workload and further used for the assessment of the performance of the proposed system in various researches.

The PlanetLab dataset was mostly used to predict and manage the resource in the cloud datacenter. CPU was considered as the main resource in most of the research work, and it has to be managed in the datacenter to maintain the consumption of energy (Table 3). So to deal with only CPU values, it was found that PlanetLab is the preferable choice among researchers and still, it is used to predict resource utilization, detection of PM overloading and underloading, VM selection, and migration.

The remaining part of this paper has been arranged in various sections as follows; a brief explanation and related work have been presented in Section 2 that also described the stepwise approach to predict and manage the resources in the cloud datacenter.

The various ML models with their applications to manage the resources in the cloud datacenter have been discussed in Section 3. This section also demonstrated the research done in the related field. A brief description of the work done in previous years along with the ML model and workload data used have been given in Table 3 with the various objectives achieved using different ML models and workload data.

2. PREDICTION AND MANAGEMENT OF RESOURCES

Datacenter consists of several PMs or servers equipped with a large amount of CPU, memory, disk and bandwidth (BW) resources. The efficient utilization of these resources is the primary task done by the cloud vendor to maintain the service quality and power consumption cost. Maintaining the quality of service and power cost are the major research challenges in cloud computing. To achieve this, various research works have been successfully done in the field of optimum utilization of resources in the datacenter while considering the VMs scheduling and placement as a primarily focused task. The cloud service providers have to provide optimized VMs placement in the datacenter while maintaining the SLA and power consumption cost. This is an NP-hard problem and can be solved using a bin packing approach [15]. A lot of methods were proposed using heuristic, meta-heuristic and ML-based solutions for optimization. In most of the ML methods, historical data or workload traces were used as a required input to predict future resource utilization. The prediction of resource utilization

requires system behavior by learning hence various algorithms were developed to make an accurate prediction for VM management. VMs management is applied when the required resources are being used completely. But in the case of less resource utilization PMs suffer the under-loading situation. Moreover, due to static allocation, these resources cannot be redistributed among PMs. The solution for such a problem is dynamic VMs consolidation that can be developed using the various ML techniques. In dynamic VM consolidation, the selection of VMs was done for migration in suitable PM to avoid PM overloading and under-loading. Most of the research work was done for dynamic resource management using the ML strategies on current and past resource usage data. This data only provides information about utilization values in a particular duration of time. It does not provide any information about resource utilization in the future, so various approaches are being developed to predict the utilization of resources.

Providing cloud services and maximizing their capital gain is still a challenge, while maintaining the SLA and power consumption cost. To achieve this, the efficient management of resources can be done using the prediction of the future workload and the utilization pattern. Therefore, ML-based techniques are useful approaches for prediction and found better for resource allocation and management.

Our findings regarding the work done related to the prediction and management of resources in the cloud datacenter were mentioned as follows:

2.1. Resource utilization monitoring and storing

Resource utilization monitoring and storing were used to keep data in a file (utilization traces) that contains the utilization of resources in a fixed time interval, e.g. Planet Lab data. Utilization traces are required to predict the system behavior so that future utilization of resources can be estimated for successful VM scheduling and placement in the cloud datacenter. Processing, memory, disk, bandwidth and I/O utilization records are to be maintained in a file by utilization monitor so that the cloud system can easily analyze the utilization statistics of VMs and PMs for prediction. Resource utilization of VMs is mainly concerned with the utilization percentage in a time frame (e.g., Five minutes in case of Planet lab data) and PMs Resource utilization statistics are considered as total utilization of VMs (hosted by the corresponding PM).

The agent-based approach was used to store VM behaviors (resource utilization over time) in a global database to make accessible for all agents [14]. RL based learning agent observed the PM status and collects the current total utilization of PM [19]. An approach was presented to monitor and collected PM resource utilization using the local agent and the global agent respectively to develop migration mechanisms [20]. The monitoring of resources was done to collect resource utilization data and collected on different periods of each second, 1, 10, and 30 min of resources [25]. A monitoring engine was implemented to collect resource utilization data from VMs. It was used to collect resource usage data in a small interval of time for preparing that data in a file. The data contained CPU, memory, disk and IO statistics in terms of usage [29].

2.2. Analysis of workload data and prediction

Historical data of datacenter can be accessed from cloud traces which contain the utilization percentage of CPU, memory, disk, BW, and I/O resources of PMs by VMs in a certain time interval. This is further used for statistical analysis or training data for the prediction of resource utilization. The researcher used historical data to predict CPU utilization, so the training dataset was collected for each PM and each VM has been assigned a workload from traces as an input that consists of the total values of resource utilization of all VMs in a PM.

A model based on RL for adaptability and scalability was performed by integrating the neural network with RL, this model relates the current configuration, action, and the reward. Further, it was trained from previously collected samples to predict future values [10]. An algorithm was developed to predict future load requirements using a neural network. It was presented with the aim to optimize the EC in the cloud datacenter and minimize the number of PMs making them shutdown or start when required. The algorithm took the PM list and their current states as input and decided turning on or off PMs according to the requirement. If the number of active PMs was less than required PMs, then the algorithm will turn on the more PMs, else shutdown the unnecessary PMs [11]. Historical data were generated by running the standard client-server benchmark. Then the prediction model was trained and tested in order to predict the resource requirement in the next time interval. The continuous training was performed to make accurate predictions on future resource requirements [16]. Linear regression (LR) was applied to develop LiRCUP for prediction, which took historical CPU utilization as input and established a linear relationship between the future and current CPU utilization. Coefficient parameters of regression equations were initialized randomly and dependent variables were considered as expected utilization while independent variables represented current utilization values [17]. K-NN was applied for the forecasting of resource requirements using historical data. K-NN–UP algorithm fetched historical dataset as an input and predicted the CPU utilization. The algorithm used the past and current CPU utilization as the input dataset. The CPU utilization was the output data in the next time instance to forecast CPU usage in each PM. This predicted CPU utilization values were further used for detecting overloaded and underutilized PMs [18]. CPU utilization values were used for training prediction models to forecast the upcoming resource utilization. The network received current and previous CPU utilization values as input and calculated the output as predicted utilization. In the training phase, this output was compared with the real CPU traces and the difference was considered as error. That error was propagated back and weights were adjusted to minimize the prediction error [24]. A resource monitor was

