



Mutual Profit Increment for Cloud User and Resource Providers Efficient Resource Utilization Auction Method in Cloud Computing

Madhina Banu Dawood Ali^{1*}

Aranganathan Somasundaram¹

¹*B.S.A Crescent Institute of Science and Technology, Chennai, Tamilnadu, India*

*Corresponding author's Email: madhinabanu.phd@gmail.com

Abstract: Nowadays, cloud computing becomes emerging technology, where any kind of data can be shared. Software and resources can be offered based on the requirement of customers. Effective resource allocation is a challengeable task in a Cloud environment. Cloud resource providers have sufficient resources which should be efficiently allocated. Otherwise, a problem will occur in underutilization devices. The main difficulty is to identify an appropriate cloud resource provider for each cloud user who could sell various resources in minimum possible cost according to their requirements and simultaneously find suitable cloud user for a cloud service provider who values his resources at the most. To overcome the above issues, Efficient Resource Utilization Auction Method is designed to solve resource allocation issues by using dynamic cost model and effective resource utilization for cost resources. The method offers one type of safe auction process between resource provider and user to grow their business mutually in win-win condition. Where, cloud user looks to get actual requirement with an affordable cost; simultaneously cloud resource provider also looks to generate good revenue and meaningful profit. The method fascinates the cloud resource provider to decide the resource allocation for the submitted virtual machine (VM) to the physical machine (PMs). The proposed method works based on a dynamic economic auction model. Based on the experimental evaluation, proposed Efficient Resource Utilization Auction Method improves the 18% resource utilization (RU), and reduces 20 seconds task completion time (TCT) compare than conventional methodologies.

Keywords: Cloud auction, Resource utilization, Winner declaration, Bidding process, Resource provider, Cloud user, Task completion time, Efficient resource utilization auction method.

1. Introduction

Cloud computing is Internet-based computing where any kind of information can be shared. Software and resources can be provided based on customer/client demand devices. The surfacing of the cloud has significantly distorted the global insight of infrastructure, software delivery services and development models. Optimal resource allocation is one of the biggest challenges in the Cloud environment. Cloud providers have resources which should be efficiently allocated; otherwise, the problem will occur underutilization. Moreover, multiple providers in the cloud market want to sell resources. Each service provider wants to sell resources to users and earns maximum revenue. It is difficult for users to choose the best resource among

cloud providers. It is also challenging for providers to choose the user whose resources values are more. So, the problem is to determine an appropriate cloud provider for each user who could sell various resources in minimum possible amount according to their requirements and simultaneously find a suitable customer for a cloud service provider who values his resources at the most. The model should satisfy all the requirements of users at a given point of time. It should also impose some minimum restrictions on the cloud providers to fulfil requirements that violate the conditions.

Many market-based models for resource allocation have been introduced in the literature. But, there exists many gaps in the literature which are explained here. Most of the techniques are biased towards the provider. However, none of the methods works on behalf of clouds user and as well as the

cloud market. Current resource allocation works based fixed pricing model. Resource monitoring is utilized for reservation of resources. However, user requirement is dynamic which cannot be controlled by a static pricing model. Most of the cloud resource provider companies prefer fixed price based model for quoting of their resources. But, the existing model is not compatible with every scenario of the cloud market. Dynamic Pricing-Based Allocation Mechanism (DPAM) introduced for the auction-based cloud market, to deploy the workflows with minimal task completion time and cost. The motto of the technique is to offer the best price to cloud users to gain the highest revenue. However, the method failed to optimize resource utilization (RU). Biobjective Scheduling Strategy (BOSS) worked based on a reverse auction for resource allocation for tasks of workflows and every task proceed with bidding operation to reduce the price cost and execution time. However, the methods follow that fixed cost economic model during bidding. So, some cloud resource providers have dull competition, and they can lose the bidding almost time. The method has very less resource utilization which increases task completion time. This way cloud services provider resources are not appropriately optimized. Here, procurement of resources is performed through the manual process. It is difficult for a cloud user to check their requirement with all cloud resource provider individually. Most of the existing work completed with single resource providers who fulfil the requirements of users. Users often required their resources in bundles. Where, existing auction-based method does not consider minimal restriction for cloud resource provider to satisfy the user requirements. In the scenario, Cloud providers may fraud the cloud user for generating more revenue by offers poor quality of services.

To offer the best solution for the above problems, Efficient Resource Utilization Auction Method proposes to offer efficient resource utilization with a dynamic cost model. The proposed method creates a win-win condition for cloud user and as well as cloud resource providers. Each physical machine (PM) has a local manager and local managers manage resources. Then, the proposed method assists to a local manager for deciding which PMs should associate with which VMs for optimal resource allocation and job execution in minimum time and cost. The proposed method works based on a dynamic economic auction model. The dynamic pricing model is presented in the algorithm for pricing of resources. Where, cloud user looks to get actual requirement with a reasonable cost; simultaneously cloud resource provider also works

to increase the sale of their resource for meaningful profit. Generally, cloud user looks forward to seeing their resources and resource provider filling the bids to offers the services. The proposed algorithm is implemented using cloudsims which proves the efficiency of the algorithm to increase the revenue of the Cloud provider. The paper contributions are followed as:

- To analyze cloud user requested resource need and simultaneously find resource allocation optimal mechanism for efficient resource allocation of resources.
- To offer a dynamic cost economic model for making a win-win condition for cloud user and cloud resource providers.
- To connect actual requested resource for cloud user with affordable cost
- To develop an efficient method optimal resource utilization to minimize task execution time and cost.
- To evaluate the performance, the proposed algorithm with existing methodology in terms of task completion time and resource utilization.

