

# Improved Hyper-Heuristic Scheduling with Load-Balancing and RASA for Cloud Computing Systems

Geetinder kaur and Sarabjit kaur

*Department of Computer Engineering & Technology  
CTITR Maqsudan Jalandhar, Punjab, India  
geetinder810@gmail.com  
er\_sarabjit35@rediff.com*

## **Abstract**

*Nowadays cloud computing has turned into a key innovation and has become a great solution for indulging a flexible utility oriented , online allocation and storage of computing resources and client's information in lower expense, on- interest and dynamically scalable framework on pay per use premise. This technology is a new pattern emerging in IT environment with immense necessities of framework and resources. Job Scheduling Problem is an essential issue. For efficient usage and managing resources, administrations, scheduling plays a critical role. This paper apportion the performance enhancement of Hyper- Heuristic Scheduling Approach to schedule cloudlets and resources, by taking account of both , computation time and transmission cost with two detection operators. Load Balancing and RASA concept is applied for efficient Load Scheduling, resource utilization and thereby enhancing the overall performance of cloud computing environment. The numerical investigations of HHSA were performed on CloudSim. Experimental results generated via simulation shows that enhanced heuristic scheduling approach is much better than individual heuristic approach in terms of minimizing makespan time.*

**Keywords:** *Cloud Computing; Job Scheduling Problem; Hyper-Heuristic Scheduling Algorithm; Load Balancing; Resource-Aware Scheduling Algorithm; Makespan Time*

## **1. Introduction**

Numerous accessible arrangements have been expected to enhance, to improve the execution of information systems; generally with respect to calculation, examinations and limit. Since distributed and in addition parallel computing was comprehensively used to update the execution of an assortment of information systems for diverse methodologies and genetic limitations in distinctive ages. The rule issue is the approach to manage these computer resources viably, regardless of which consideration it is for. Among them, the most crucial one for the successful operation of computer system is scheduling. Scheduling is essentially the allotment of various jobs to given resources in given time period. It is one of the most eminent activities that executes in the cloud environment, to expand the efficiency of the workload to get utmost profit.

Lately, the new propel paradigm has been effectively utilized for computer and data frameworks i.e. Cloud computing. It is exceptionally mechanized data innovation administrations similar to the way; automobile manufactures adopted the mechanical production system in the mid twentieth century. Organizations today are embracing cloud computing as more proficient and eligle way for building their data innovation stages. Cloud computing conveys self-administration, self-register, organizing, self-administration stockpiling, client relationship administration and huge information investigation. It is really about hardware and software capability delivered virtually to any

device. This innovation is overall philosophy and outline concept for empowering progressive business models that are significantly changing the nature of society and commerce. Cloud approach is the union of three major trances; Virtualization, Utility Computing and Software-as-a-Service based on pay-as-you-go premises, which is the hallmark of today's computing. It is the mass of framework, applications, administrations which are floated in datacenter. A large portion of the Traditional scheduling algorithms are standard –based scheduling algorithms generally used on today's appropriated systems. These algorithms are simple and easy to actualize; in view of issue in handling the broad scale or complex scheduling issues these algorithms are improper in acquiring the ideal results. Heuristics are applied to scheduling on cloud computing Systems. To solve issue, and reasons of learning and disclosure, heuristics alludes to several experience based frameworks in discovering a solution which is not guaranteed to be ideal but it is adequate for the given arrangement of objectives regardless. The most crucial objective of Hyper-Heuristic is to choose right algorithms for a specific issue on the bases of bundle of existing algorithms and their execution to some degree. The essential arrangement of the proposed algorithm is to hold the quality of low-level heuristic algorithms such as, First-In-First-Out (FIFO) and Max-Min by coordinating them into single algorithm. To further improve the performance of Hyper-Heuristic approach, various techniques have been applied Load Balancing is an essential aspect of cloud computing environment. Productive Load Balancing plan ensures effective load scheduling of overloaded tasks, effective resource use by provisioning of user requirements and resources to cloud clients on – interest premise in pay-as –you-go basis. For further resource utilization in respect of concurrency Resource-Aware Scheduling algorithm is used to enhance the overall performance of the system by degrading the makespan time.

The remaining paper is coordinated as follows. Section II begins with a brief retrospect of Job Scheduling problem and Hyper-Heuristic. Related Work is examined in Section III. In Section IV, The Proposed work has been discussed. In Section V Experiments and Results are computed and Finally, Section VI draws the conclusion remarks together with some conceives about the future research.