used to collect the utilization of PM resources on a different time interval. Further, these are stored in a buffer and read using the preprocessing unit to make them smooth and sent for the normality test. If data passes the normality test, then Autoregressive Neural Network (AR-NN) was used for the prediction and prediction results were stored in a buffer. AR-NN consists of three layers, the input layer takes eighteen-lagged inputs, the hidden layer to perform processing of data and forwarding to the output layer of the single neuron. The workload traces were collected from randomly selected VMs from fastStorage data [25]. The CPU demand on PM from the previous time instance was forwarded through the network to predict future CPU demand as an output signal. An input signal corresponding to CPU demand on PM from the previous time steps was considered as an input for the input layer neurons, then the hidden layer processes the data and finally predicted output was calculated using output layer neurons [30].

2.3. Detection of overloaded and underutilized PM

Various algorithms were developed to detect overloaded PMs in simulation using historical data as an input to the system. These values are read from workload data and assign to VMs for a given time interval as resource requirement, then VMs are mapped to PMs for placement according to the algorithm. In this allocation mechanism, some PMs may be overloaded. The prediction of future utilization has been done using the ML algorithm to achieve early detection of overloaded PMs. PM overload detection algorithm executes periodically to identify when a PM is overloaded. It requires PM resource utilization, which are processing, memory, and network BW utilization in a fixed time interval.

2.4. Migration of VMs from the overloaded and underutilized PM

Detection of overloaded or under-loaded PMs initiates the process of VMs migration and keeps balancing the load on PMs. All VMs from underutilized PMs are migrated to suitable PMs and underutilized PMs are turned off. In the case of PMs overloading, some of VMs are to be processed for migration from overloaded PMs to get the optimum load balancing. The selection of VMs for migration from overloaded PM is a key task. When VM is migrated to PM, then it is necessary to reconfigure the VM for new PM, in order to achieve improvement in performance. VM selection techniques were applied to transfer VM which reduced more power consumption after migration in comparison to other allocated VMs in the same PM.

Ahmad et al. reviewed about VM migration, server consolidation, network BW optimization, and power consumption. In their review work, strength and weakness of VM migration mechanisms were discussed. They identified various issues in the current solution and highlight the recent trends in VM migration [32].

2.5. Selection of PM

The selection of PM to place migrating VMs is done to complete the consolidation process to maintain the SLA and power consumption cost. In a cloud datacenter, VMs are required to place in appropriate PM for the best service delivery and minimum power cost. Migrating VMs from overloaded PMs were placed using various techniques, in this survey we found that the RL technique was applied for the selection of PM. The algorithm, based on Best RAM and bandwidth (BRB) was developed in which,PMs with enough resources for VM was assigned a mark (as a ratio of the required VMs random access memory (RAM) and available BW of PM) and the lowest mark PM was opted for placement of migrating VM [27]. In order to achieve the optimum cost of EC and SLA, the process of VM scheduling and placement optimization (Algorithm 1) is started from the prediction of resource utilization to detect overloaded (Algorithm 2) and underutilized PMs. All VMs are migrated from the underutilized PM and PM has to be made switched off or in sleep mode for maintaining the power consumption cost. VM is selected (Algorithm 3) from overloaded PM for the migration to maintain the load and finally, a PM is selected to place the migrated VM (Algorithm 4) [14].

ML models cover almost all goals to perform resource management in the cloud datacenter. Hence, it clearly indicates that it may be an acceptable approach to manage resources in the datacenter. The majority of work is done using ANN and LR for resources prediction while RL and SVM were used mainly for VM management activities.

Algorithm 1: VM Management Process

Input: PM list **Output:** PM-VM mapping
1. for each PM in Pm list **do**
2. Resource utilization monitoring
3. Generation of utilization history
4. History analysis
5. Prediction of resource utilization in the next time frame
6. PM overload detection
7. if (PM is overloaded == true) **then**
8. VM selection from overloaded PM
9. initiated VM migration
10. PM selection for migrating VM
11. VM placement

Algorithm 2: PM Overload Detection

1. for each PM in PM list **do**
2. Apply prediction model on utilization history
3. Prediction of resource utilization
4. if (predicted utilization > threshold)
5. PM is overloaded
6. Added to overloaded PM list

Algorithm 3: VM selection for migration

1. for each overloaded PM **do**
2. Apply VM selection policy
3. Selection of appropriate VM for migration
4. Added to migrating list

Algorithm 4: VM placement

1. Input: PM list, migrating VM list Output: PM-VM mapping
2. for each PM (overloaded PM is excluded) **do**
3. for each VM in migrating list **do**
4. if (PM is overloaded after allocating VM) **then**
5. continue
6. else
7. apply VM placement policy
8. selection of appropriate PM for VM placement

Above algorithms are based on the various work presented in this survey, these algorithms present the entire VM management process. This process starts with resource utilization monitoring then utilization history is generated for further analysis of the utilization prediction. Prediction can be done using LR, ANN, or K-NN. This predicted value of utilization is further used to find out the overloaded PMs, if predicted value is larger than a certain threshold then PM is considered as overloaded otherwise it can accommodate new VMs. When a PM is found overloaded then the suitable technique is required to find out the VM for migration. VM is selected in such a way that PM overloading chances should be migration duration minimum. Finally, the VM placement policy is developed to find out a suitable PM for hosting the migrating VMs. This process is demonstrated using an example in which a cloud environment was simulated and results were generated using CloudSim.

