A Survey on Resource Allocation Strategies in Cloud

Satveer Singh
Computer Science and Technology
Sant Longowal Institute of Engineering and Technology (SLIET)
Sangrur, Punjab, India, 148106
Email: veerboss.singh@gmail.com

Jaspal Singh
Computer Science and Technology
Sant Longowal Institute of Engineering and Technology (SLIET)
Sangrur, Punjab, India, 148106
Email: safrisoft@yahoo.com

Abstract— The cloud computing is an emerging technology in the era of Internet. It is advancement to grid computing. Basically cloud computing is combination of distributed, parallel and grid computing. The cloud is a shared pool of virtual resources and these resources are shared by multiple users at different geographical area. As cloud computing is being enhanced day by day, several issues are also increased like security, resource allocation, performance and cost. However one of the major pitfalls in cloud computing is related to optimizing the resources being allocated. In this paper, various resource allocation strategies are classified on the basis of different parameters, priority, hybridization, and other techniques. Each category has details of number of resource allocation techniques. This work is useful for researchers to get an idea of various resource allocation strategies at a glance.

Keywords- cloud computing; resource allocation; different parameters; priority; hybridization

I. LITERATURE: A SURVAY OF RESOURCE ALLOCATION

The National Institute of Standards and Technology (NIST) defines, "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models". The five essential characteristics of cloud computing are: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service [1].

In cloud computing, resources are provided by the cloud provider to cloud users. Efficient resource allocation is central challenge for cloud provider. Many resource allocation strategies have been proposed by many researchers. The resource allocation strategies have been classified based on multiple criteria, out of which four different criteria have been discussed in this paper. These criteria are: (1) resource allocation strategy based on different parameters, (2) resource allocation strategy based on priority, (3) resource allocation strategy based on hybridization, and (4) resource allocation strategy based on particular technique.

A. Resource Allocation Based on Different Parametrs

In this category, the researchers have proposed various resource allocation strategies on the basis of different parameters. The cloud resources can be classified on the basis of these different parameters. The cloud user can easily access a resource by giving input parameter. The different parameters which are generally used by researchers are: processing element, memory, network bandwidth, time deadline, cost etc. The various resource allocation techniques are:

1) Optimal resource allocation with limited energy power consumption: Kazuki and Shin-ichi proposed an optimal resource allocation algorithm with limited energy power consumption. Three parameters were taken into consideration to allocate resources: processing element, network bandwidth, and electric power consumption. The maximum numbers of request were processed by this approach. Total consumption of electric power was reduced by aggregating requests being processed at multiple area.

2) Optimal joint multiple resource allocation: Shin-ichi proposed a resource allocation method in which processing element and network bandwidth were taken as parameters. Both type of the resources were allocated simultaneously to each cloudlet and services were rented on an hourly basis. A fair joint multiple resource allocation was proposed based on this model in which fair allocation was done between multiple users.

3) Cost and deadline optimization with resource allocation: Joy and Kumar proposed resource allocation approach with two parameters, cost and deadline. A virtual machine was allocated according to time deadline of user request. A benefit was provided to the user in terms of the user annotated parameters and a virtual machine was created based on the user demand. Since, this approach also involves the migration of cloudlets, which introduces the overhead. It might delay the response time of the cloudlet.

4) Job oriented scheduling: Vignesh et al. proposed a model for job-oriented scheduling. The computing resource could be allocated according to the rank of the job. The time parameter of three algorithms Rond Robin (RR), pre-emptive priority, and Shortest Remaining Time First (SRTF) had been taken into consideration. As a result, it had been computed that SRTF had the lowest time parameter in all respects and was the most efficient algorithm for resource scheduling. The time parameter includes Turn Around Time (TAT) and Waiting Time (WT).

5) Dynamic resource allocation: Saraswathi et al. proposed an approach in which high priority jobs (time deadline of job is low) were not delayed by lower priority jobs (time deadline of job is high). The cloud resources were dynamically allocated to a user request within the time deadline. When a job arrived, the availability of the virtual machine was checked. If there was no virtual machine available, then a low priority job which was in execution would pre-empted for high priority jobs. When other jobs running on virtual machines were completed, the job which was paused earlier could be resumed if the job type was suspend-able. The amount of computational overhead by suspending the lower priority jobs becomes more, when numbers of high priority jobs were increased. This algorithm might be suffered from the problem of starvation.