## 2. Scheduling in Cloud

In Cloud Computing, Scheduling assumes a fundamental part in effectively dealing with the computer administrations; it is the movement of captivating choices in regards to the distribution of accessible limit and/or resources to jobs and/or clients on time. Million of clients offer cloud administrations by presenting their large number of processing tasks to the cloud computing environment. Scheduling of these huge numbers of tasks is a conflict to the cloud environment. The scheduling crisis in cloud makes it hard to work out, dominantly on account of substantial composite jobs like workflows, so to solve these types of large problems many algorithms are proposed. Scheduling is the procedure of apportioning tasks to accessible resources on the premise of task's qualities and need. The primary objective of scheduling is expanded use of the resources without influencing the administrations provided by cloud. Scheduling technique in cloud is segregated into three stages to be specific;

- 1) **Resource finding and sifting:** In Resource finding and sifting the datacenter expert finds the advantages present in the framework structure and assembles status data about the benefits.
- 2) **Resource determination:** In Resource determination the objective resource is picked in context of the necessities of assignments and resources. This is a picking stage.
- 3) **Task section:** In errand parcel, the task is dispersed to pick resource.

The clients will only need to present the task and indicate the details with that of their necessities. Everything else is handled by the representative i.e. broker of the cloud supplier. The task is assigned to the resources of the virtual machine and executed.

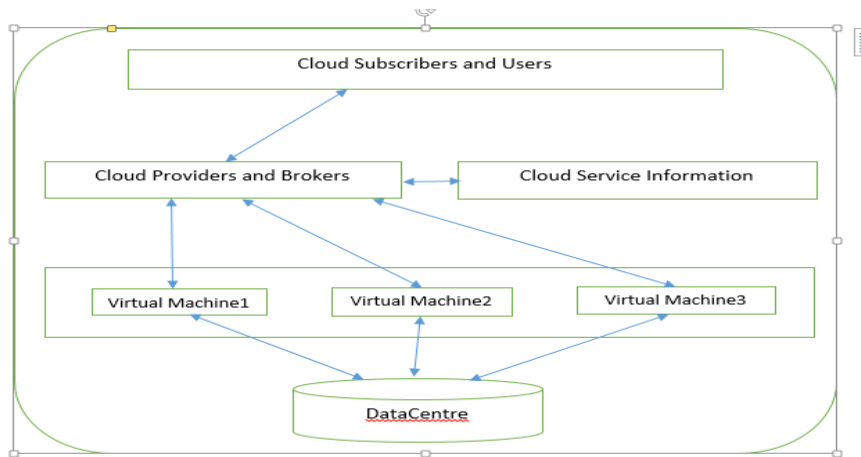


Figure 1. A Block Diagram of Scheduling Process [26]

### 3. Job Scheduling Problem

In software engineering and operation research, JSP is an optimization issue in which ideal jobs are allocated to resources at specific times. The ideal response for issue including  $n$  jobs must be transformed on  $m$  machines, chooses the case of landing of jobs on every one machine in order to complete all the jobs on all the machines in the base total time imitating the same handling operation request when going through the machines with no priority requests. [4] The issue is to find the find the ideal job groupings, setup times on the machines in minimum time by using the ACO computation. A job shop ordinarily comprises of large number of general purpose machines, as opposed to several purpose machines which would typically happen in a assembly line. Every job relying on its technological prerequisites, requests handling on machines in an order. The JSP should be an extremely puzzling issue. Numerically, the greatest no. of possible progressions with  $n$  jobs and  $m$  machines is  $(n!)^m$  i.e. greatly considerable. The issue is ordinarily explained by close estimation or heuristic procedures. The prerequisite for job scheduling in cloud focuses on a couple of parameters, for instance, load equality, throughput, Quality of Service (QoS), running time, adequacy, cost, space and so forth. Besides, upgrade the class of entire distributed computing environment.

