CHAPTER 2
LITERATURE SURVEY

2.1 INTRODUCTION

Cloud computing is a technology that simplifies sharing of resources such as server, software, hardware and information to the cloud users in an on-demand basis. In order to reduce the power consumption, Virtual Machines (VMs) are incorporated into the cloud infrastructure. VMs efficiently utilize the hardware resources by allowing multiple operating system environments on a single node without any intervention. It is capable of sharing all the physical resources and plays an important role in cloud computing to enhance the scalability. Load balancing is the process of utilizing all VM resources to increase the performance of cloud computing. The workload is distributed among the available resources by efficient resource allocation and task scheduling to achieve user satisfaction. According to a group of polices or rules the tasks are allocated to the VM node by the scheduling process. It involves resource discovery and filtering, resource selection, and task allocation. The objective of scheduling is to sustain a tradeoff between the performance and QoS. It is necessary to achieve load balancing, QoS, high throughput, economic principles and best running time.

2.2 OPTIMAL VIRTUAL MACHINE SELECTION

A novel adaptive heuristics for dynamic consolidation of VMs was proposed by analyzing the historical data and resource usage of the VMs [1]. The workload information and queuing network system model was used to improve the system efficiency. The adaptive provisioning mechanism utilized the available resources and guaranteed the QoS. A Weighted Active Monitoring Load Balancing Algorithm (WAMLBA) was presented to balance loads of the VMs in an IaaS framework by selecting the optimal virtual machine. [2],[3]. The results of WAMLBA algorithm showed that, it was efficient in optimizing the response time and the data processing time. A Fuzzy Multi Criteria Decision Making (MCDM) was proposed to address the ranking inconsistency problem [4]. Three aggregation functions and three defuzzification methods were developed based on the group averaging methods. The results proved that the MCDM was applicable even in large group networks.
A cost effective bulk data transfer application, namely, NetEx was designed to deliver data in an efficient manner [5]. The NetEx achieved high performance and delivered a high data transfer rate. The algorithms for selecting the data center including a throttled load balancing strategy, round robin algorithm, vector dot, compare and balance, and event-driven algorithms were analyzed [6]. These algorithms were simulated using cloud analyst in terms of fault tolerance, migration time, overhead, and response time and resource utilization. An adaptive VM replacement methodology was proposed for multi-dimensional server consolidation in cloud data center [7]. The results of the Vector Packing with Minimal Placement Cost (VP-MPC) achieved higher efficiency and a tradeoff between utilization and replacement cost. A cloud brokering mechanisms for the deployment of VMs based on the pricing policy was presented in a heterogeneous infrastructure [8]. The results showed that it achieved high throughput than the other VM deployment schemes.

A Service Level Agreement (SLA)-based resource provisioning was proposed for heterogeneous workloads in cloud datacenter [9]. The suggested scheduling and administrator control mechanisms increased the resource utilization and profit, and also ensured the SLA requirements. When compared to static server consolidation, the suggested approach reduced the SLA violations. A virtual infrastructure gateway, namely, Nefeli was proposed to map the VMs to the physical nodes [10]. The results of the proposed approach showed that the energy consumption was lesser than the traditional approaches. A database provisioning system, namely, Dolly was proposed for VM cloning in a commercial replication platform [11]. Further various cost functions were proposed to reduce the resource usage in private cloud. The analysis showed that Dolly supported the deployment of replicated databases. An unified resource allocation framework was constructed using Mixed Integer Programming (MIP) to map the resources in an efficient manner [12].

A reconfiguration strategy handled the dynamic allocation of VMs in the cloud environment. The performance of the proposed framework was high and it handled the resource allocation in an efficient manner. A soft computing approach was proposed using Stochastic Hill Climbing (SHC) [13] for load balancing in cloud computing. Complete methods produced guaranteed output, whereas incomplete
methods did not produce guaranteed output for all the cases. The results showed that the SHC approach outperformed the Round Robin (RR) and First Come First Served (FCFS) algorithms. A novel VM-assign load balance algorithm was proposed to allocate the incoming job requests in an efficient manner [14]. When compared to the active-VM load balance algorithm, the suggested algorithm solves the problem of inefficient resource utilization. A strategy for virtual link creation across VMs was presented to utilize the resource in a distributed cloud environment [15]. The suggested strategy reduced the number of virtual links, when compared to the physical links for resource optimization. It was concluded that the Steiner Tree Algorithm (STA) was suitable for large virtual networks, whereas Greedy Hub Selection (GHS) for small virtual networks. A SLA-based interoperable and self-adaptive approach were proposed for service provisioning in heterogeneous Cloud environments [16]. The proposed approach was a combination of negotiation, deployment and brokering of resources. When compared to other service provisioning approaches, the SLA-based approach was more reliable and effective.