A simulation environment is created in CloudSim to represent the cloud datacenter, in which 800 PMs were used. The details of the cloud environment are mentioned in Table 1. We simulated this cloud setup and performed simulation using the PlanetLab workload (for the day 03/04/2011). The workload was fed into the simulator and after certain time when sufficient CPU utilization history of PMs is available then Algorithm 2 finds out the overloaded PMs and calculates the predicted utilization of PMs. Algorithm 2 was implemented using various ML models to predict the CPU utilization for the next time interval. The selection of a suitable VM from the overloaded PM was done using Algorithm 3, the selected VM was further migrated to the appropriate PM for the placement using algorithm 4. ML model was also applied for the VM selection and placement. The entire process is known as dynamic consolidation of VMs, which is implemented to address the EC and SLA violation issues in a cloud datacenter.

EC, SLA and VM migration was the key parameter to assess the performance of most of the developed solutions in literature. We have executed the solution developed by [15] for the cloud environment mentioned in Table 1, the findings are mentioned in Table 2:

Amir et al., addressed the problems of dynamic consolidation of VMs, they presented a survey and taxonomy for consolidation methods of server in datacenter. They described various parameters and algorithms to perform dynamic consolidation of VMs. They also classified the server consolidation techniques in terms of time, constraints, requirements and algorithmic with their objective functions and evaluation methods [33].

Table 1. Cloud environment

Specification	Physical machine type HP proliant ML110		Virtual machine type				Workload data	No of VMs	PM overload detection	VM selection	VM placement
	G5	G4	Type I	Type II	Type III	Type IV	Planet lab	1463	Algorithm 2	Algorithm 3	Algorithm 4
Computing capacity	1860 MIPS (2 cores)	2660 MIPS (2 cores)	500 MIPS	1000 MIPS	2500 MIPS	2500 MIPS	Day: 03/04/2011				
Memory	4096 MB	4096 MB	613 MB	1.7 GB	1.7 GB	0.85 GB					

Table 2. Performance metrics

EC (kWh)	SLA (%)	VM migration
219.64	0.00451	38,104

3. VM MANAGEMENT METHOD BASED ON MACHINE LEARNING (ML)

In most of the research, various methods have been developed for predicting CPU utilization using workload traces. Various ML models can be used to develop VMs scheduling strategies for load balancing of PM resources. ML models provide a capability in cloud systems to learn automatically and improve the system performance. It was used for VM scheduling and placement in the cloud data center. It also develops the mechanism to learn and predict future resource utilization using historical data. The process of scheduling begins with observations of historical data e.g. past CPU, RAM, disk, and BW utilization to find out the patterns in the data to make better predictions for future utilization. In the environment of a cloud system, the huge amount of data is processed with high computing requirements. Hence, resource management with optimized EC, cost, and maintaining the SLA is a critical issue in the management of VMs. So to deal with large data and huge resource requirements, ML may be used to improve VM management in the cloud datacenter. A better prediction can be made using the ML model to provide effective management of resources and deduce the EC in the datacenter. Workload data is also useful for the ML model to learn system behavior and the prediction of resource utilization. The prediction of the workload is helpful to develop a cost-effective cloud system. It was found in our survey that CPU utilization prediction is the most trending research area in cloud computing for the development of VM management process while maintaining SLA and EC.

It was observed in our survey that starting from 2009 ML model was applied on historical data to develop an optimized mechanism for VMs and PMs mapping and still it is in practice. Hence, learning using past resource usage data is a widely accepted approach among the research community for the scheduling and placement of VMs in the cloud datacenter. ML-based algorithms were used in literature to improve resource allocation, scaling and live migration of VMs in a cloud system. This survey explains about ML algorithm with its application in cloud resource management. RL [31], LR [34], Support Vector Machines (SVMs) [35], ANN [36], and K-Nearest Neighbors [37] ML models have been applied to develop a selfadaptive mechanism in the cloud datacenter. In this survey, above ML models based application for resource management are deeply studied and analyzed with their role on VM scheduling, migration, placement, suitable PM selection, load balancing, SLA management and power consumption of datacenter. Year-Wise research work was also mentioned in Table 3, which shows the various ML models along with corresponding workload data.

3.1. Linear regressions (LRs)

The LR model can be applied for the forecasting of the upcoming workload in a cloud system. It uses the number of utilization values at a given point of time in certain time intervals (e.g., planet-Lab data). This is applied to establish the relationship between the predicted and the current workload. It can be observed that the cloud workload trend is linear for a very short time interval [15]. So, an LR model can be used to solve such a problem. It is used to predict the real utilization (e.g., CPU, for the single resource) based on continuous values. LR can be used as an efficient statistical tool to develop a relationship between the independent variable and the dependent variable. The relationship is established between independent (current utilization in a given point of time) and dependent variables (predicted utilization) using the regression line. The LR model can predict future utilization values by analyzing the history of resource utilization. The slope and intercept of the regression line are updated based on past utilization values to predict future utilization. The difference between the predicted and real values of resource utilization in each data point is to be minimized using this method to predict the approximate correct utilization. Most of the regression-based approach follows the single VM resource management e.g. CPU utilization and provides VMs migration based on CPU utilization by considering the relationship between the total power consumed by PM and its CPU utilization. The value of CPU utilization was considered as the main parameter to manage SLA and power costs. However, RAM and BW utilization are rarely considered for VM scheduling and placement, but their impact on power consumption and SLA cannot be neglected. Multiple regression approach can be used for dynamic VMs consolidation considering CPU, Memory, Disk, Network BW utilization as the factor to detect overloaded PMs using multiple regressions. Multiple regression is a simple LR that predicts the value of the dependent variable using two or more other independent variables. It is to learn more about the relationship between more than one independent variable and a dependent variable. CPU, memory, disk, BW and I/O utilization values can be used to enhance VMs scheduling and placement using multiple regression algorithms to predict future resource requirements.