The rest of paper is organized as Section 2 briefs the literature review with closest methodology features and function details. Section 3 explains the proposed method processing steps, algorithm details with architecture diagram. Section 4 discusses deployment setup, input parameters and simulation result. Section 5 concludes overall work with future work details.

2. Related work

In [1] combined two existing combinatorial auction-based allocation mechanisms that solved the problem that occurred in allocating the VM instances in clouds. The first mechanism considered a problem of combinatorial auction. Where, it allowed the users to request more than one item of the same type. The second mechanism considered several types of items, where each type of items has a few instances. In [2] developed a double-sided Truthful Multi-Unit Double Auction mechanism (TMDA) to understand researchers to design a trustful double auction mechanism involving users and providers. In [3] developed a mathematical model that analyzed cloud resource allocation problem. It also included the trust evaluation mechanism and combined genetic algorithm along with a simulated annealing technique. In [4] developed a web application model that captured the behavioural patterns among different profiles of users. The model provided a support analysis and

simulated resource utilization in the cloud. These methods performed well on reliable resource allocation. However, they failed to reduce job execution time and as well as cost.

In [5] developed an energy-aware pricing model that investigated the different pricing models which influenced energy consumption, performance and cost in cloud services. The developed model charged the consumer based measurement of actual resource usage per unit like CPU, RAM, Network, and Disk that also included energy consumption. In [6] developed a pricing scheme that addressed geographically distributed green data centres which was considered a multi-leader, single follower game specific. Where, it has been associated as grid and cloud respectively. It considered two key suspects first with hybrid green generation capabilities, the second smart grid that served different data centres in the cloud. In [7] expressed SoftBW that enabled pricing bandwidth which guaranteed bandwidth. It considered the low network utilization related to cloud-scale data centres. The SOftBW leveraged usage-based is charging guaranteed price-performance consistency within the tenants. It implemented a fulfilment based scheduling that provided guarantee on bandwidth as well as fairness. In [8] described an adaptive strategy that customized the provider's business objective. It depicted on the market of resources to provide a balance of demand and supply in auction-based sales. It stated that the model helped in maximizing the revenue with the context of procurement auctions. Existing methods are discussed about the price of resource bidding and price calculation based on resources. But, the current method failed to discuss client requirement matching with exact, reliable resources of source providers.

In [9] addressed an adaptation of Consensus Revenue Estimate auction mechanism that settled the multi-unit online auction on cloud resources. It combined the auction design and oriented scheme for dynamically calculating reserve prices related to data centered power usage effectiveness (PUE) and appropriate cost of electricity. In [10] investigated the meta-heuristics resource allocation techniques that maximized the gains and minimized the expenses in the financial aspect of the cloud users in IaaS cloud platform. In [11] developed a market model called CDARA (combinatorial double auction resource allocation) for resource allocations in cloud computing environments. In [12] developed the combinatorial double auction-based market, where the broker allocated the providers' to VMs based on the users' requests. The model formulated as an integer linear programming model that

maximized the profit for both users and well service providers. Conventional methods expressed their view about the double auction model for resource allocation and cloud user request handling in the cloud. However, these methods are not capable of reducing job execution time and resources cost.

In [13] addressed the multiple resource procurement from various cloud vendors who participated in the bidding which was accomplished with the assignment of dynamic pricing of resources. The work pre-processed the user requests, analyzed the corresponding auction and declared a set of vendors that are bidding towards the auction as winners using combinatorial auction branch bids (CABOB) model. In [14] recommended an effective pricing strategy for the reverse auction. It expressed an efficient resource allocation mechanism with support of dynamic pricing reverse auction. It stated that the auction process, providers changed the resource price based on the trade situations. In [15] described a fair multi-attributed combinatorial double auction that addressed the allocation of cloud resources process, the work also considered the quality of service parameters apart from pricing strategy. The developed model was the extension of two models namely combinatorial double auction that addressed cloud resource allocation and a reverse auction with multidimensional fairness for allocation of the resources. In [16] utilized a multi-objective particle swarm optimization technique based on the concept of Crowding Distance (MOPSO-CD) to address the optimization problem. It attained the major critical requirements; those maximized the users and providers revenue and found the optimal solution at the desired specific time. The existing methods described the resource allocation optimization mechanisms and cloud requirement solution with reliable service resource providers. But, these methods performed weakly in terms of cloud user task allocation and price evaluation for services.