A novel Memory Compression Based VM Migration Approach [17] was proposed to guarantee the service provided by the VM. Further, an adaptive zero-aware compression algorithm was proposed to reduce the VM cost, according to the memory page characteristics the results showed that the proposed approach reduced the downtime, migration time and transfer time. A multi-resource constraint VM placement solution was proposed to increase the performance of network in the cloud environment [18]. Further, the quadratic assignment problem was analyzed to minimize the communication traffic and to optimize the maximum link utilization in the network. The results showed that the hot links count was reduced by 37% and the link utilization was reduced by 20%. A VM migration strategy was implemented to optimize the CPU resources for load balancing [19]. The resources are exactly mapped to the suitable VMs of two types such as low-powered and high-powered VMs to reduce the execution time. The results showed that the migration of VMs improved the overall load balancing.

A paradigm, namely, prepartition [20], was proposed to migrate VMs for load balancing. The results showed that the proposed approach achieved 10-20% higher performance than the other methods. The proposed algorithm ensured a high level of adherence to the SLA and reduced the energy consumption, when compared
to the traditional algorithms. An adaptive provisioning technique was proposed to adapt the varying workload for achieving QoS in the dynamic environment [21]. A scheduling model was presented for the dynamic placement of VMs in a cost effective manner [22]. The price of the virtual machines was varied according to the number of user requests. The average prices and the cloud prices were considered for VM deployment. When compared to the other models, the proposed model deployed VMs with a reduced deployment cost. A Total Cost of Ownership (TCO) approach was presented for reducing the cost of resource allocation in cloud computing [23]. Further, a multi-method approach was applied to evaluate the cost related mathematical model. The results showed that the TCO supported better decision making in cloud computing. A Backward Speculative Placement (BSP) was presented for placing the VM in a virtualized cloud environment [24]. The results proved that the high demand satisfaction was achieved by the BSP technique.

2.3 LOAD BALANCING

The various load balancing algorithms were investigate to resolve the issues in cloud computing by assigning the client’s request to the available cloud nodes [25], [26]. The challenges for load balancing in cloud were spatial distribution of cloud nodes, storage replication, algorithm complexity, and single point of failure. The dynamic load balancing algorithms assigned the tasks based on the gathered and calculated attributes. The stochastic models [27] were proposed for load balancing and scheduling in cloud clusters. Further, a max weight algorithm was presented to allocate jobs to the servers based on the queue length information. The proposed model provided a good throughput and delay performance in the network. The Ant Colony Optimization (ACO) based load balancing techniques were surveyed for cloud computing [28]. This survey also studied the load balancing algorithms based on Non Polynomial (NP)-hard problems, ACO techniques, heuristic optimization etc. The results showed that the ACO algorithms effectively utilized the resources in the cloud environment. The heuristic based load balancing algorithms [29] in cloud computing were reviewed for optimal resource utilization. Further, an improved online load balancing algorithm was proposed to increase the efficiency. The results showed that the system performance was improved and achieved high user satisfaction.
The set of load balancing and scheduling algorithms were surveyed for cloud environment to increase the throughput, and to minimize the response time [30]. Various load balancing algorithms such as Load Balancing Max-Min (LB3MM), Min-Min (MM), Round Robin (RR), equally spread current execution, throttled balancing, connection mechanism, biased random sampling, and Honey Bee Foraging (HBF) were studied. A modified version of shortest job first algorithm was proposed for efficient utilization of resources [31]. The results showed that there was a decrease in average response time and increase in availability of VMs. A Distributed Virtual Environment (DVE) architecture was proposed by combining the cloud and the peer nodes [32]. The computational cost was reduced with the help of greedy heuristics for interaction between the users. When compared to the traditional methods, the suggested approach was more effective.

The different virtual machine load balancing techniques were reviewed along with its challenges [33]. The challenges in load balancing include spatial distribution of nodes, data replication, and algorithm complexity and point of failures. Thus, the study concluded with the solutions to the abovementioned challenges. An enhanced equally distributed load balancing algorithm was introduced for cloud computing [34]. The performance of the proposed algorithm was analyzed in terms of scalability, resource utilization, and response time and fault tolerance. The results showed that the response time was reduced, when compared to the existing algorithms. A profit aware load balancing was proposed for the data centers distributed in the cloud environment [35]. The suggested approach was better in terms of energy efficiency, optimization, and resource allocation. Further, the requests were dispatched in the order of the electricity prices.