Farahnakian et al., developed a LR method for forecasting the upcoming CPU utilization of PM (LiRCUP) using the PlanetLab historical data. They have established a linear relationship between the future and current CPU utilization in which expected utilization is the dependent variable and current utilization is the independent variable. Their model detected overloaded PMs by comparing the predicted CPU utilization value with current utilization and maintained the SLA and power consumption using the migration of some VMs from the overloaded PMs. LiRCUP algorithm was developed to detect overloaded PM using past

Table 3. Algorithms used in literature for VM management in the cloud datacenter: Year wise study

Paper	ML model	Data source	Work done	Performance metrics	Year	Resource
Rao et al. [10]	RL	TPC Benchmarks	VM auto-configuration	SLA, Response time, Throughput	2009	CPU & Memory
Vinh et al. [11]	ANN	NASA & ClarNet	Load prediction, PM selection	EC	2010	CPU
Kousiouris et al. [12]	ANN	Matlab Bechmark	VM performance analysis and placement	Scheduling and placement decisions of VMs	2011	CPU
Niehorster et al. [13]	SVM	RUBiS	Resource configuration, VM placement	Service level objectives	2011	CPU & Memory
Xu et al. [14]	RL	TPC Benchmarks	VM auto-configuration	SLA, Response time, Throughput	2012	CPU & Memory
Islam et al. [16]	ANN/LR	TPC-W	CPU utilization Prediction	SLA satisfaction	2012	CPU
Farahnakian et al. [17]	LR	PlanetLab	Utilization prediction, PM overload and under load detection	EC, VM migration, SLA	2013	CPU
Farahnakian et al. [18]	K-NN	PlanetLab	Utilization prediction	EC, SLA	2013	CPU
Farahnakian et al. [19]	RL	PlanetLab	VM placement	EC, SLA	2014	CPU
Farahnakian et al. [20]	K-NN	Planet Lab	Utilization prediction	EC, VM migration, SLA	2015	CPU
Duggan et al. [21]	RL	PlanetLab	Network aware VM migration	EC, VM migration, SLA	2016	CPU
Duggan et al. [22]	RL	PlanetLab	VM selection	EC, VM migration	2016	CPU
Patel et al. [23]	SVM/SVR	Real Data	Dirty page prediction, Live VM migration	Migration time, Service time, Total transferred pages	2016	CPU & Memory
Duggan et al. [24]	ANN	PlanetLab	CPU utilization prediction	CPU utilization	2017	CPU
Qazi et al. [25]	ANN	FastStorage	CPU utilization prediction	CPU utilization	2017	CPU
Abdelsamea et al. [26]	MLR	PlanetLab	PM overload and under load detection	EC, VM migration, SLA	2017	CPU
Khoshkholghi et al. [27]	LR	PlanetLab	Utilization prediction, PM overload under load detection, VM selection	EC, VM migration, SLA	2017	CPU
Shaw et al. [28]	RL	PlanetLab	PM–VM mapping	EC, VM migration, SLA	2017	CPU
Stelios et al. [29]	SVM/SVR	YCSB, Real data	VM placement	CPU resource utilization	2018	CPU & Memory
Mason et al. [30]	ANN	PlanetLab	CPU utilization prediction	CPU resource utilization	2018	CPU

utilization values and determined the future CPU usage in PM, if the value of predicted utilization was found larger than the currently available CPU resource then PM is in the overloaded state [17]. Multiple regression was applied to find out the overloaded PMs using CPU, RAM, and BW utilization. These values were used as the independent variables in the multiple regression equation to predict the utilization at the next time interval. If utilization was more than or equal to one then PM was considered as overloaded. Abdelsamea et al., proposed a method that uses multiple regression where CPU, RAM, and network BW were used for the detection of PM overloading and significantly reduced EC. They used multiple factors to perform VM management while maintaining the consumption of energy and SLA [26]. Khoshkholghi et al., proposed a dynamic and adaptive energy-efficient management of VMs using iterative weighted linear regression (IWLR) to detect overloaded and underutilized PMs. They reduced the consumption of energy in the cloud datacenter and guarantee the system performance regarding CPU, RAM, and BW. IWLR was applied for predicting the PM utilization which determines the upper and prethreshold. If the predicted utilization of the resource is greater than or equal to one, then PM will not accept new VM and moved to under pressure list, PM will be moved to the overloaded list if the upper threshold is greater than or equal to one [27]. Islam et al., developed a technique for the prediction of future resource requirements using the LR method. They used historical data generated by running the standard client-server benchmark (TPC-W) as the input data set. CPU usage percentages of all VMs were used to train the system for prediction [16].

3.2. Support vector machine (SVM)

SVM is used to provide a solution for multi-dimensional modeling problem using the concept of decision hyperplane to define decision boundaries which separate the set of objects of different classes. It is a supervised ML model, which is used for the classification of data and regression [35]. SVM was used to develop dynamic and adaptive VM scheduling algorithms based on resource utilization traces of VMs and PMs. The resource analyzer prepared the data set in a regular interval using PMs and VMs utilization history. This dataset was used to predict future resource utilization for VMs scheduling and placement methods. VMs scheduling optimization was done by defining the weights of the PMs according to the VMs resource utilization using SVM. Historical data of PMs and VMs were evaluated and classified according to the overall resource utilization [29]. In some research support vector regression (SVR) was also used as a learning approach and SVM classify data and make predictions effectively on overlap regions of two classes, it is widely used for forecasting and classification [23].