In [17] expressed an approach of cloud resource procurement that automated the selection towards the appropriate cloud vendor along with implemented dynamic pricing. It suggested the three possible mechanisms towards cloud resource procurement: cloud-Bayesian incentive compatible (C-BIC), cloud-dominant strategy incentive compatible (C-DSIC), and cloud optimal (C-OPT). In [18] suggested specific schedule virtual machines (VMs) that were driven by the fact of range-based non-linear reductions towards the utility. These are specific to every user and across several ranges in the allocation of resources like a partial utility. The work also defined an inclusive cost model that

incorporated partial utility provided by clients towards a certain level of degradation during the overcommitted moment of VMs. In [19] elaborated an adaptive VM resource scheduling algorithm with auction mechanism that also considered multiple factors like auction deadline, network bandwidth etc. In [20] described a procurement market to achieve the computation of resources for making the right actions. It explained resource overbooking mechanism that overcame the problem of resource underutilization. It discussed related to the use of alternative auction-based mechanisms that sold the “residual” computing capacity occurred in it. Where, the capacity by the provider would not be able to allocate by direct-selling. Existing algorithm explained about resource procurement with market analysis, VM scheduling for resource utilization. However, the methods failed to bring dynamic cost model which offers win-win conditions for cloud user and service providers.

3. System methodology

The proposed algorithm “Efficient Resource Utilization Auction Method” in Cloud is introduced to solve the problem of resource allocation using auction based resource allocation technique by using a dynamic cost model for the pricing of resources. The proposed architecture consists of multiple cloud resource providers and multiple cloud users who want resources to complete their task. Where, resource providers want to increase their sale of a

resource. Figure (refer with: Fig. 1) explains the system architecture of the proposed algorithm with processing steps.

3.1 Cloud user

It is the entity who wants resources to fulfil its requirement for completing the job. It will go to the auctioneer for raising resource request in which will mention the total number of required resources for each type, time for resource requirement and their budget. When, cloud user is assigned specific cloud resource provider, and then cloud user will get a notification. When, requirements are fulfilled then the user needs to pay the claimed amount by the auctioneer.

3.2 Auctioneer

The module plays a vital role to complete the auction process. It is responsible for assigning specific resource provider to cloud user. The auctioneer will fix a price that claimed the cloud user should pay the amount. Cloud user will raise their requirement request to auctioneer and auctioneer will transmit the messages to Cloud resource providers for the bidding process. Auctioneer is responsible for declaring auction winners and alerting message to the cloud resource providers.

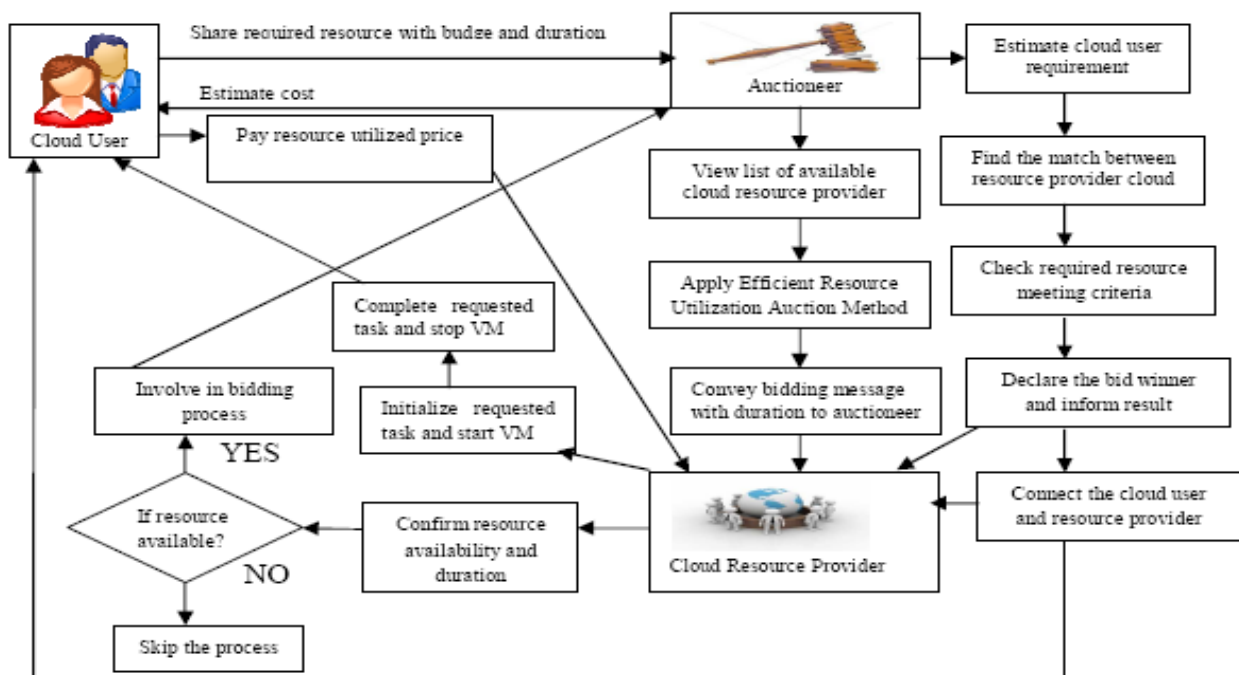


Figure.1 System architecture of the proposed algorithm

3.3 Cloud resource provider

The module contains resources availability like Memory, CPU, processor, virtual machine, bandwidth, disk storage. Cloud resource provider is authorized to sell the resources for gaining maximum revenue based on the resource request of cloud user. The cloud resource provider will engage in the auction process; when, auctioneer conveys the messages. If, a service provider has sufficient resources then they can participate in the bidding process. Once, the resource providers win the auction then it is liable to offers services to specific cloud users and gets profit. Cloud resource provider can join nth number of auction processes based on their resource availability. However, if, cloud resource provider is a winner of the bidding process then resource provider is responsible for offering the services.