The performance of several dynamic resource allocation techniques were surveyed [36] in this paper. This survey summarized the advantages and disadvantages of different dynamic resource allocation approaches. The various resource allocation strategies were linear scheduling strategy, parallel data processing and topology aware resource allocation. An Ant Colony Optimization (ACO) method was proposed to resolve the problem of load balancing in cloud environment [37]. The highly preferred routes by the other groups were avoided to increase the chance of distributing data traffic. The proposed heuristic based ACO algorithm distributed the service loads in an effective way. A cloud portioning based
load balancing model was introduced for the public cloud [38], which used a switch mechanism for choosing different strategies. To improve the efficiency of load balancing in the public cloud environment, the game theory concept was applied. When compared to other models, the cloud partitioning based load balancing model was more flexible and efficient.

An Autonomous Agent Based Load Balancing (A2LB) algorithm was proposed for dynamic load balancing in cloud environment [39]. The proposed algorithm comprised of load agent, channel agent and migration agent. The A2LB algorithm calculated the load accurately, which helped in the replacement of VMs, thereby, balancing the load. A Honey Bee Behavior Inspired Load Balancing (HBB-LB) algorithm was proposed to increase the throughput by balancing the loads of VMs [40]. The waiting time of each task in the queue was reduced and they were shared among other VMs in the cloud. A Hierarchical Topology-Aware Load Balancing Algorithm (HwToPoLB) was presented to provide an optimal solution for load balancing in cloud [41]. The combination of the machine topology model and the information regarding applications helped to improve the load balancing process. The results showed that the proposed load balancing algorithm achieved 19% improvement than the existing load balancing strategies.

A Task based System Load Balancing method was proposed using Particle Swarm Optimization (TBSLB-PSO), which transferred the tasks from the overloaded VM to other VMs [42]. Further PSO method was applied to move the extra tasks from the old VM to the new VM. The suggested approach eliminated the pre-copy process as there was no downtime. The Extended Divisible Load Theory (EDLT) was applied to enhance load balancing in cloud computing [43]. The EDLT approach decreased the time required for demand assignments and maximized the performance by eliminating the idle nodes. The results proved that the failure rate during simultaneous reporting was less than sequential reporting. A Greedy Particle Swarm Optimization (GPSO) was proposed to prevent the local optimum from reaching global optimum [44]. The configuration of the VMs was enhanced and the cost was minimized. The automatic calibration tool was used to improve the virtual design advisor.
A dynamic load balancing algorithm was proposed to improve the efficiency of cloud computing by avoiding the fault tolerance [45]. The exchange of status information introduced immense stress on the system during dynamic load balancing. An enhanced and secure approach was proposed for load balancing in cloud computing [46]. Four policies such as FCFS, RBAC, job mapping and virtualization were merged together for enhancement. This hybrid approach achieved high load balancing, when compared to the various existing techniques. A multi-criteria strategy based load balancing [47] and resource scheduling was proposed by combining three strategies such as min-min, max-min, and suffrage. The proposed algorithm was compared with FCFS and found that it reduced the response by utilizing the VMs with high Million of Instructions Per Second (MIPS).

A central load balancer based load balancing algorithm was presented in a large-scale cloud computing environment [48]. The proposed algorithm achieved minimum resource consumption, and scalability. The results showed that it also eliminated the bottlenecks and over provisioning of resources. The game theory [49] approach was applied to improve the efficiency of load balancing in the public cloud. The results showed that the overhead and the migration time were reduce, thereby increasing the performance. A hybrid meta-heuristic algorithm was proposed by integrating the ACO and PSO techniques to schedule the VMs for load balancing [50]. The Particle Swarm based Ant Colony Optimization (ACOPS) predicted the workload of the user request by applying the historical information. The proposed algorithm was compared with the methods that use fixed task sets and the results showed that the ACOPS maintained the load of dynamic environment in an efficient manner.

A Token based Heuristic Algorithm (THA) and genetic algorithm based integrated approach were proposed for load balancing [51]. The token routing moved the tokens around the system to reduce the system cost but it lacked scalability. The processing time was reduced and the life span was improved by extracting the advantages of THA and GA in the integrated approach. A firefly algorithm [52] was proposed for load balancing in cloud to maximize the resource utilization. According to the factors such as processing time, memory usage and access rate, the load model for every resource was derived. To overcome the exploration problems in GA, the firefly algorithm was introduced. The results
showed that the average time required for load balancing using firefly algorithm was 0.934 ms. Three load balancing approaches such as biased random sampling, active clustering, and HBF behavior were analyzed [53]. The advantages of these approaches that included the fault tolerance, availability and scalability were identified. The results showed that these algorithms achieved flexibility and high customer satisfaction.