Niehorster et al., presented an approach for the provisioning of VMs using SVMs. They developed an algorithm to perform a feasibility check, which takes the application, workload, service level objectives (SLOs) and possible resource configuration (resource allocation to the VMs) as input and classify the possible configuration into two sets, in which one set satisfies the SLO and another does not. Possible configurations that were closer to the hyper-plane were considered as selected configuration [13]. They developed a selfconfigurable and self-optimized multi-agent system to learn the behavior of the system to estimate the cost. Their system learns performance models of various applications and derives a behavior model from the data collected from all agents during their runtime and then SVM is applied to classify the data of the knowledge base.

Patel et al., developed an SVR based model to predict dirty pages and configured the live migration of VMs. Their algorithm configured the live migration of VMs in the datacenter using dirty pages analysis and prediction, to make the migration process effective. Dirty pages prediction algorithm based on the SVR model took time series data of dirty pages and predicted as output series. They have measured the total migration period and live migration performance. They also developed Autoregressive Integrated Moving Average (ARIMA) based model and experimental results state that SVR predicts dirty pages with more accuracy than ARIMA. Migration time and total transferred pages were used as the main performance indicators to assess the live migration performance of their proposed system [23].

Sotiriadis et al., presented the VM scheduling approach which uses extracted data from past resource utilization of VMs and PMs with the aim of defining the weights of PMs according to the resource utilization of hosted VMs on that PM. They considered the dataset of resource utilization (CPU, memory, and disk usage percent) in X, Y plane and represented those data by vectors then apply SVM to classify VM status according to historical records. Filtering and weighing methods were applied in VM scheduling for the selection of PM for hosting the VM. Suitable PMs were selected to host VMs after the filtering process, in this process, it is checked that whether a PM has enough resources to host VM. The weighing process assigns high weights to best PMs based on historical records. The experimental results show that their solution improved PM selection by learning the system behavior [29].

4. ARTIFICIAL NEURAL NETWORK (ANN)

ANN has an input layer, hidden layers, and an output layer. Each layer has certain numbers of nodes (neurons), integrated with the sigmoid activation function. This technique provides selflearning mechanisms based on the concept of neurons, connections and transfer functions. Research works based on neural networks indicate that the prediction of resource utilization is an effective approach to manage the resources in the cloud datacenter. The neural network is used as an efficient tool for forecasting in various kinds of research problems. It was also applied to forecast resource requirements in cloud computing. It can be used to extract and analyze actual workload patterns to predict the

upcoming workload on the datacenter in the next time instance. Historical data or workload traces were divided into training and testing data for the prediction model. In most of the research, the CPU resource requirement at each time interval in the historical data was used mostly as an input data for the prediction of resource utilization. Further, it was taken as an input for the neural network to propagate through the network with a certain weight. The output layer was used to provide an output signal after the processing of neurons.

Hence, the input signal was considered as the CPU requirement at a certain time. CPU requirement for the next time interval was represented by the output signal. Neural networks are better ML approaches to predict CPU utilization. The prediction of future CPU, Memory, Disk, network or other required resources based on historical data will identify unused, under-loaded and overloaded PMs. This knowledge is helpful to determine which PM should be made ideal and from which PM migration of VMs will be done to reduce consumption of energy and the cost of the datacenter.

Vinh et al. [11], presented an energy-aware method to forecast future load requirements using neural networks based on historical data and reduced the number of PMs by making them shutdown or start when required. Their work aimed to optimize the power consumption in the cloud datacenter. Their system turns on/off some PMs when the system load decreases/increases.

Kousiouris et al. [12], focused on allocation percentages, co-placement of VMs and real-time scheduling in the same PM based on the prediction of the various parameters' effect on VMs performance. They optimized ANN by genetic algorithm and investigated the degradation in prediction.

Islam et al. [16], applied neural networks for resource allocation and implemented adaptive management of resources in the cloud system. They trained the neural network using the back-propagation algorithm and their experimental results shown that Neural Network-based predictions have less percentage error than LR.

Duggan et al. [24] developed a method using the recurrent neural network for predicting the future values of CPU utilization. Their neural network was enabled to retain information for making an accurate prediction with the time series data. They also investigated the accuracy of the network for prediction. Experimental results presented that the CPU utilization prediction was possible with high accuracy for fluctuating data sets in terms of changes.

Zia Ullah et al. [25], developed a system for the prediction of resource utilization in real-time that took real-time resource utilization and prepared buffers to store them. These buffers were based on the time span size and type of resources. They applied the Autoregressive Neural Network (AR-NN) model on data buffers where data does not follow Gaussian distribution in the system with the real CPU utilization traces in the datacenter consisting of one hundred and twenty servers. The experimental results displayed that there was an improvement in the results when compared with ARIMA for a data set.

Mason et al. [30] developed a mechanism for predicting the CPU usage of PM using evolutionary Neural Networks and Particle Swarm Optimization (PSO), Differential Evolution (DE), and Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) optimization techniques were used for network training. The experiments results demonstrated that CMA-ES performs best in comparison to other optimization techniques used and trained networks accurately to predict CPU utilization.

4.1. Reinforcement learning (RL)

The RL framework contains state-space, action-space, and a reward signal. In this approach learning agent repeatedly performs interactions with the environment and receives a reward or penalty depending upon the action performed. It enables the learning-agent to achieve the maximum reward using repeated trial and error interactions so that it can learn a policy to select the best action [31]. RL can be used to enable an agent for learning the optimized solution for VMs scheduling and migration. It learns by interaction with the dynamic environment and performs optimization based on given states. This survey found out that the RL approach was mainly applied to develop the live migration, VM selection, configuration and PM–VM mapping mechanisms in the datacenter. Optimization of VMs scheduling, configuration and allocation were automated in the cloud environment using RL based approach. Other ML methods were focused on the utilization prediction of resources.