3.4 Efficient resource utilization auction method

Efficient Resource Utilization Auction Method is a safe auction process between resource provider and user to grow their business mutually in win-win condition. Where, cloud user looks to get actual requirement with the affordable cost; simultaneously cloud resource provider also looks to generate good revenue with meaningful profit. Generally, clouds look resources and resource provider filling the bids to offers the services. Where, auction winner is only able to offer the services to the user. Initially, cloud user submits their requested resources to the auctioneer. Where, the auctioneer has various types of cloud resource provider who have different kinds of resources. Once, auctioneer receives the request from cloud user then, it will convey message among resource provider for the bidding process. Only, that resource provider can involve in the bidding process. If service providers have sufficient resource for bidding, then they can proceed with overbooking. During bidding time, the resource provider's resources are busy with another user. However, during the service period, the resource will be surely available then resource provider can involve in the current auction process. After a completion bidding process, the method will start to map the cloud user and resource provider who is ready to work with the lowest bid and high resource availability. The proposed method has various resource optimizations based policies to a cost-effective framework with good reliability and stability.

Here, cloud resource providers initially register with an auctioneer to get bid from an auctioneer in future. Cloud user share their required resource and

duration details with the auctioneer. Hence, the auctioneer will display the price and time details for requested resources. The auctioneer collects the user requirement and conveys the message with time duration to all registered cloud resource providers for the bidding process. Cloud resource providers will check their available resource and time duration, after receiving the message from the auctioneer. If, resource provider has sufficient resources then it will proceed for bid booking and advertise their resource availability. The auctioneer will close the bidding operation after completion of duration.

Hence, the system will not allow anybody to involve in the bidding process. Hence, the auctioneer will estimate the overall process; like an evaluation of total resource requirements as per cloud, user need. The method will find the required resource bidding value. The system gives higher priority to those cloud users who look resources for a long duration. The system calculates the resource value of all bidding booked cloud resource provider. The cloud resource provider list will be filtered with the maximum value of services with minimum bidding to find best resource provider. Cloud user mapping will be done with cloud resource providers. When, cloud user requested resource criteria meet with resource provider available resource. Otherwise, it will iterate to check another resource provider to find actual resources. This process until continue; till, cloud user is not satisfied with available resources. After fulfilling the first cloud user requested resource, the resource provider can try next cloud user. If, resource provider has sufficient resource to offers service to next cloud user. Once, user-requested resources match with available resource then resource provider can instruct to the virtual machine to executing the cloud user task. Once, resource providers win the bid then they have complete user task with their stratifications.

Let be U_c total amount of cloud users and n_c represents cloud user number, $U_c = \{1, 2, 3 \dots n_c\}$. Consider R_c is the total amount of cloud resource providers, and r_c expresses cloud resource provider number; where $R_c = \{1, 2, 3 \dots r_c\}$. Total amount of resource types are T_r and t_r indicates position of resource type. Where, $T_r = \{1, 2, 3 \dots tr\}$. Every cloud user request for different types of resource; where, every cloud user have a different price for different resources. The user resource requirement vector can be expressed as $U_m = \{u^m_1, u^m_2, u^m_3, \dots, u^m_{tr}, t_n, b_n\}$ where t_n indicates the total duration for resource utilization hour basis and b_n is the budget of cloud user. Where, cloud resource provider will involve in the bidding process and expose the available resource that is expressed by vector R_c

$=\{ r^c_1, r^c_2, r^c_3, \dots, r^c_{t_r} \}$ and cost of each resource type as a $C_{rc}=R_c=\{ c^{rc}_1, c^{rc}_2, c^{rc}_3, \dots, c^{rc}_{t_r} \}$. Weight vector W_v expressed the different type of weight for every resource can be described as $W_v=\{w^1_v, w^2_v, \dots, w_{kt_r}\}$. The pseudo code of the proposed Algorithm is given below in details.