#### Need of scheduling

The task scheduling objectives of Cloud processing provide ideal task scheduling to clients, and provide the entire cloud framework throughput and QoS in the meantime. Following are the needs of job scheduling in cloud computing:

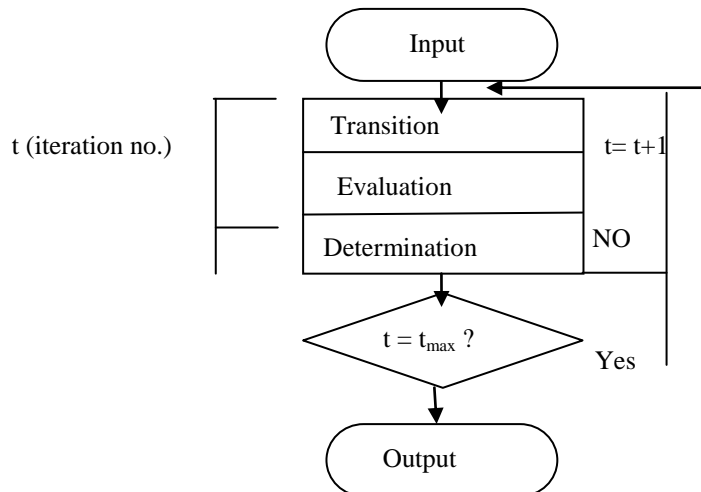
- 1) **Load Balance** - Load Balancing and task scheduling has firmly related with one another in the cloud environment, task scheduling system responsible for the ideal coordinating of tasks and resources. Task scheduling algorithm can maintain load balancing. So load balancing get to be another imperative measure in the cloud.
- 2) **Quality of Service** - The cloud is mainly to furnish clients with processing and cloud storage administrations, resource interest for clients and resources supplied by supplier to the clients in such a way so that , quality of administration can be

accomplished. When job scheduling administration comes to job portion, it is important to ensure about QoS of resources.

- 3) **Best running time** - Jobs can be partitioned into diverse classes as per the needs of clients, and after that set the best running time on the premise of distinctive objectives for every task. It will enhance the QoS of task scheduling indirectly in a cloud framework.
- 4) **Economic Principles** - Cloud computing resources are broadly appropriated all through the world. These resources may belong to distinctive associations. They have their own particular administration approaches. As a business model, cloud computing as indicated by the distinctive necessities, give significant administrations. So the interest charges are reasonable.

#### 4. Hyper-Heuristics

Heuristics are the problem solving mechanism which can be utilized to settle the testing and non-routine issues. The chief three key administrators; Transition, Evaluation and Determination (TED) of heuristics has been utilized to search for the conceivable solutions on the convergence procedure. Transition (T) makes the arrangement (s), by utilizing routines which could be either perturbative or constructive or both. Evaluation (E): measures the fitness of (s), by utilizing predefined estimation. Determination (D): decides the next search directions based on s) from the transition operator and evaluation operator. The extended adaptation of heuristic merge two or all the more high – performance scheduling algorithms which can give a better scheduling in a reasonable time i.e. Hyper-heuristics. It is intended to expertise their heuristic evaluation process. Hyper-heuristics plan to discover some algorithms that are equipped for solving a entire scope of issues, with little or non-coordinate human control. There may be an incalculable number of heuristics from which one can decide for solving an issue, and each heuristic has its own specific points of interest and inconveniences. The thought is to consequently devise algorithms by solidifying the quality and modifying for the shortcomings of known heuristics. A typical infrastructure incorporates a high level methodology and several low level heuristics [1]. At the point when an issue case is given, the high level strategy picks which low level heuristic ought to be applied and this depends on the search space of the issue and the present issue state. The issues are solved by finding a solution from the arrangement of every possible solution for a given issue, which is viewed as the "hunt space". The learning point ought to refine the algorithms, so that the algorithm solutions consequently address the needs of the preparation set and issues of a certain class can be clarified more gainfully. The reaction mechanism ought to move towards ideal algorithm solutions in the workspace, as it helps the determination of heuristic. Hyper Heuristic is the coordination of two or more heuristic algorithms and expects to tell what arrangements of meta-heuristics are used to solve the current issue. Moreover it can be used to characterize what meta-heuristic fits better to which issue. The key idea of Hyper-heuristic is to use "one and one and only "heuristic algorithm at each iteration, with a specific end goal to keep up high search diversity to build the possibility of finding better solutions at later cycles while not growing the count time .