Rao et al., proposed a VM Configuration technique (VCONF) that was based on the RL method. It automated the VMs configuration process and addresses the scalability and adaptability issues in the system. VCONF generates policies by learning from iterations with the environment to perform the auto-configuration of VMs. VCONF used the current state to compute the required action and initiate VMs reconfiguration. Performance feedback was collected to calculate the reward signal and this reward signal was used to update the configuration policy. VCONF monitored the feedback of performance from each VM to manage the configuration process, the performance of VM defined the reward signal and to achieve optimal configuration, this reward signal has to be maximized. This approach provides the optimized configuration in a cloud environment with better adaptability and scalability. Their Experimental results show VCONFs optimality in controlled problems and good adaptability and scalability in a larger system [10].

Xu et al. developed an unified RL approach that configures VMs and their application automatically with the possible adaptation of the VM resource budget and provides quality assurance in service [14]. They performed their experiments on XEN VMs with different workloads.

Farahnakian et al., presented a method to perform dynamic consolidation of VM using RL, Reinforcement Learning based Dynamic Consolidation (RLDC), which used a learning agent to determine the power policy of PMs. The agent selects PM to make it sleep or active by learning system behavior and optimizes the number of active PMs. The learning agent observed the current status of PM (active or sleep) and acted with the combine reward function of

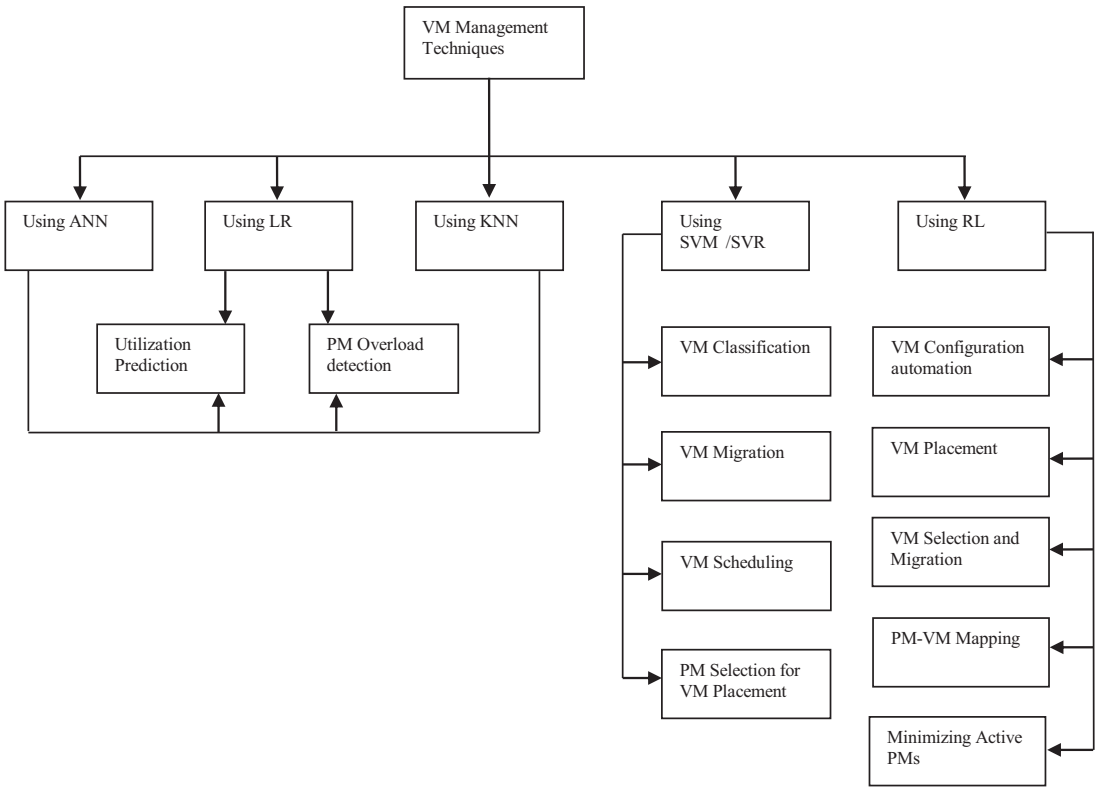

Fig. 1. An overview of VM management techniques using ML

power consumption and SLA violation. RLDC improved the resource allocation process based on the PM state (sleep or active). The learning agent selected a PM to be made sleep after migration of all VM completed. VM allocation algorithms found out a new PM to allocate migrating VMs. RL–DC used LirCUP [17] to predict PM utilization and find out the overloaded PMs. Experimental results with the Planet-Lab traces demonstrated that their model minimizes the consumption of energy and improves performance [19].

Duggan et al. [21] proposed a live migration strategy, Reinforcement Learning network aware Live Migration (RLLM) that was network-aware to monitor the BW demand and performed proper action based on experience when network congestion occurs. Their system performed as a decision support system in which an agent was made able to schedule VM migration by learning an optimized time. The migration process depends on the BW usage in a cloud datacenter. Their experiments' results described that an agent can learn available BW at the peak network saturation and be capable to schedule VMs for migration from underutilized PM at the appropriate time using available BW in a cloud datacenter. In the RLLM algorithm, the agent performed live migration from underutilized PM at the appropriate time considering the BW level and reduced congestion in the datacenter. They have also presented an RL based energy-efficient approach (RLVM) to select VMs for migration from overloaded PMs. RLVM is a VM selection policy for migration from over-utilized PM; this determines which VM is to be migrated from the set of VMs. The overall utilization of PM was calculated using each VM utilization percentage from historical data. SLA violation and power consumption were used as a reward function. This approach was used to determine the suitable time to schedule the VMs for migration considering that one or two VMs migration from underutilized PM had no effect on the cloud resource management process. They used the Local Regression technique [15] to determine overloaded PM. The learning agent decided an optimal VM from overloaded PM for migration and managed the consumption of energy [22].