Input: Collect resource provider details
Output: Visualize resource utilization, task completion time and cost;

- Steps 1: Define j, k, and temp←0;
- Steps 2: Proceed cloud user request with duration and budget to the auctioneer;
- Steps 3: Apply Efficient Resource Utilization Auction Method;
- Steps 4: Assign deadline and convey bidding message among cloud resource providers;
- Steps 5: Confirm the user requested resource for bid booking;
- Steps 6: **If** a resource provider has sufficient resource
- Steps 7: Proceed for bidding operation and advertise the availability of the resources;
Else
- Steps 8: Terminate the bidding process and remove from a list;
- Steps 9: Map nth cloud user with available resource provider;
- Steps 10: Visualizes the resource provider resources;
 $r_c \leftarrow temp$;
 For $j \leftarrow 1$ to t_r ;
 $R_c[r_c][j] = R_c[r_c][j] - U_m[u_m][j]$
 End for
 Define $n \leftarrow n + 1$;
- Steps 11: Filter higher priority based cloud user according to long duration service selection for resources;
 Define $n \leftarrow 1$;
 For $i=1$; r_c to R_c ;
 For $j \leftarrow 1$ to k ;
 If $(R[r_c][j]) < U_m[u_m][j]$
 $i++$;
 Break;
 End for;
 Temp← i;
 Break;
 End for
- Steps 12:** Declare bid auction winners from the bidding process
- Steps 13:** Evaluate bidding value resource wise for cloud user;

- Steps 14:** Estimate the resource cost;
- Steps 16:** Initialize the cloud user requested task for execution;
- Steps 17:** Initiate the virtual machine (VM) for task deployment;
- Steps 18:** Finish cloud user task and destroy the VM;
- Steps 19:** Pay the utilized resource cost;
- Steps 20:** Visualize utilized resources, completion time and cost;
 End

3.5 Winner declaration

The cloud user request is required resources from auctioneer; where the auctioneer will forward available cloud resource provider for the bidding process. Auctioneer declares a winner of the bid, after evaluation of bids application of resource providers. Initially, auctioneer calculates the total weighted requested resources cloud user which can be expressed by weight as explained in equation (refer with: Eq. (1)).

$$W_v t_n = \sum_{i=1}^{t_r} (a_i w_{v_i}) \tag{1}$$

$$Z_{t_n} = \frac{b_n}{w_{v_{t_r}}} \tag{2}$$

Then, auctioneer calculates the price of each resource bid value of cloud user as given by equation (refer with: Eq. (2)). Next, each resource bid value will be multiplied by the time for cloud user-requested resources to offer reliable cloud with a reasonable cost for a long duration; it will ultimately generate profit for resource provider which is expressed in equation (refer with: Eq. (3)).

$$Z_{tn} = Z_n \times T_n \tag{3}$$

The system gives higher priority for long duration resource utilized (planning for long duration), and it will be filtered in descending order with first ranking customers. Simultaneously, the method evaluates bid density of resource for providers' resource provider, and it will be listed in ascending order which is expressed in equation (refer with: Eq. (4)).

$$R_c = \frac{\sum_{i=1}^{t_r} (U(U_c)_{r_i})}{TR} \tag{4}$$

In a filtered list, cloud resource provider has the lowest bidder in the auction. If, cloud resource provider has sufficient resources for cloud user then the method will be mapped the cloud user to the respective cloud resource provider. If the first resource provider is unable to fulfil the cloud user's requirement, then it will iterate with the next resource provider to check their requirement. The process will be until continue; till, cloud user is unable to satisfy with their required resources. After finding a suitable resource, cloud user will pay the resource cost, and the process will be applicable for all users.

3.6 Cost evaluation

Once, cloud user satisfies with their received resources then cloud user needs to pay the resource cost to resource provider that price details are explained in equation (refer with: Eq. (5)). The auctioneer estimates the total resource cost. The auctioneer will share the price information to the cloud user. Cloud user need to pay total utilized resource cost at the end of successful job completions:

$$U_{pn} = \left\{ \left(\frac{b_i}{\left(\sum_{i=1}^{t_n} (a_i) \times t_n \right)} + R_{cr_c} \right) \times \frac{1}{2} \times t_n \times \sum_{i=1}^{t_n} (a_i) \right\} \quad (5)$$

4. Result and discussion

4.1 Implementation setup

The experiment deployed on Intel core i^{6th} processor, 8 GB RAM with 1000 GB Memory. The implementation is done in Java programming languages using JDK 1.8, Net Beans 8.0.2. The implementation is used Cloudsim library to evaluate the performance of the proposed method.

4.2 Input parameters

For a specific experiment, the proposed method prefers workflow's balance factor which is classified into three parts: balanced, semi-balanced and unbalanced [14]. Each workflow has 10, and its workload considers normal distribution (1000000; 1000). The cost of resources is setting from the real Amazon Web Services cost (<http://aws.amazon.com>). To implement dynamic cost strategy, all resources have peak cost, bottom cost and cost change rate. The peak cost and bottom cost are set according to Amazon web services cost; the change rate is 20 %. In general experiment, there are 10 workflows. The amount of job workflow increases one by one from 11 to 20. The input parameters are allocated to deploy the experiment to identify the performance of the proposed algorithm. The input parameter details are given below in Table (refer with: Table 1).