**Figure 2. A Basic Outline of Heuristics [21]**

## 5. Prior Work

Various authors have mulled over the JSP and have been seen as N-P hard.. With the usage of Heuristic strategy, a couple of frameworks were proposed by authors to handle this scheduling issue and among those schedules that have accomplished best results are: In 1985, Davis proposed Job Shop Problem with the use of Genetic Algorithm. There are various such works nearby the use of advancement procedures. Shortest Processing Time (SPT) cross breed heuristic methodology has been proposed by Zhon and Feng for dealing with scheduling issue. Feasible pheromone adjustment system for advancement of fundamental ant structure which offers in examination of the arrangement space is proposed by Zhang.J. [1] For viable Task scheduling and resource usage, RASA a new task scheduling algorithm , comprised of two conventional algorithms Max-Min and Min-Min , involving both the distribution and scalability attributes of grid resources is proposed by S.Parsa. [3] In 2010, Y.Fang proposed a two low-level task scheduling mechanism taking into account Load Balancing , for the improvement of resource usage , viable meeting of dynamic user necessities , thereby enhancing the overall performance of the cloud computing environment.[5] H.chang has introduced an resource scheduling algorithm based on dynamic load balance. Distinctive information processing and transferring power of hubs in cloud and transfer deferral between hubs in cloud is considered. For the improvement of proficiency of cloud computing and minimizing the average reaction time of tasks , the algorithm chooses the "best" hub to fulfill the assignment.[10] Total Make span time of set of jobs is minimizing by applying heuristic method of SPT (Shortest Processing Time) and methodology of LMC (Largest Marginal Contribution) by Aftab.M.T. [12] In 2012, S.B.Zhan has proposed an examination concerning the usage of Improved Particle Swarm Optimization combined with Simulated Annealing Algorithm in resource scheduling procedure of cloud computing to streamline the JSP, by expanding the convergence speed and use proportion of resources. [11] O.M.Elzeki has proposed an Improved Max-Min Algorithm in Cloud computing to outperform scheduling map at least similar to RASA ( Resource Aware Scheduling Algorithm ) map altogether finish time for submitted jobs. The algorithm is based on thorough investigation of RASA algorithm and the Max-Min system. Expected execution time is taken rather than complete time as a selection. [14] In 2013, R.G.Babukarthik have proposed a Hybrid Algorithm build on Ant Colony Optimization and Cuckoo Search to advance JSP.[15] M.Katyal has discussed the overview of cloud innovation and its components by concentrating on load Balancing of cloud computing with some of existing load balancing procedures , which are dependable to deal with the load when

some hub of the cloud framework is over-loaded and others are under stacked. [16] H.Chen has proposed an enhanced Load balance algorithm on the ground of Min-min calculation guided by user- priority for the complete fulfillment of user's interest, in order to diminish the makespan time and expand the resource usage. [18] In 2014, C.W.Tsai proposed a novel Hyper-Heuristic Scheduling Algorithm to comprehend JSP to decrease makespan time and to discover better scheduling solutions for cloud computing frameworks. Two detection operators have been utilized by the proposed algorithm to adjust the intensification and expansion in the hunt of solutions during the convergence procedure. [21] In 2014, Y.Mao has proposed a Max-Min task scheduling algorithm for load-balance in the elastic cloud. To realize the load- balancing, the proposed algorithm maintains a task status table to estimate the real-time load of virtual machines and the expected completion time of tasks, which can allocate the workload among nodes and realize the load balance.[24] B.T.Bini has proposed Hyper-Heuristic Scheduling on Cloud based frameworks. Genetic and Simulated Annealing Algorithms are utilized as a part of the candidate pool as low-level heuristic algorithms. In further the Differential evolution consolidated with the Genetic algorithm to expand the execution, maximum Lateness, maximum tardiness, makespan and maximum stream time are the execution measurements, utilized to make examinations.[27] B.Kanani has examined a brief review of Max-Min Task Scheduling Algorithm for Cloud Computing .In this paper the Proposed algorithm is acquainted to avoid drawbacks of Max-Min algorithm in order to lessen the makespan and build the resource usage with considering client priority, so that the import job will execute first as indicated by their priority. The job which has higher priority will execute first than other lower priority job with the goal that client's interest can be fulfilled more completely.[28]

## 6. The Proposed Work

To contract the makespan time of jobs on cloud computing systems, a high level performance, enhanced Hyper-Heuristic Scheduling Algorithm is proposed. This Algorithm combines two low-level scheduling algorithms i.e. FIFO and Max-Min to initiate ideal scheduling solutions with minimized calculation time. From the pool of candidate one algorithm is choose as heuristic algorithm. Two operators are utilized i.e. diversity detection operator that automatically figure out which algorithm is picked and perturbation operator to optimize the solutions created by each of these algorithms to further enhance makespan time. The elementary idea of Hyper-heuristic is to use "one and one and only "heuristic algorithm at each iteration, in order to keep up high search diversity to construct the possibility of finding better solutions at later cycles while not growing the calculation time.