The max-min task scheduling algorithm was applied to balance the load of the servers in the cloud [54], [55]. The max-min task scheduling distributed the local workloads evenly among all the nodes. A Genetic Algorithm (GA) based load balancing strategy was proposed to balance the load of the cloud infrastructure [56], [57]. The GA was composed of three operations such as selection, genetic operation and replacement. The results of the GA based load balancing strategy outperformed the existing approaches such as FCFS, RR and SHC. An Energy-Aware Multi Objective Chiropteran Algorithm (EAMOCA) was proposed for load balancing in a hybrid cloud computing [58]. The proposed EAMOCA integrated the properties of echo-localization and hibernation and it attained maximum utilization with low execution time and energy.

A Datacenter Energy-efficient Network-aware Scheduling (DENS) was introduced to balance the load of the cloud datacenters [59]. The amount of energy, job performance and traffic demands were balanced by the DENS methodology. The proposed approach achieved the tradeoff between job consolidation and traffic patterns. A Resource Intensity Aware Load (RIAL) balancing method was proposed to achieve load balancing in cloud environment [60]. According to the usage of bandwidth, the RIAL allocated VMs to minimize the performance degradation. The results proved that the time and cost was reduced and the load imbalance was eliminated. The Cloud Light Weight (CLW) model was introduced to balance the workload of the VMs in the cloud datacenters [61]. The CLW was simulated using the simulator, namely, Cloudsim and the results showed that the CLW assured QoS to the cloud users.

2.4 RESOURCE SHARING

An agent-based cloud resource scheduler was proposed to share resources in an inter cloud environment [62]. Information dissemination and match
making were combined for resource sharing. The dissemination overhead was reduced up to 19% with the help of proxies. A Collaborative Cloud Computing (CCC), namely, harmony for resource sharing was proposed in an efficient and a trustworthy way [63]. The issues in the resource and reputation management schemes were resolved to ensure the deployment of CCC with entities having autonomous features. The results proved that the issues of the existing approaches were solved and a trustworthy sharing was enabled.

A dynamic resource sharing method was proposed for cooperative sharing in mobile cloud environment [64]. The proposed method included four major components such as mobile resource monitor, job handler, resource handler and results consolidator. The results of the proposed method proved that it was feasible in terms of computational power. An exam resource sharing mechanism was presented by applying multi-person prisoner’s dilemma technique [65]. The game theoretic approach was applied in the multi-prisoner’s dilemma along with an incentive mechanism to encourage the providers to share multiple resources in cloud. The proposed approach ensured maximal system and resource utilization. A greedy approximation algorithm was applied to solve the problem of Multi-resource Sharing-Aware Virtual Machine Maximization (MSAVMM) [66]. The suggested approach required a set of VMs to be initialized at server to derive the profit. It also enabled the sharing of memory pages among VMs to maximize the performance and profit.

A hybrid load balancing strategy was proposed by integrating the genetic algorithm and the concepts of gravitational emulation to provide collaborative services [67]. Further, the suggested approach ensured QoS, interoperability, reliability, resource provisioning and scalability. The results proved that the suggested hybrid approach outperformed the abovementioned algorithms. An energy efficient resource management system was proposed to reduce the operational costs in the virtualized data centers [68]. The VMs were consolidated based on the utilization scale and topology to save the energy. The VMs were dynamically reallocated based on the CPU requirements. The results proved that the proposed scheme was more reliable and energy saving.
2.5 RESOURCE ALLOCATION

The skewness algorithm [69] was presented for dynamic resource allocation by determining the unevenness in the multi-dimensional resource utilization of server. A working set prober was implemented on each hypervisor to estimate the working set size of VMs running on it. The proposed skewness algorithm dynamically allocated the data center resources and supported green computing by server optimization. A learning automation based proactive resource allocation approach, was presented to decrease the impact of SLA violations [70].

The pairing problem was defined using two games such as User Brokers (UB) game and VM Brokers (VB) game. The results proved that the proposed approach was highly applicable in service oriented environments like cloud. A tenant-based resource allocation model was designed to allocate the software resources in the cloud infrastructures [71]. The over and under provisioning of resources were solved by the designed approach. The suggested resource allocation model achieved high scalability and cost-effectiveness.