6) Cloudlet allocation strategy: Banerjee et al. proposed a new cloudlet allocation strategy with one parameter million instructions per second (MIPS). The virtual machines and the cloudlets were sorted in descending order according to their MIPS and million instructions (MI) respectively. All the sorted virtual machines and the cloudlets were stored in a VM-List and Cloudlet-List respectively. In allocation, the maximum capacity virtual machine was allocated first. After allocation, the remaining load capacity of the allocated virtual machine was compared with maximum load capacity of the other virtual machine in the VM-List and the VM-List was sorted again. The performance of this strategy would degrade if the configuration of most of the cloudlets was same. In this case, the approach resulted in the increased overhead only.

7) Demand-based preferential resource allocation: Kumar and Saxena proposed a demand-based preferential resource allocation technique, which was used to design a market-driven auction mechanism by which users were identified for resource allocation based on their payment capacities. A payment strategy was also implemented based on buyer's service preference. Three types of resources (virtual machine with processing element, memory, and bandwidth) were maintained by cloud provider. A preference was given by cloud provider to user if maximum amount was paid by him. Service rates were also decided by user demands.

TABLE I. VARIOUS RESOURCE ALLOCATION STRATEGIES BASED ON DIFFERENT PARAMETRS

S. No.	Method Proposed (Year)	Parameters Used	Goal Achieved	Simulator
01.	Optimal Resource Allocation with Limited Energy Power Consumption (2011)	PE, RAM, BW	Energy reduced, Maximum request processed	C
02.	Optimal Join Multiple Resource Allocation (2011)	PE, BW	reduce the request loss probability	C
03.	Cost and Deadline Optimization with Resource Allocation (2013)	Cost, Deadline	User can give parameter as input, VMs are created as user input	Customized environment
04.	Job Oriented Scheduling (2013)	Time (TAT, WT)	Resources are allocated based on rank of job	Customized environment
05.	Dynamic Resource Allocation Scheme (2015)	Deadline, Priority	Assignment of resource is done within given time deadline	CloudSim
06.	Cloudlet Allocation Strategy (2015)	MIPS	Highest capacity resource is allocated to highest demand request	CloudSim
07.	Demand-Based Preferential Resource Allocation (2015)	PE, RAM, BW	Resources are allocated based on payment capacities of user	CloudSim

B. Resource Allocation Based on Priority

In this category, the researchers have been proposed various resource allocation strategies based on priority. Priority can be given to cloud request or cloud resources as you need. If the priority was given to cloud resource than a cloud resource with highest priority would be allocated first. Similarly a cloud request with highest priority would be server by cloud provider as soon as possible. The various resource allocation techniques based on priority are:

1) Priority based dynamic resource allocation: Pawar and Wagh proposed a priority based dynamic resource allocation approach with a pre-empt-able task execution. The multiple parameters such as memory, network bandwidth, and required CPU time was also considered by this approach for resource allocation. Initially, the virtual machine list and cloudlet list was sorted based on priority. Then tasks were executed on virtual machines based on their priority and preemption was made by controller only if when any high priority cloudlet arrived. This approach suffered from the problem of starvation.

2) Priority based resource allocation: Natasha and Gill proposed a priority based resource allocation method for handling a situation, where two or more requests at a particular instance of time had same priority. The work flow of this model was discussed into three stages. In stage I, initial parametric values were generated such as attaching priority to request of cloud user and the user requests were sorted in descending order based the priority. After priority assigned, requests with same priority were grouped into a group known as open group. A ready queue was generated for all the requests which were in open group and had not been executed yet in stage II. In stage III, grouped requests were executed. A threshold value of available resources was set and when the load needed by one or more requests in ready queue exceed the threshold limit; request should wait in waiting queue.

3) Pre-emptive scheduling of online real time services: Santhosh and Ravichandran proposed a scheduling algorithm with pre-emptive execution to overcome from the non-pre-emptive scheduling limitations. In non-pre-emptive scheduling, if any high priority task arrived and wait because of un-availability of virtual machine then system performance degrades. In this work, When a high priority task arrived in between execution of other task and the deadline of the task which was about to miss then the task would be migrated to another virtual machine. This work was compared with traditional EDF and other non-pre-emptive scheduling algorithms. The results show that response time was reduced and overall system performance was improved.