For the further improvement in the performance of HHSA approach, in terms of lower makespan time, A Load –Balancing Technique and Resource-Aware Scheduling Algorithm is applied.

### Task-Based System Load Balancing Method:

Load balance of a virtual machines is achieved by first mapping tasks to VM's and then all the VM to host resources, using the Task-Based System Load Balancing method. This algorithm ensures the system load balancing through only transferring extra tasks from an overloaded VM instead of migrating the entire overloaded VM. The loads are formulated as:

$$\text{VMload} = \text{BW} + \text{RAM} + \text{MIPS} \quad (1)$$

$$\text{LowloadedVM} = \text{maxcap} - (\text{load}/\text{capacit}) \quad (2)$$

$$\text{OverloadedVM} = (\text{load}/\text{capacit}) - \text{maxcap} \quad (3)$$

### **RASA (Resource-Aware-Scheduling-Algorithm):**

Resource utilization is achieved by using RASA a combined approach of Max-Min and Min-Min algorithms) to further optimize the resources in terms of accuracy and efficiency. To achieve this, it first estimates the completion time of the tasks on each of the available cloud resources and then applies the Max-Min and Min-Min algorithms, alternatively. Small tasks are executed by using Min-Min strategy before the large ones. To avoid delays in the execution of large tasks to support concurrency in the execution of large and small tasks Max-Min strategy is used.

### **Pseudo code for Improved HHSA procedure:**

1. Setup the parameters over the search space.
2. Input the job scheduling problem.
3. Initialize the population of solutions  
 $y = \{y_1, y_2 \dots y_n\}$
4. Random selection of low-level heuristic algorithm  
 $H_j$  from the pool of candidate.
5. Until Termination criterion is met.
6. Update the populations of solutions  $x$  by using the selected algorithm  $H_j$ .
7.  $E_1 =$  Diversity detection ( $y$ ).
8. If  $\psi(H_j, E_1)$
9. Randomly select the new  $H_j$ .
10.  $y =$  Perturb ( $y$ )
11. end
12. end
13. Output the optimal solution of HHSA.
14. Configure jobs into Datacenter and VM.
15. Update the status of Jobs.
16. Collect the VM status.
17. Calculate the load  
 $VM\ Load = Bandwidth + RAM + MIPS$
18. Analyze the balanced and Imbalanced load by using the resources of VM.
19. Analyze the priority of VM's Load.
20. Schedule the cloudlets.
21. Allocate the jobs to VM.
22. Output the best optimal solution as final solution.

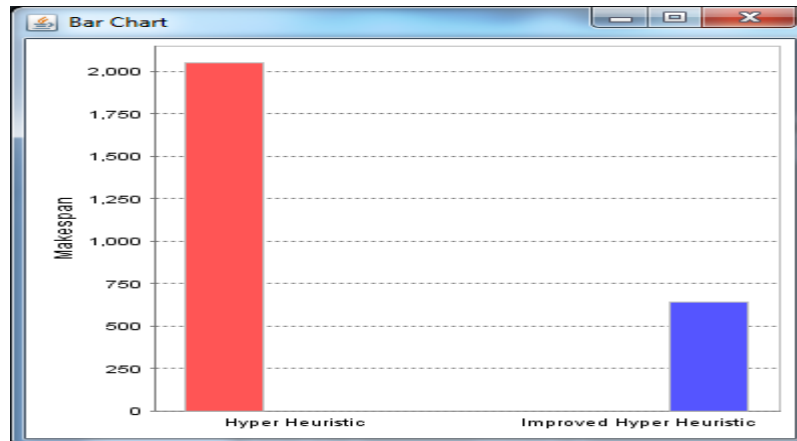
## **7. Experiments and Results**

### **7.1. Problem Formulation**

The objective of this project is applying the best solution method in the job scheduling problem to locate the ideal solution for the NP-hard problems. For too large and complex problems conventional scheduling strategies are not feasible to perform and to get ideal solutions in a reasonable time. A proficient scheduling algorithm is required for cloud computing frameworks called Enhanced Hyper-Heuristic Scheduling algorithm. An integration of two or more heuristic algorithms to maintain high search diversity, decrease makespan time and in order to keep up high search diversity to build the possibility of discovering better solutions at later cycles while not growing the calculation time.