4.3 Simulation results

The module expresses the performance of

Table.1 Cloud experimental input parameter details

Parameters	Value
User	06
Region	06
Data centres	4(DC1,DC2,DC3 and DC2)
Virtual Machine	5(DC ₁),50(DC ₂),25(DC ₃)& 100(DC ₄)
Data centres VM	Xen
No. of PM Wise	4
DC Processing Speed	10000 MIPS
DC VM Policy	Time Shared
Data Centre OS	Linux
VM Memory	512
DC Architecture	X86
Bandwidth	1000 Mbps
Service Broker Policy	Closest Data Centre, Optimize Response Time, Reconfigure Dynamically With load,
User Grouping Factor	10
Request Grouping Factor	10
Executable Length	102400

Table 2. Resource utility (RU) and Task completion time (TCT) in second for balanced, semi-balanced and unbalanced load data categories

Algorithm	Balanced		Semi-balanced		Unbalanced	
	RU	TCT	RU	TCT	RU	TCT
BOSS	66	76×10 ¹⁰	29	81.8×10 ¹⁰	42	81.5×10 ¹⁰
DPAM	85	74×10 ¹⁰	59	79×10 ¹⁰	72	78×10 ¹⁰
ERUAM	93	52×10 ¹⁰	82	61×10 ¹⁰	89	58×10 ¹⁰

evaluation parameters to find out the improvement in the proposed system compare than existing methods. A proposed method highly dedicates to optimize cost, resource and task completion time issues in the cloud. It works between a service provider and user task to minimize the deployment cost, job completion time and optimize the resources in the cloud environment. Proposed technique expresses Resource Utility (RU), and Task Completion Time (TCT) Mathematical model below in details:

4.4 Resource utility (RU)

Resource utilization rate is the percentage of the resources in a given a data centres DC topology that is assigned to VMs. It is measured after the first time the placement algorithm fails to assign a VM topology to any DC (resources achieve saturation). A different utilization rate exists for each type of resource, e.g. memory utilization, bandwidth utilization etc. A high utilization rate indicates that the resources are assigned efficiently. The cloud resource provider will engage in the auction process; when, auctioneer conveys the messages. If, the service provider has sufficient resources then they can participate in the bidding process. Once, the resource providers win the auction then it is liable to offers services to specific cloud users and gets profit. Cloud resource provider can join nth number of auction processes based on their resource availability. So that a significant number of VM requests are responded successfully. We assume that user B spends the cost B_c on specific resource R_c when the cost of resources is fixed. When the cost is dynamic, another resource R_{c0} with the same computing ability may decrease the cost to B_{c0} which is lower than B_c. So user B will select resource R_{c0} and decrease his cost. The cost of resource in auction changes dynamically according to equation (refer with: Eq. (6)):

$$R_p = \begin{cases} R_p^w & \text{if } p \text{ is winner} \\ R_p^w \times \gamma & \text{if } p \text{ is loser and } R_p^w \times \gamma \supset R_p^L \\ R_p^L & \text{if } p \text{ is loser and } R_p^w \times \gamma \subset R_p^L \end{cases} \quad (6)$$

Where p_j refers to the current price of resource that provider j owns. p_i mini refers to the bottom price of resource; γ denotes the rate of cost reduction. Each provider set peak cost, minimum cost and change rate of cost for resources. When, a task starts the auction, providers join the auction and give their bids included computing ability and cost. After one auction is over, providers change the resource cost according to transaction situation. The process is described as equation (refer with: Eq. (6)). Finally, the effective cost is an efficient cost strategy in the cloud market. Resource providers decrease the cost to increase the chance of winning an auction and gain more revenue; simultaneously the users will choose the resource with lower cost.

4.5 Task completion time (TCT)

Task Completion time includes both latency and running times. It is an average value for all users but is not normalized by job size. The TCT carries out the whole request in data centres processing, which is requested by cloud user from different regions. It estimates task scheduling time and VM load optimizations. Therefore, it processes for retrieving the requested query from the cloud data centres. It indicates the promptness in responding to user requests. TCT_{i,j} is the product of completion time and monetary cost of task i on resource j. During the auction, the task selects the resource with minimum TCT as winner according to equation (refer with: Eq. (7)):

$$TCT_{ip} = \left(TS_{ip} + \frac{taskload_i}{Ability_p} \right) \times \left(price_p \times \frac{taskload_i}{Ability_p} \right) \quad (7)$$

For each task, it starts execution only, when its parent tasks are finished because of the partial ordering relation of tasks. t_{i,j} refers to the start time of task i executing on resource j which equals to the latest time. When, its father tasks are finished, and the resource j is spare. The workload i refers to the workload of task ability j, price j are computational ability and price per unit time of jth resource respectively.

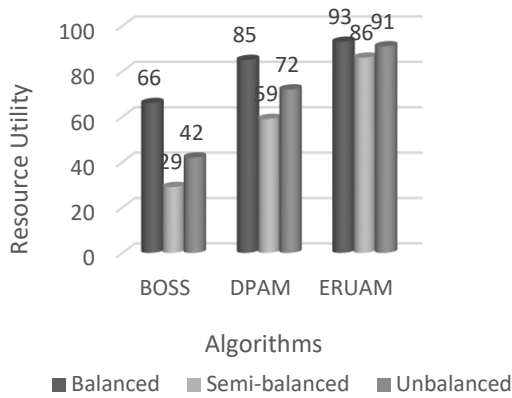


Figure.2 Resource utility (RU) in percentage (%) for Balanced, Semi-balanced and unbalanced load data categories