A heterogeneity aware architecture [72] was proposed for resource allocation and scheduling in cloud environment. The problem of resource allocation and scheduling in the heterogeneous cloud clusters were resolved by the proposed architecture. The results proved that the suggested architecture achieved a good fairness and high performance. A class of resource allocation algorithms were proposed in the shared hosting platforms to optimize the resource allocation [73]. The resource allocation problem was formulated to increase the yield of services. The results showed that the suggested algorithms outperformed the greedy algorithm and allocated the resources in an efficient way. A dynamic planning based scheduling algorithm was proposed using resource allocation policies such as immediate, best effort, advanced reservation and deadline sensitive [74]. To evaluate the performance, the policies based dynamic scheduling algorithm was compared against the existing algorithms of Haizea. The metrics such as system utilization, number of leases accepted and number of leases rejected were used for evaluation.

The proposed algorithm increased the number of accepted leases by applying swapping and multiple slots concept. A priority based resource allocation algorithm [75] was proposed to allocate resources with minimum wastage and
maximum profit. Further, the priority based algorithm satisfied both the user’s requirement and server’s performance by allocating resources in an efficient way. The results showed that the priority based algorithm was better than the grid and utility based resource allocation. A SLA driven resource allocation scheme was proposed to support flexible SLA establishment and to manage cloud resources for preventing the violations of SLA [76]. The proposed framework consisted of a service broker and a cloud service provider, where the broker was used to connect a consumer to the provider. The results showed that the SLA-driven resource allocation scheme outperformed other schemes in terms of SLA violations and profits.

An Intelligent Economic approach was proposed for Dynamic Resource Allocation (IEDRA) to allocate resource to a large number of consumers and providers [77]. A price formation mechanism was proposed to improve combinatorial double auction protocol for reasonable bidding and asking. The price formation mechanism utilized a back propagation neural network based price prediction and price matching algorithm. Further, a reputation system was introduced to detect the malicious users in the cloud market. The paddy field algorithm was applied to solve the winner determination problem. The suggested approach was economically efficient and more trustful, when compare to the existing approaches. A Semi-Markov Decision Process (SMDP) based resource allocation was proposed for vehicular cloud computing [78]. The iteration algorithm described the action to be occurred at a particular situation. The resource allocation scheme based on optimal computation maximized the reward expected for a long-term. The results proved that the SMDP based approach gained a better performance with minimum complexity.

A preference based resource allocation technique according to the demand of the users [79]. Further, a market driven auction mechanism was designed for the user identification on the basis of their payment abilities. When compare to the off-line VCG auction mechanism, the proposed system resulted in a high performance and achieved more profit to the service providers. The agreement protocols based adaptive resource allocation was proposed to achieve an adaptive fault tolerant system [80]. The agreement protocol allocated the resources only when they were needed to reduce the cost. When switching between two service providers,
the agreement protocol based approach was more cost effective and efficient. A fuzzy-logic based trust and reputation model [81] was built to allocate resource in a secure and safe way.

A network aware resource allocation algorithm was developed to minimize the maximum distance and latency between the selected data centers [82]. To minimize the inter datacenter traffic, the data-center assignment for each individual VM was determined. The goal of the machine selection was to find the machines that reduce the inter-rack communication and avoid long paths for the communication. When compared to the existing systems, the proposed system allocated the resources in an effective way. A Position Balanced Parallel Particle Swarm Optimization (PB-PPSO) [83] method was introduced to allocate the resources with high profit and achieve high user satisfaction level in the cloud computing environment. The performance metrics such as total profit, average response time, and number of VMs were considered for evaluation. The results showed that the PB-PPSO method achieved high profit and small response time with a less number of VMs.