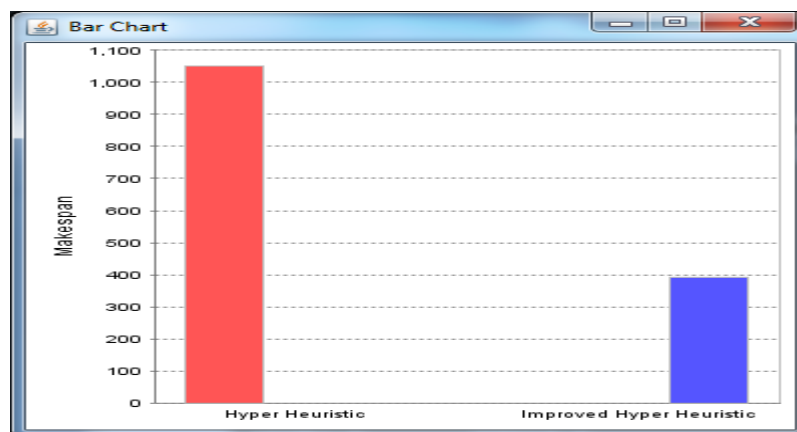
## 7.2. Simulation and Analyses

While going for implementation part, we solve the JSP by creating a Virtual Cloud Environment for the inputs. We take total Makespan time of jobs as the key factors on the basis of which the results of HHSA and Improved HHSA are shown and compared. In order, to validate the effectiveness of the proposed system, simulation experiments are performed on the platform of NetBeans IDE and CloudSim. PhpMyAdmin is used for creating database and WampServer is used for the server connectivity. The makespan time results have being illustrated in the following figures Fig.3, Fig.4 and Fig.5. And comparison between HHSA and Improved HHSA has been made in Fig.6.



**Figure 3. Makespan Time for HHSA(2050ms) and Improved HHSA (690ms) for Dataset [5,3,4,6]**

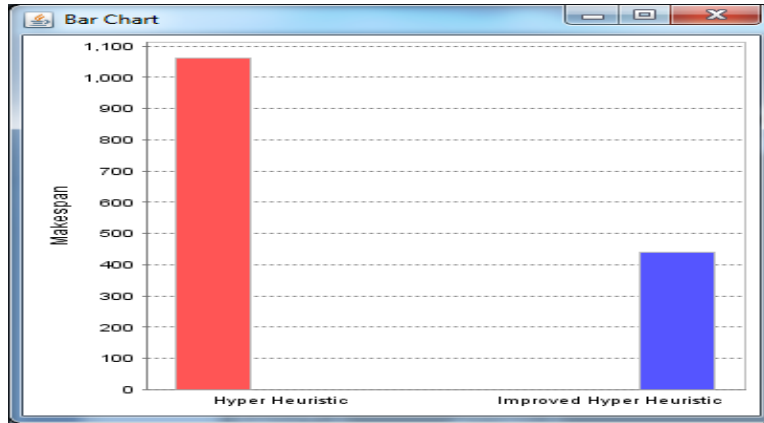
In the above figure , makespan time i.e. the total length of the schedule is displayed for the Hyper-Heuristic Scheduling Algorithm and Improved Hyper-Heuristic Scheduling Algorithm for the dataset taken [5,3,4,6] here 5 are the datacenters, 3 are brokers ,4 are virtual machines and 6 are cloudlets.



**Figure 4. Makespan Time for HHSA(1050ms) and Improved HHSA (389ms) for Dataset [6,4,7,10]**

In the above figure , makespan time i.e. the total length of the schedule is displayed for the Hyper-Heuristic Scheduling Algorithm and Improved Hyper-Heuristic Scheduling Algorithm for the dataset taken [6,4,7,10] here 6 are the datacenters, 4 are brokers ,7 are virtual machines and 10 are cloudlets.

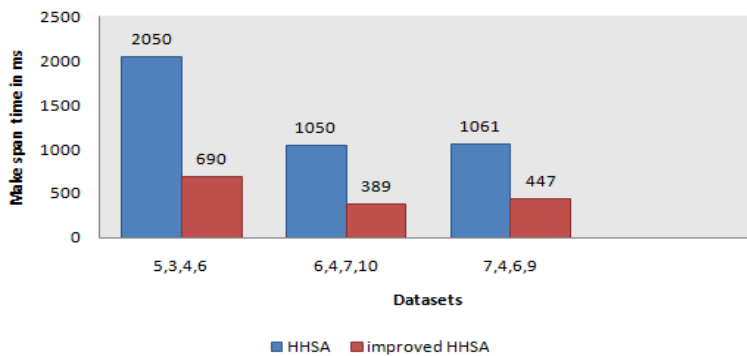




**Figure 5. Makespan Time for HHSA (1061ms) and Improved HHSA (447ms) for Dataset [7,4,6,9]**

In the above figure , makespan time i.e. the total length of the schedule is displayed for the Hyper-Heuristic Scheduling Algorithm and Improved Hyper-Heuristic Scheduling Algorithm for the dataset taken [7,4,6,9] here 7 are the datacenters, 4 are brokers ,6 are virtual machines and 9 are cloudlets.