TABLE II. VARIOUS RESOURCE ALLOCATION STRATEGIES BASED ON PRIORITY

S. No.	Method Proposed (Year)	Priority Assigned to	Goal Achieved	Simulator
01.	Priority Based Dynamic Resource Allocation (2012)	VM-List, Cloudlet-List	Resource allocation based on priority with pre-emption	Customized environment
02.	Priority Based Resource Allocation (2013)	Cloudlet-List	Issue of when two job have same priority at same instance of time	C#
03.	Pre-emptive Scheduling of Online Real Time Services (2013)	Cloudlet-List	Overcome from limitations of non-pre-emptive priority algorithms	Customized environment

C. Resource Allocation Based on Hybridization

In this category, the researchers have proposed several resource allocation strategies on the basis of hybridization of two or more techniques. With the help of this kind of mixture of resource allocation strategies, limitations of particular algorithms are reduced. The various resource allocation techniques based on hybridization are:

1) Efficient resource allocation: Dinesh et al. proposed a combination of Berger model and Neural Network to overcome from limitations of Berger model. The submitted jobs were classified based on different parameters like bandwidth, memory, and completion time and resource utilization. The classified user tasks were passed to the Neural Network. With the help of hidden layer of Neural Network, the jobs were matched with the resources by weight adjustment. In this work, results show that utilization of bandwidth was improved, completion time was reduced, and the performance of the system was increased.

2) Modified throttled: Domanal and Reddy proposed a local optimized load balancing approach which was used to handle the load at servers by considering both availability of VMs for the request and uniform load sharing among the VMs for the number of requests served. Two objectives were taken, one was the response time required to serve the requests and other was the distribution of load among existing VMs. Results show

that, the response time of modified throttled algorithm had improved considerably and distribution of load nearly uniform among VMs.

3) Priority based earliest deadline first: Gupta et al. proposed two task scheduling algorithms, one was priority based and other was earliest deadline first (EDF) scheduling algorithm. In this work, tasks were considered as pre-empt-able on the basis of priority. When a task with high priority was arrived in between execution of other task and if deadline of task which was in execution was about to miss then that task would be migrated to another virtual machine. Basic EDF algorithm was non-pre-emptive but in this work pre-emptive EDF was introduced.

4) Load balancing using novel hybrid scheduling: Shridhar et al. proposed a hybrid algorithm for load balancing in a distributed environment in which methodology of Divide-and-Conquer and Throttled were combined. Combination of these two algorithms was referred as DCBT. In this work, total execution time of the tasks was reduced and resource utilization was maximized. The DCBT algorithm was compared with Modified Throttled algorithm and results show that execution time was reduced and requests were evenly distributed over virtual machines.

TABLE III. VARIOUS RESOURCE ALLOCATION STRATEGIES BASED ON HYBRIDIZATION

S. No.	Method Proposed (Year)	Combination Used	Goal Achieved	Simulator
01.	Efficient Resource Allocation (2012)	Berger Model And Neural Network	Overcome from limitations of Berger model	CloudSim
02.	Modified Throttled (2013)	Round-Robin And Throttled	Workload distributed uniformly compared to RR and Throttled	CloudAnalyst
03.	Priority Based Earliest Deadline First (2014)	Priority And EDF	Overcome from limitations of non-pre-emptive EDF	CloudSim
04.	Load Balancing Using Novel Hybrid Scheduling (2015)	Divide & Conquer And Throttled	Reduced execution time & increased resource utilization compared to modified throttled	Customized environment And CloudAnalyst

D. Resource Allocation Based on Perticular Techniques

In this category, the researchers have proposed several resource allocation strategies on the basis of particular techniques. The various resource allocation techniques are:

1) Threshold-based dynamic resource allocation: Lin et al. proposed the main idea of the threshold-based dynamic resource allocation scheme to monitor and predict the resource needs of the cloud application and to adjust the virtual resources based on application's actual needs. In this scheme, virtual resources for cloud application could dynamically reconfigured according to the load changed in cloud application, so by doing this resources were saved and resource utilization increased.

2) Dynamic resource management using virtual machine migration: Mishra et al. the important role of live VM migration in dynamic resource management of virtualized cloud system was discussed. Migration enables several resource management goals like consolidation, load balancing, and hot spot mitigation. The components were discussed: when to migrate, which VM to migrate, where to migrate. In this work, categorization and details of migration heuristics at reducing server sprawl, minimized power consumption, load balance across physical machines and so on was done.