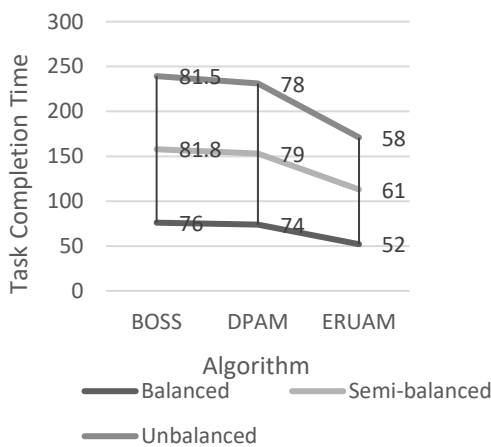


Fig.3 Task completion time (TCT in seconds (sec) for balanced, semi-balanced and unbalanced load data categories

Table (refer with: Table 2) expresses Resource Utility (RU) and Task Completion Time (TCT) for Balanced, Semi-balanced and unbalanced load data categories. Where, proposed Efficient Resource Utilization Auction Method (ERUAM) is compared with Biobjective Scheduling Strategy (BOSS) [14], Dynamic Pricing-Based Allocation Mechanism (DPAM) [14] techniques. Table (refer with: Table 2) expressed the average value of resource utilization (RU) in % (percentage) and task completion time (TCT) in seconds for Balanced, Semi-balanced and unbalanced load data categories of proposed ERUAM with existing methodologies BOSS [14] and DPAM [14]. Proposed ERUAM is evaluated in terms of resource utilization to evaluate how the proposed method is useful for properly utilising the available resource to complete the task in the minimum period with affordable cost. The proposed method has competed with techniques as mentioned above. However, proposed ERUAM results are far better than existing methods. Based on tabular

observation, it noticed that proposed ERUAM performs best compare than existing methods.

According to Figure (refer with Fig. 2 and Fig. 3), proposed Efficient Resource Utilization Auction Method (ERUAM) expresses the resource utilization (RU) and Task Completion Time (TCT) in seconds for Balanced, Semi-balanced and unbalanced load data categories. In terms of resource utilization (RU), the proposed method is competed by BOSS [14] and DPAM [14] on Balanced, Semi-balanced and unbalanced load data categories. Where, DPAM [14] introduced for the auction-based cloud market, to deploy the workflows with minimal task completion time and cost. The motto of the technique is to offer the best price to cloud users for gaining the highest revenue. However, the method failed to optimize resource utilization (RU). BOSS [14] worked based on a reverse auction for resource allocation for tasks of workflows and every task proceed with bidding operation to reduce the price cost and execution time. However, the methods follow that fixed cost economic model during bidding. So, some cloud resource providers have less competition, and they can lose the bidding almost time. The method has very less resource utilization which increases task completion time.

The proposed method works based on a dynamic economic auction model. The dynamic pricing model is presented in the algorithm for pricing of resources. Where, cloud user looks to get actual requirement with the reasonable cost; simultaneously cloud resource provider also works to increase the sale of their resource for meaningful profit. The auctioneer collects the user requirement and conveys the message with time duration to all registered cloud resource providers for the bidding process. Cloud resource providers will check their available resource and time duration, after receiving the message from the auctioneer. If, resource provider has sufficient resources then it will proceed for bid booking and advertise their resource availability. The auctioneer will close the bidding operation after completion of duration. Hence, the system will not allow anybody to involve in the bidding process. Hence, the auctioneer will estimate the overall process; like an evaluation of total resource requirements as per cloud, user need. The method will find the required resource bidding value. The system gives higher priority to those cloud users who look resources for a long duration. The system calculates the resource value of all bidding booked cloud resource provider. The cloud resource provider list will be filtered with the maximum value of services with minimum bidding to find best resource provider. Hence, Auctioneer calculates the

total weighted requested resources, cloud user. Finally, proposed ERUAM improves the 18% resource utilization (RU) and minimizes 20 seconds task completion time (TCT) for Balanced, Semi-balanced and unbalanced load data categories.

5. Conclusion

The paper presents an Efficient Resource Utilization Auction Method to solve the problem of resource allocation using an auction-based resource allocation technique. The proposed method prefers dynamic cost model for the cost of resources. The proposed method provides mutual benefits for both cloud user and cloud resource provider to solve the resource allocation issues. The proposed algorithm efficiently allocates the cloud resource providers' resources and fulfils the demand of cloud users. The system encourages the effective cost mechanism based on resource utilization. Hence, the cloud user request required resources from auctioneer; where the auctioneer will forward available cloud resource provider for the bidding process. Auctioneer declares the winner of the bid, after evaluation of bids application of resource providers. The method will find the required resource bidding value. The system gives higher priority to those cloud users who look resources for a long duration. The system calculates the resource value of all bidding booked cloud resource provider. The cloud resource provider list will be filtered with the maximum value of services with minimum bidding to find best resource provider. Hence, Auctioneer calculates the total weighted requested resources, cloud user. Based on the on obtained results, the proposed method offers mutual profits for cloud resource provider to generate more revenue and cloud users for actual resource finding to fulfil the demand of cloud users. Finally, proposed ERUAM improves the 18% resource utilization (RU) and minimizes 20 seconds task completion time (TCT) for Balanced, Semi-balanced and unbalanced load data categories

In future, the paper can be extended to enhance the business model in cloud computing with the privacy of the bidding process and as well as cloud user and resource providers. The paperwork also should be continued to offers promo offers for regular and newcomer user to increase the sale of a product.