### Comparison b/w HHSA and Improved HHSA



**Figure 6. Comparison for HHSA and Improved HHSA**

In the above figure, comparison bar chart of makespan time is displayed for HHSA and Improved HHSA for all the three datasets i.e. [5,3,4,6] ; [6,4,7,10] and [7,4,6,9].

**Table 1. Makespan Time Output of HHSA and Improved HHSA**

Virtual cloud Environment (No. of)				Makespan Performance Time (mili sec)	
DCC	Broker	Virtual Machine	Jobs	HHSA	Improved HHSA
5	3	4	6	2050	690
6	4	7	10	1050	389
7	4	6	9	1061	447

Table 1 shows the comparison of Improved Hyper-Heuristic Scheduling Algorithm with the Hyper-Heuristic Scheduling Algorithm in terms of least optimized makespan

time , by taking dataset [5,6,7] of data centers, dataset [3,4,4] of brokers , dataset [4,7,6] of virtual machines and dataset [6,10,9] of jobs as inputs to virtual environment for solving job scheduling problem (JSP).

## 8. Conclusion and Future Scope

In this paper an effective adjustment of the high –level heuristic for a JSP to minimize the aggregate make span time of given arrangement is displayed. Load of virtual machines are balanced through directly transferring the extra tasks from the overloaded VM's instead of migrating the entire overload VM by using the Task Based System Load Balancing Method, and to increase the efficiency and accuracy of resource utilization, Resource-Aware-scheduling algorithm is used. The simulation results shows that The Improved Hyper-Heuristic Scheduling algorithm reaches its objective: compared with the Hyper-Heuristic Scheduling algorithm, it deeply reduces the total makespan time of jobs, increase the overall performance of the whole system of Cloud computing environment. The future extension is to comprehend the kind of job failures with the expectation of enhancing the dependability of cloud base from the cloud suppliers view point.

## Acknowledgments

The authors would like to thank professors of CTITR, (Maqsudan) Jalandhar for suggestions and support on this paper.

## References

- [1] J.Zhang, X.Hu, X.Tan, J.H.Zhong, and Q.Huang," Implementation of an Ant Colony Optimization Technique for Job Scheduling Problem ", Transc. Of the Institute of Measurement and Control, (2006), pp. 93-108.
- [2] I. Foster, Y.Zhao, I. Raicu, and S.Lu," Cloud Computing and Grid Computing 360 –degree compared," Proceedings of the Grid Computing Environments Workshop, (2008), pp. 1-10.
- [3] S.Parsa, R.E.Maleki," RASA: A new Task Scheduling Algorithm in Grid Environment," World Applied Sciences Journal 7, (2009), pp.152-160.
- [4] Y.Fang, F.Wang and J.Ge," A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing," Springer – Verlag Berlin Heidelberg, (2010), pp.271-277.
- [5] G. Ming , H. Li , " An Improved Algorithm Based on Max- Min for cloud Task Scheduling," Tunnam University, China , (2011).
- [6] N. Kaur, T.S. Aulakh, and R.S. Cheema , " Comparisons of workflow scheduling algorithms in cloud computing , " International Journal of Advanced Computer Science and Applications , vol.2 ,(2011), pp. 81-86.
- [7] H.Chang, X.Tang," A Load-Balance Based Resource-Scheduling Algorithm under Cloud Computing Environment," Springer –verlag Berlin Heidelberg, (2011), pp.85-90.
- [8] S.B. Zhan, H.Y.Huo," Improved PSO-based Task Scheduling Algorithm in cloud computing," In: Journal of Information and Computational Science, vol. 9, (2012), pp.3821-3829.
- [9] M.T.Aftab,M.Umer,R.Ahmad , "Job Scheduling and Worker Assignment Problem to Minimize Make span using Ant Colony Optimization Metaheuristic ",International Journal of Mechanical, aerospace, Industrial , Mechatronic and Manufacturing Engineering ,vol .6,(2012).
- [10] N.J.Kansal, I.Chana," Existing Load Balancing Techniques in Cloud Computing: A Systematic Review," Journal of Information Systems and Communication, vol.3, (2012), pp-87-91.
- [11] O.M. Elzeki, M. Z. Reshad," Improved Max-Min Algorithm in Cloud Computing," International Journal of Computer Applications, vol.50, (2012), pp. 22-27.
- [12] L. Huang , H. Chen , T. Hu , " Survey on Resource Allocation Policy and Job scheduling Algorithms of Cloud Computing 1, " Journal of Software, (2013) , pp. 480-487.
- [13] R.G. BabuKarthik, R.Raju, and P.Dhavachelvan," Hybrid Algorithm for job scheduling: Combining the benefits of ACO and Cuckoo Search" Advances in Computing and Information Technology. Springer Berlin Heidelberg, (2013), pp.479-490.
- [14] M.Katyal, A.Mishra," A Comparative Study of Load Balancing Algorithm in Cloud Computing Environment," International Journal of Distributed and Cloud Computing, vol.1, (2013), pp.5-14.
- [15] Y.Sahu, R.K.Pateriya," Cloud Computing Overview with Load Balancing Technique," International Journal of Computer Applications, vol.5, (2013), pp.40-44.