The algorithms such as Joint-The-Shortest-Queue (JSQ) routing and power-of-two-choices routing algorithms with MaxWeight scheduling algorithm [84] were applied to allocate resources in the heavy traffic limit. Load balancing was done using the diffusion limit arguments and a steady state version was presented using Lyapunov drift arguments. When compared to the traditional schemes, the throughput of JSQ, power-of-two-choices routing and MaxWeight was found to be optimal. A hierarchical resource allocation scheme was proposed to allocate resources in a virtualized cloud environment [85]. According to the frequency, capacity and load, a greedy algorithm was applied to assign the tiers to the servers. The capacity allocation and load balancing was improved using a fixed-point iteration. An intuitionistic fuzzy set based decision making approach was proposed for multi-attribute systems [86]. A score function and an accuracy function based ranking method were proposed to score the attributes. The degree of membership, non-membership and hesitation were considered in the suggested approach. The results showed that the time and cost was reduced by the hierarchical allocation method.
A game theory based resource allocation was proposed to reduce resource wastage in cloud computing [87]. The tradeoff between the fairness and utilization was maintained using the backward induction approach. The proposed algorithm was compared with the Hadoop scheduler and found that the game theory based resource allocation achieved better performance. A prediction-based dynamic resource allocation algorithm was presented for delivering video transcoding service [88]. The resource allocation was performed in a proactive manner with the help of a two-step load prediction method. The results were demonstrated in a discrete-event simulation, and proved that it achieved higher prediction accuracy. An optimal task scheduling and resource allocation was proposed using Particle Swarm Optimization (PSO) based fitness function [89]. To balance the load the PSO based fitness function was applied to reduce the make span and to maximize the processing capacity. The results showed that the PSO based method resulted in less execution time and cost.

The resource allocation in cloud was automated by the application of reinforcement learning [90]. The temporal difference reinforcement learning algorithm, namely, Q-learning was applied to determine the optimal scaling policies. A novel parallel Q-learning approach was used to minimize the time to find out the optimal policies. Markov decision process was one of the decision theoretic frameworks, which was utilized to make decision for resource allocation. A cross layer model was proposed to allocate resources in Cloud Radio Access Network (C-RAN) [91]. The cross layer model utilized the mixed-integer nonlinear programming, Remote Radio Heads (RRHs) selection, elastic service scaling and joint beam forming. The cross layer approach minimized the energy consumption at the RRHs and the fiber links of C-RAN. Multiple attributes based dynamic resource allocation were proposed in collaborative cloud computing [92]. The suggested methods combined the machine learning and multi attribute tuning to ensure QoS. The reputation based algorithms were used to ensure the trustworthiness, to reduce the complexity and to prevent node overhead.

A resource allocation framework was proposed based on demand prediction in multi-tenant cloud environment [93]. The suggested framework classified the tenant based on the requirement of the resources. A best-fit heuristic approach was used to match the service tenants with the VMs. The results proved
that the best-fit heuristic approach utilized the overall capacity of the VMs. An auction based resource allocation algorithm in cloud was presented for market scenarios [94]. The dynamic pricing was computed to generate the revenue and to reduce the price. A server consolidation based resource allocation was proposed to schedule the jobs in an energy efficient manner [95]. The number of active servers was minimized to reduce the energy consumption.

A max-heap was constructed in such a way that the cluster with highest rank as the root and the clusters with minimum rank were placed at the leaf. The results showed that the proposed approach achieved high energy conservation. A Reciprocal Resource Fairness (RRF) [96] was proposed to enable fair and fine-grained resource sharing in multi-tenant IaaS cloud. Two hierarchical and complementary mechanisms such as inter tenant resource trading and intra tenant weight adjustment algorithms were implemented. The fairness was achieved by improving the density of VM and revenue of cloud providers. The results were simulated in the Fortan to C (F2C) Xen platform, and it guaranteed 95% economic efficiency. A bin packing technique was proposed for dynamic resource allocation in cloud [97]. For allocating the physical machines or cloud resources the drip based resource allocation technique, which used the bin packing technique was introduced. The results showed that a huge number of requests were handled only by using a few number of physical machines.

An agent based best fit resource allocation scheme was proposed to increase the resource utilization [98]. The results showed that the best fit approach was efficient in terms of job execution time, cost, VM allocation and resource utilization. A continuous resource allocation strategy was presented to optimize the scheduling process in cloud [99]. The suggested resource allocation mechanism adopted minimal domination matching to compensate the tradeoff space. The probability of migration was reduced using the resource reserve scheme. The proposed approach resulted in a great flexibility and low computation overhead. A priority based resource allocation algorithm was proposed by switching between the lower and higher priority tasks [100]. According to the SLA and priority, the suitable VMs were selected for providing the requested service. The provision engine was managed by the virtualization layer, which interacted with the physical resources through the provision engine.
A Markov Chain model [101] based monitoring technique was proposed to predict the state of the resources available in the cloud. As the mobile devices were not reliable and had no guarantee, it was difficult to utilize a mobile device as a resource. The proposed technique eliminated the volatility and accurately predicted the future of the resources. Two pricing schemes such as on-demand and reserved pricing mechanisms were investigated to allocate the resources in a monopoly market [102]. Here, the backward method was applied, which initially analyzed the cloud users’ decision. The results summarized various cloud users and VM instances, by analyzing the revenue of cloud providers. A game theoretic framework was proposed to solve the problem of SLA-based resource provisioning and management [103]. The results showed that the proposed approach achieved existence and uniqueness of Nash equilibrium.