Shaw et al., presented an RL method Advanced Reinforcement Learning Consolidation Agent (ARLCA) for optimum VM allocation which was capable to optimize the VM distribution in the data center with significant improvement in energy saving and reduction in SLA violation. They developed a state-action space, in which state was defined as the total active PMs as a percentage of the total PMs. The action was a combination of the utilization rate of any PM and the size of the VM to be placed. ARLCA took VM placement list as an input and performed the PM–VM mapping mechanism in the datacenter. The agent determined the global state then each PMs CPU utilization rate and VM size is computed to generate possible actions to map all VMs on to PMs in the datacenter [28].

4.2. K-Nearest Neighbor (K-NN)

It is a supervised learning algorithm for classification and prediction purposes. It performs classification based on the concept of the majority votes of nearest points to the unknown variable. In the case of regression, the average of

closer data values is taken to predict new data. The K-NN classification process predicts the class of the object and K-NN regression predicts the data value of the object [37].

Farahnakian et al., developed the method for dynamic consolidation of VMs that minimized active numbers of hosts or PMs using the present and past historical usage. K-NN method was used to predict the CPU usage of each PM. The Resource utilization was predicted using some sample of the training dataset; each sample contains input and one output variable for modeling the relationship between input/output parameters. Their prediction method was aimed to detect PMs overloading and under-loading to improve the dynamic consolidation of VM. The experimental results presented that the proposed system minimizes the consumption of energy and maintains the SLA [18].

Farahnakian et al., investigated in another paper about the effectiveness of VMs and utilization predictions using workload traces to perform consolidation of VMs. Their proposed solution consolidated VMs dynamically for reducing the consumption of energy and SLA violations. It also allocated the VMs to PMs using current and future usage of resources that result in the reduction of VM migration counts. The migration of VMs from the overloaded PMs was done to reduce the SLA violation. They predicted the CPU utilization using K-NN from historical data. Each data sample contains input and output variables to represent the CPU usage values in the current time and next time instance respectively. The predicted CPU utilization values of VM and PM are aggregated and bounded using the upper threshold of the total capacity. If the predicted utilization of the PM and VM are larger than the threshold value then PM is considered as a destination PM with enough capacity to fulfill current and future resource requirements. The algorithm creates a list of sorted PMs based on their CPU load in decreasing order. All VMs were transferred from the least loaded PM in the list. The selection of VMs is based on their CPU load further the least loaded VM is migrated to PM with the highest load in the list and process will continue to perform resource allocation. The UP–BFD algorithm works with the aim to migrate all VMs from underutilized PMs to reduce consumption of energy. The process of migration of some VMs from overloaded PMs is done to maintain the SLA and load. Their experimental analysis on real workload traces validates the effectiveness of the system for EC, VMs migration and SLA [20].

In Fig. 1, an overview of VM management using ML model was presented based on the survey; it includes the VM management activities that were performed using ML models. It can be seen in Fig. 1, that prediction of resource utilization was done using ANN, LR, and K-NN based ML model while most of the activities related to VM management (VM selection, VM migration, VM placement, etc.) was mainly based on SVM and RL.

5. CONCLUSION AND FUTURE DIRECTION

In this survey, a systematic and deep study was done about the existing work to manage the resources in the cloud datacenter using the ML approach. It was found that most of the research works mainly concentrated on maintaining the consumption of energy and SLAs. ML models were found suitable to develop various solutions that automate resource management, reduce the extra load of manpower and enhance the performance of the cloud datacenter. Various researches were done to optimize the scheduling techniques of VMs for the placement in the datacenter using the ML approach. In the survey, it was observed that most of the researchers have used ANN and LR based ML models for prediction. PlanetLab data was used mostly in literature when CPU is considered as the main factor for optimization. The entire survey describes that violation of SLA, consumption of energy, and migration of VMs are main performance parameters to evaluate the VM management algorithm. It was mentioned in Table 3 that LR, K-NN, and ANN-based models are mainly used for resource utilization prediction and detection of overloaded and underloaded PMs, while SVM based models are used for PMs and VMs classification based upon their usage. The work based on the RL method was mainly concerned with VM migration, placement, and configuration in datacenter. As a future direction for further research, an approach may be proposed in which ANN based prediction is used to detect underloaded and overloaded PMs. Moreover, the classification of PMs and VMs may be done based upon their resource usage pattern using SVM. Finally, the RL method is applied to perform VM selection, migration and selection of PM.

REFERENCES

[1] N. Bobroff, A. Kochut, and K. Beaty, *Dynamic Placement of Virtual Machines for Managing SLA Violations*,2007 10th IFIP/IEEE International Symposium on Integrated Network Management, pp. 119–28, 2007.

[2] W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, "Cost of virtual machine live migration in clouds: A performance evaluation," in *Proceedings of the 1st International Conference on Cloud Computing (CloudCom)*, vol. 2009. Beijing, China, Springer, 2009.

[3] K. S. Park and V. S. Pai, *CoMon: A Mostly-scalable Monitoring System for PlanetLab*, ACM SIGOPS Operating Systems Review, pp. 65–74, 2006.

[4] R. Buyya, R. Ranjan, and R. N. Calheiros, *Modelling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities*, International Conference on High Performance Computing & Simulation, pp. 1–11, 2009.

[5] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of Cloud computing environments and evaluation of resource provisioning algorithms," *J. Software: Pract. Exper.*, vol. 41, pp. 23–50, 2011.

[6] GWA-T-12 Bitbrains", http://gwa.ewi.tudelft.nl/datasets/gwa-t-12-bitbrains.

[7] TPC Benchmarks Overview", http://www.tpc.org/information/benchmarks5.asp.

[8] "RUBiS: Rice University Bidding System", https://rubis.ow2.org/index.html.

[9] Standard Performance Evaluation Corporation", http://www.spec.org/web2005/index.html.