References

[1] S. Zaman and D. Grosu, "Combinatorial auction-based allocation of virtual machine instances in clouds", *Journal of Parallel and*

Distributed Computing, Vol. 73, No. 4, pp. 495-508, 2013.

- [2] D. Kumar, G. Baranwal, Z. Raza, and D. P. Vidyarthi, "A systematic study of double auction mechanisms in cloud computing", *Journal of Systems and Software*, Vol. 125, pp. 234-255, 2017.
- [3] Y. Xia, H. Hong, G. Lin, and Z. Sun, "A Secure and Efficient Cloud Resource Allocation Scheme with Trust Evaluation Mechanism Based on Combinatorial Double Auction", *KSII Transactions on Internet and Information Systems*, Vol.11, No.9, pp. 4197-4219, 2017.
- [4] D. Magalhães, R. N. Calheiros, R. Buyya, and D. G. Gomes, "Workload modelling for resource usage analysis and simulation in cloud computing", *Computers and Electrical Engineering*, Vol. 47, pp. 69-81, 2015.
- [5] M. Aldossary and K. Djemame, "Energy consumption-based pricing model for cloud computing", In: *Proc. of the 32nd UK Performance Engineering Workshop, University of Bradford*, pp. 16-27, 2016.
- [6] Z. Zhou, F. Liu, and Z. Li, "Bilateral electricity trade between smart grids and green data centres: Pricing models and performance evaluation", *IEEE Journal on Selected Areas in Communications*, Vol. 34, No. 12, pp. 3993-4007, 2016.
- [7] J. Guo, F. Liu, T. Wang, and J. C. Lui, "Pricing intra-data centres networks with over-committed bandwidth guarantee", In: *Proc. USENIX ATC*, pp.69-81, 2017.
- [8] P. Bonacquisti, G. Di Modica, G. Petralia, and O. Tomarchio, "A strategy to optimize resource allocation in auction-based cloud markets", In: *Proc. of 2014 IEEE International Conference on Services Computing*, pp. 339-346, 2014.
- [9] A.N. Toosi, K. Vanmechelen, F. Khodadadi, and R. Buyya, "An auction mechanism for cloud spot markets", *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 11, No. 1, pp. 2-19, 2016.
- [10] S.H.H. Madni, M.S.A. Latiff, and Y. Coulibaly, "An appraisal of meta-heuristic resource allocation techniques for IaaS cloud", *Indian Journal of Science and Technology*, Vol. 9, No. 4, pp. 1-15, 2016.
- [11] P. Samimi, Y. Teimouri, and M.A. Mukhtar, "A combinatorial double auction resource allocation model in cloud computing", *Information Sciences*, Vol. 357, pp. 201-216, 2016.
- [12] S. A. Tafsiri and S. Yousefi, "Combinatorial double auction-based resource allocation

- mechanism in cloud computing market”, *Journal of Systems and Software*, Vol. 137, pp. 322-334, 2018.
- [13] G. V. Prasad, A. S. Prasad, and S. Rao, “A combinatorial auction mechanism for multiple resource procurement in cloud computing”, *IEEE Transactions on Cloud Computing*, Vol.4, pp.1-11, 2016.
- [14] X. Li, X. Liu, and E. Zhu, “Auction based dynamic resource allocation in cloud”, In: *Proc. of Asia-Pacific Conference on Business Process Management*, pp.59-69, 2015.
- [15] G. Baranwal and D. P. Vidyarthi, “A fair multi-attribute combinatorial double auction model for resource allocation in cloud computing”, *Journal of systems and software*, Vol. 108, pp. 60-76, 2015.
- [16] F. Sheikholeslami and N. J. Navimipour, “Service allocation in the cloud environments using multi-objective particle swarm optimization algorithm based on crowding distance”, *Swarm and Evolutionary Computation*, Vol. 35, pp. 53-64, 2017.
- [17] A.S. Prasad and S. Rao, “A mechanism design approach to resource procurement in cloud computing”, *IEEE Transactions on Computers*, Vol. 63, No. 1, pp. 17-30, 2014.
- [18] J. Simão and L. Veiga, “Partial utility-driven scheduling for flexible SLA and pricing arbitration in clouds”, *IEEE Transactions on Cloud Computing*, Vol. 4, No. 4, pp.467-480, 2016.
- [19] W. Kong, Y. Lei, and J. Ma, “Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism”, *Optik-International Journal for Light and Electron Optics*, Vol. 127, No. 12, pp. 5099-5104, 2016.
- [20] P. Bonacquisto, G. Di Modica, G. Petralia, and O. Tomarchio, “A procurement auction market to trade residual cloud computing capacity”, *IEEE Transactions on Cloud Computing*, Vol. 3, No. 3, pp. 345-357, 2015.