2.6 PARALLEL JOB SCHEDULING

A Berger model based scheduling algorithm was designed to establish the dual fairness constraint in virtualized cloud environment [104]. The fairness of resource allocation was judged by the application of justice function. The results showed that the user tasks and the fairness were effectively executed. A Biogeography-Based Optimization (BBO) was proposed to solve the binary integer problem in job scheduling through better solution adaptation strategy [105]. In BBO, the GA and ACO strategies were incorporated to generate a new set of solutions, at each iteration; the Mann-Whitney test was performed to evaluate the performance results of the BBO algorithm. The results proved that the performance of BBO was better than the GA and PSO algorithms. An Improved Genetic Algorithm (IGA) was proposed for job scheduling by speeding up the process of GA [106]. The proposed model had five components such as preprocessing unit, job scheduler, users, data center, and data center manager. The preprocessing unit encoded the attributes into Users Job Attribute Vector (UJAV), which included Expected Instruction Count (EIC), job deadline and delay cost.

A mathematical model proposed for resource provisioning in heterogeneous cloud environment [107]. The data migration in cloud caused the performance degradation, which was resolved by the mathematical model. When compared to the traditional methods, the mathematical model achieved optimal
performance and better response time. A clustering based parallel task scheduling algorithm was proposed in cloud computing environment [108]. The suggested scheduling utilized fuzzy clustering approach to match the tasks and the resources in a concurrent manner. The proposed scheduling algorithm outperformed other algorithms in terms of efficiency. When compared to the conventional algorithms, the IGA provided higher user satisfaction in terms of QoS. A scheduling model was described to reduce the risk of SLA violations in cloud infrastructure [109]. Further, a data distribution monitoring approach was proposed using semantic metadata annotations. The results of the proposed scheduling approach outperformed the traditional models.

Various priority based job scheduling techniques [110], [111] were reviewed in cloud computing. The credibility based resource provisioning was proposed to achieve credibility with high efficiency. The job scheduling methods such as law of failure based resource allocation strategy, credibility based resource assignment, ant colony optimization, threshold based scheduling algorithm and dual fitness based optimized genetic algorithm were summarized for improving the performance. A Multi-Objective Genetic Algorithm (MO-GA) was proposed based job scheduling model for cloud computing [112]. The suggested model also focused on selection operators, crossover operators, sorting praeor solutions and encoding rules. When compared to other algorithms the MO-GA based algorithm attained 5.73% profit and consumed 44.46% energy. A job scheduling algorithm was presented for Wireless Sensor Network (WSN) based cloud computing [113]. This study resulted in the generation of two algorithms such as Priority-Based Two Phase Min-Min (PTMM) algorithm and Priority-Based Two Phase Max-Min (PTAM) algorithm. The proposed algorithms were compared with the min-min and max-min algorithms and proved that the PTMM and PTAM achieved shorter completion time.

A double level priority based task scheduling was proposed for preserving energy in the cloud environment [114]. The energy related challenges were addressed by the suggested algorithm based on a hierarchical scheduler. When compared with sequential scheduling, the resource and task selection achieved more efficient results. A decentralized multi-agent approach was proposed to schedule jobs in the cloud environment [115]. The Pareto dominance concept was implemented at the client level for the construction of global scheduling solution.
The results showed that the decentralized approach minimized the probability of execution failure and completion time by allocating minimum number of resources. A dynamic resource allocation and prediction system was designed using VMs in the cloud environment [116]. The overhead was reduced by combining the cloud technologies and the resource scheduling strategies. The performance of the system was enhanced by dynamically changing the status information of the resources.

A genetic algorithm based efficient job scheduling was proposed in expert cloud platform [117], [118]. The proposed method was compared with FCFS, shortest process next and highest response ratio next algorithms in terms of execution time, service wait time, human resource utilization and failure rate. The results proved the efficiency of the genetic algorithm based scheduling method by providing high accuracy and performance. A deadline aware two stage scheduling algorithm [119] was proposed to schedule VMs in the datacenter for the user requests. The results proved that the deadline aware two stage algorithm reduced the waiting time and response time by a constant factor.