[10] J. Rao, X. Bu, C. Z. Xu, L. Wang, and G. Yin, "VCONF: A reinforcement learning approach to virtual machines auto-configuration," *Proc. ICAC*, pp. 137–46, 2009.

[11] T. Vinh, T. Duy, Y. Sato, and Y. Inoguchi, *Performance Evaluation of a Green Scheduling Algorithm for Energy Savings in Cloud Computing*, IEEE International Symposium on Parallel & Distributed Processing Workshops, pp. 1–8, 2010.

[12] G. Kousiouris, T. Cucinotta, and T. Varvarigou, "The effects of scheduling, workload type and consolidation scenarios on virtual machine performance and their prediction through optimized artificial neural networks," *J. Syst. Softw.*, vol. 84, no. 8, pp. 1270–91, 2011.

[13] O. Niehorster, A. Krieger, J. Simon, and A. Brinkmann, *Autonomic Resource Management With Support Vector Machines*, 2011 IEEE/ACM 12th International Conference on Grid Computing, 2011.

[14] C. Xu, J. Rao, and X. Bu, "URL: A unified reinforcement learning approach for autonomic cloud management," *J. Parallel Distributed Comput.*, vol. 72, no. 2, pp. 95–105, February, 2012.

[15] 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," *Concurr. Comput.*, vol. 24, no. 13, pp. 1397–420, 2012.

[16] S. Islam, J. Keung, K. Lee, and A. Liu, "Empirical prediction models for adaptive resource provisioning in the cloud," *Future Gener. Comput. Syst.*, vol. 28, no. 1, pp. 155–62, 2012.

[17] F. Farahnakian, P. Liljeberg, and J. Plosila, *LiRCUP: Linear Regression Based CPU Usage Prediction Algorithm for Live Migration of Virtual Machines in Data Centers*, 39th Euromicro Conference on Software Engineering and Advanced Applications, 2013.

[18] F. Farahnakian, T. Pahikkala, P. Liljeberg, and J. Plosila, *Energy Aware Consolidation Algorithm Based on K-nearest Neighbour Regression for Cloud Centers*, IEEE 6th International Conference on Utility and Cloud Computing, 2013.

[19] F. Farahnakian, P. Liljeberg, and J. Plosila, "Energy-efficient virtual machines consolidation in cloud data centres using reinforcement learning," in *22nd Euromicro International Conference in Parallel, Distributed and Network – Based Processing*, 2014.

[20] F. Farahnakian, T. Pahikkala, and P. Liljeberg, *Utilization Prediction Aware VM Consolidation Approach for Green Cloud Computing*, IEEE 8th International Conference in Cloud Computing, 2015.

[21] M. Duggan, J. Duggan, E. Howley, and E. Barrett, "A reinforcement learning approach for the scheduling of live migration from under-utilized hosts" *Memetic Comput.*, vol. 8, pp. 111, 2016.

[22] M. Duggan, J. Duggan, E. Howley, and E. Barrett, "A reinforcement learning approach for dynamic selection of virtual machines in cloud data centres, in *Conference: Innovative Computing Technology (IN-TECH 2016)*, 2016.

[23] M. Patel, S. Chaudhary, and S. Garg, *Machine Learning Based Statistical Prediction Model for Improving Performance of Live Virtual Machine Migration*, Hindawi Publishing Corporation Journal of Engineering, vol. 2016, p. 9, 2016.

[24] M. Duggan, K. Mason, J. Duggan, E. Howley, and E. Barrett, Predicting host CPU utilization in cloud computing using recurrent neural networks, in *The 12th International Conference for Internet Technology and Secured Transactions (ICITST-2017)*, 2017.

[25] Q. Zia Ullah, S. Hassan, and G. M. Khan, Adaptive Resource Utilization Prediction System for Infrastructure as a Service Cloud, Computational Intelligence and Neuroscience 2017, 2017.

[26] A. Abdelsamea, A. El-Moursy, E. Hemayed, and H. Eldeeb, "Virtual machine consolidation enhancement using hybrid regression algorithms," *Egypt. Inform. J.*, vol. 18, no. 3,pp. 161–70, 2017.

[27] M. Khoshkholghi, M. Derahman, A. Abdullah, S. Subramaniam, and M. Othman, "Energy-efficient algorithms for dynamic virtual machine consolidation in cloud data centers," *IEEE Access*, vol. 5, pp. 10709–22, 2017.

[28] R. Shaw, E. Howley, and E. Barrett, *An Advanced Reinforcement Learning Approach for Energy-Aware Virtual Machine Consolidation in Cloud Data Centers*, The 12th International Conference for Internet Technology and Secured Transactions, 2017.

[29] S. Sotiriadis, N. Bessis, and R. Buyya, "Self-managed virtual machine scheduling in Cloud systems," *Inform. Sci.*, vol. 433–434, no. 2018, pp. 381–400, 2018.

[30] K. Mason, M. Duggan, E. Barrett, J. Duggan, and E. Howley, "Predicting host CPU utilization in the cloud using evolutionary neural networks," *Future Gener. Comput. Syst.*, vol. 86, pp. 162–73, 2018.

[31] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.

[32] 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," *J. Network Comput. Appl.*, vol. 52, pp. 11–25, June 2015.

[33] A. Varasteh and M. Goudarzi, "Server consolidation techniques in virtualized data centers: A survey," *IEEE Syst. J.*, vol. 11, no. 2, pp. 772–83, June 2017.

[34] How To Find Relationship Between Variables, Multiple Regression", http://www.statsoft.com/Textbook/Multiple-Regression.

[35] Support Vector Machines (SVM) Introductory Overview", http://www.statsoft.com/textbook/support-vector-machines.

[36] Model Extremely Complex Functions, Neural Networks", http://www.statsoft.com/Textbook/Neural-Networks.

[37] K-Nearest Neighbors", http://www.statsoft.com/Textbook/k-Nearest-Neighbor.