3) Capacity based resource allocation: Devi and Vetha proposed a capacity based resource allocation algorithm to overcome from the situation, when multiple users might be requested the same resource such as processor, servers, storage etc. This algorithm was proposed based on the different parameters like number of processor requests, capacity requests etc. In this work, cloud consumer could know the capacity available using virtual machine server. Only theoretical framework was done in this paper.

4) Continuous resource allocation: Zhou et al. proposed an approach in which all available resources of cloud were mapped into a ZBtree. By using this approach, user request migration was reduced. Three main steps were included by this technique: information collection, resource allocation and continuous mapping. Computational overhead was reduced in this approach. However, this approach was based on the prediction of the jobs arrival, so it was prediction dependent. So if this prediction accuracy was reduced, the goal of achieving a better utilization of resources may fail.

5) User-oriented resource allocation: Zhou et al. proposed a user-oriented resource allocation based on clustering method. Time interval and amount of computing were applied by the user as resource parameters. In this paper, the scheduling problem was converted into clustering problem for resource allocation. A matrix was used for overlap degree that was a condition when a resource allocated to multiple users at same time. A user request included computing ability and time period.

6) Agent based best-fit resource allocation: Shyam and Manvi proposed an agent based best-fit resource allocation scheme in which resource utilization was increased, service cost decreased and execution time was also reduced. In this work, two types of agents were taken: user's cloudlet agent and provider's resource agent. Best-fit approach was used by resource agent at the server to allocate resources for job received from cloudlet agent. This work was compared with First Come First Serve (FCFS) and Round Robin (RR). Best-fit approach performed better in terms of virtual machines allocation, job execution time, cost, and resource utilization.

7) Efficient dynamic resource allocation: Nagpure et al. proposed a dynamic resource allocation system by which overload on server could be avoided by evenly resource allocation. The concept of skewness was used to calculate uneven utilization of multiple resources on the server among virtual machines and available server resources checked. To avoid overload on server, future load was predicted by Fast Up Slow Down (FUSD) algorithm. In this work, performance was optimized in terms of server resource utilization with minimum energy consumption and task migration among virtual machines.

TABLE IV. VARIOUS RESOURCE ALLOCATION STRATEGIES BASED ON PERTICULAR APPROACH

S. No.	Method Proposed (Year)	Based on	Goal Achieved	Simulator
01.	Threshold-Based Dynamic Resource Allocation (2011)	Prediction	Resources can dynamically reconfigure according to load change	CloudSim
02.	Dynamic Resource Management using Virtual Machine Migration (2012)	Virtual Machine Migration	When to migrate, which VM to migrate, where to migrate	Not simulated
03.	Capacity Based Resource Allocation (2014)	Capacity of Resources	Issue of when multiple users requesting for same resource	Not simulated
04.	Continuous Resource Allocation (2015)	Prediction	Request migration is reduced	C++
05.	User-Oriented Resource Allocation (2015)	Clustering of Resources	Issue of when a resource allocated to multiple users	Matlab
06.	Agent Based Best-Fit Resource allocation (2015)	Best-Fit Method	Performs better in terms of VM allocation, execution time, cost, and resource utilization	CloudSim
07.	Efficient Dynamic Resource Allocation (2015)	Prediction	Solve problem of uneven distribution of workload on servers	not simulated

II. CONCLUSION

The cloud computing has become a new age technology. Resource allocation strategies are proposed by various researchers with some objective. The motive of resource allocation may be maximum resource utilization, minimum cost, minimum execution time and response time etc. In this paper, categorization of resource allocation strategies is done on the basis of different parameters, priority, and hybridization. With the help of this categorization it is clear that cloud resources must be allocated efficiently to meet SLA rules and to fulfill several objectives. Each resource allocation strategy has its benefit in some sense.

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," In Communications of the ACM, vol. 53(6), p. 50, December 2010.
[2] K. Mochizuki and S. I. Kuribayashi, "Evaluation of optimal resource allocation method for cloud computing environments with limited electric power capacity," In Network-Based Information Systems (NBiS), 14th International Conference on IEEE, pp. 1-5, September 2011.
[3] S. I. Kuribayashi, "Optimal joint multiple resource allocation method for cloud computing environments," In arXiv preprint arXiv: 1110.1730, Vol.2, No.1, 2011, pp. 1-8.