A hybrid approach, namely, FUzzy GEnetic (FUGE) was presented by integrating the fuzzy theory and genetic algorithm to schedule the jobs in an optimized way [120]. The results proved that the execution time and cost was reduced and the performance was improved. A differentiated policy based job scheduling using queue model was proposed in a private cloud environment [121]. Further, a back filling based advanced reservation technique was presented to serve the users request. The QoS was achieved by the efficient provisioning of the resources. The improved particle swarm optimization algorithm [122] was proposed for resource scheduling in cloud computing. The merits of PSO included the following: scalability, strong robustness, parallel distribution and high flexibility. The characteristics of PSO and Simulated Annealing were mixed to devise the mixed scheduling algorithm. Improved PSO algorithm guaranteed accuracy and improved the convergence rate. When compared to the traditional systems, the Improved PSO algorithm was efficient in terms of speed and optimization. An enhance max-min task scheduling algorithm [123] was introduced in cloud computing. The results showed that the proposed algorithm improved the efficiency with limited usage of resources.
A deadline-constrained task scheduling applications were proposed in a cost-efficient manner [124]. The hybrid clouds were overloaded with huge workloads, hence application and data requirements were taken into account. The performance of the proposed algorithm was evaluated in terms of deadline, computational efficiency, investment errors and cost saving. A priority based method was proposed for the consolidation of workloads in parallel [125]. The results of the suggested approach showed that it outperformed the traditional scheduling algorithms. A fault aware honey bee scheduling algorithm was proposed to maintain the QoS to the cloud user [126]. The results showed that the proposed algorithm was efficient in terms of failure reduction, space utilization and processing power. A dynamic optimization algorithm was proposed to schedule the tasks in cloud computing [127]. The deadline constraints were considered to achieve the overall performance by minimizing the turnaround tasks. In case of profit based priority, the cost based greedy scheduling algorithm was used. The results showed that the number of tasks completed in a unit time was increased.

An Energy Efficient Scheduling with Traffic Load Balancing (e-STAB) was presented to allocate jobs in an efficient manner by reducing the communication delays and congestion related packet losses [128]. Further, the suggested algorithm optimized the energy consumption of the data center by providing load balancing of traffic flows. The results of the e-STAB scheduler outperformed the green scheduler in terms of execution time, network load and QoS. A new bee swarm optimization algorithm, namely, Bees Life Algorithm (BLA) [129] was presented to schedule the jobs among processing resources onto the cloud data centers. A two-point crossover method was used to replace the queen by the drone bee. The results proved that, the BLA outperformed the conventional approaches by scheduling the jobs in an efficient manner. A context aware job scheduling approach [130] was proposed by considering changes in user’s context to improve the QoS and resource utilization in cloud. Further, the suggested approach provided delay-tolerant job execution required in mobile environments. Based on the policy definitions the jobs were scheduled in the proposed model. The results showed that the context aware job scheduling approach was efficient, when compared to the traditional scheduling approaches.
A cost-efficient scheduling was proposed for dead line constraint resources in hybrid cloud [131]. The utilization of internal infrastructure and the cost of the outsourced tasks were minimized using the suggested approach. The results showed that the sensitivity and accuracy was higher than the existing approaches. A hybrid job scheduling algorithm [132] was proposed by combining genetic algorithm and fuzzy theory to assign the jobs to the VMs. The genetic algorithm was modified to balance the load and to reduce the execution time and cost. A dynamic quantum time based Tri Queue job Scheduling (TQS) algorithm [133] was proposed for cloud environment in an efficient manner. The TQS algorithm allocated jobs in a dynamic quantum time based round robin fashion. The results showed that the TQS algorithm achieved the optimized solution by using the resources in an efficient manner.

2.7 SUMMARY

Cloud computing is novel technique that plays a significant role in the day-to-day applications, where complex computing is required. The cloud providers offered services to the cloud users in flexible, scalable, reliable and secure manner. To achieve a QoS assured and high performance computing, resource utilization was an important part in the cloud. The cloud environment consisted of a large number of VMs in the datacenters, which were used to service the requests of the users. The resource optimization was achieved through the proper selection of VMs. Different VMs selection and migration techniques were discussed for appropriate choice of VM. Various load balancing mechanisms and algorithms were surveyed to improve the lifespan of the VMs. The resource sharing and resource allocation were necessary to balance the load of the VMs. The underutilized VMs were provided with tasks from the over utilized VMs. Existing algorithms for resource allocation and sharing were studied to identify and resolve their issues. According to the allocated resources, the conventional job scheduling approaches were discussed to schedule the jobs in the VMs.