[4] J. Joy and L. K. Kumar, "Cost and deadline optimization along with resource allocation in cloud computing environment," In Advanced Computing and Communication Systems (ICACCS), International Conference on IEEE, pp. 1-6, December 2013.
[5] V. Vignesh, K. S. Sendhil Kumar, and N. Jaisankar,"Resource management and scheduling in cloud environment," In International Journal of Scientific and Research Publications, vol. 3(6), 2013, pp.1-6.
[6] T. Saraswathi, Y. R. A. Kalaashri, and S. Padmavathi, "Dynamic resource allocation scheme in cloud computing," In Procedia Computer Science, vol.47, 2015, pp. 30-36.
[7] S. Banerjee, M. Adhikari, S. Kar, and U. Biswas, "Development and analysis of a new cloudlet allocation strategy for QoS improvement in cloud," In Arabian Journal for Science and Engineering, vol. 40(5), 2015, pp. 1409-1425.
[8] N. Kumar and S. Saxena, "A Preference-based Resource Allocation in Cloud Computing Systems," In Procedia Computer Science, vol. 57, 2015, pp. 104-111.
[9] C. S. Pawar and R. B. Wagh, "Priority based dynamic resource allocation in cloud computing," In Cloud and Services Computing (ISCOS), 2012 International Symposium on IEEE, pp. 1-6, December 2012.
[10] N. Gill, "Enhanced priority based resource allocation in cloud computing," In The Next Generation Information Technology Summit (4th International Conference), pp. 121-126 , September 2013.
[11] R. Santhosh and T. Ravichandran, "Pre-emptive scheduling of on-line real time services with task migration for cloud computing," In Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on IEEE, pp. 271-276, February 2013.
[12] K. Dinesh, G. Poornima, and K. Kiruthika, "Efficient resources allocation for different jobs in cloud," In International Journal of Computer Applications, vol.56(10), 2012, pp.30-35.
[13] S. G. Domanal and G. R. M. Reddy, "Load Balancing in Cloud Computingusing Modified Throttled Algorithm," In Cloud Computing in Emerging Markets (CCEM), IEEE International Conference on IEEE, October 2013, pp. 1-5.
[14] G. Gupta, V. K. Kumawat, P. R. Laxmi, D. Singh, V. Jain, and R. Singh, "A simulation of priority based earliest deadline first scheduling for cloud computing system," In Networks & Soft Computing (ICNSC), 2014 First International Conference on IEEE, August 2014, pp. 35-39.
[15] S. G. Domanal and G. R. M. Reddy, "Load Balancing in Cloud Environment Using a Novel Hybrid Scheduling Algorithm," In 2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM) on IEEE, November 2015, pp. 37-42.
[16] W. Lin, J. Z. Wang, C. Liang, and D., "A threshold-based dynamic resource allocation scheme for cloud computing," In Procedia Engineering, vol. 23, 2011, pp.695-703.
[17] M. Mishra, A. Das, P. Kulkarni, and A. Sahoo, "Dynamic resource management using virtual machine migrations," In Communications Magazine on IEEE, vol.50(9), 2012, pp.34-40.
[18] K. Vimala Devi and S. Vetha, "Capacity based resource allocation in cloud," In Communication and Network Technologies (ICCNT), 2014 International Conference on IEEE, December 2014, pp. 24-26.
[19] Z. Zhou, H. Zhang, X. Yu, and J. Guo, "Continuous resource allocation in cloud computing," In Communications (ICC), 2015 IEEE International Conference on IEEE, June 2015, pp. 319-324.
[20] H. Zhou, S. Deng, H. Huang, and Y. Wu, "Resource allocation in cloud computing based on clustering method," In Systems Conference (SysCon), 2015 9th Annual IEEE International on IEEE, April 2015, pp. 489-494.
[21] G. K. Shyam and S. S. Manvi, "Resource allocation in cloud computing using agents," In Advance Computing Conference (IACC), 2015 IEEE International on IEEE, June 2015, pp. 458-463.
[22] M. B. Nagpure, P. Dahiwale, and P. Marbate, "An efficient dynamic resource allocation strategy for VM environment in cloud," InPervasive Computing (ICPC), 2015 International Conference on IEEE, January 2015, pp.1-5.