- [16] P. Kowsik , K. RajaKumari ,” A Comparative Study on Various Scheduling Algorithms in Cloud Environment , “ International Journal of Innovative Research in Computer and Communication Engineering , vol.2 , (2014).
- [17] K.M .Cho, P.W.Tsai, C.W. Tsai,” A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing,” Neural Computing and Applications, (2014).
- [18] C.W.Tsai,W.C.Huang,,” A Hyper-Heuristic scheduling Algorithm for Cloud,” IEEE Transactions on Cloud Computing , vol.2 , (2014), pp.236-250.
- [19] C.W. Tsai and J. Rodrigues, “Metaheuristic Scheduling for Cloud: A Survey,” IEEE Systems Journal, vol.8, (2014), pp. 279-297.
- [20] L.Tripathy and R.R.Patra,” Scheduling in Cloud Computing,” International Journal on Cloud Computing: Services and Architecture, vol.4, (2014), pp-21-27.
- [21] A.Bhatia and R.Sharma, “An Analysis Report of Workflow Scheduling Algorithm for Cloud Environment,” International Journal of Computer Applications, vol.119, (2015), pp. 21-25.
- [22] B.T.Bini and S.Sindhu ,” Scheduling In Cloud Based On Hyper-Heuristics ,” International Journal for Research in Applied Science & Engineering Technology , vol.3 , (2015), pp. 380-383.
- [23] B. Kanani and B. Maniyar,” Review on Max-min Task Scheduling Algorithm for Cloud Computing ,” Journal of Emerging Technologies and Innovative Research ,vol. 2 , (2015) , pp. 781-784.
- [24] S. Selvarani , G. Sadhasivam ,” Improved Cost –based algorithm for task scheduling in cloud computing ,” Proceedings of IEEE International Conference on Computational Intelligence and Computing Research , (2010), pp. 1-5.
- [25] S. Pandey , L. Wu , S.M. Guru and R.Buyya ,” A Particle Swarm Optimization – based Heuristic for scheduling Workflow Applications in cloud computing environments ,” Advanced Information Networking and Applications (AINA) ,24 th IEEE International Conference , (2010).
- [26] A.Bala, I. Chana,” A survey of various workflow scheduling algorithms in cloud environment,” Proceedings of the National conference on Information and Communication Technology, (2011), pp. 26-30.
- [27] H.Chen,F.Wang,N.Helian and G.Akanmu ,” User-Priority Guided Min-Min Scheduling Algorithm for Load Balancing in Cloud Computing ,” IEEE Parallel Computing Technologies (PARCOMPTECH) National Conference , (2013), pp.1-8.
- [28] A.Agarwal, G. Manisha ,R.N.Milind and S.S.Shylaja,” A Survey of Cloud Based Load Balancing Techniques ,” Proceedings of International Conference on Electrical , Electronics , Computer Science and Mechanical Engg., (2014) , pp.9-13.
- [29] Y.Mao, X.Chen and X.Li ,” Max-Min Task Scheduling algorithm for Load Balance in Cloud Computing ,” Proceedings of International Conference on Computer Science and Information Technology , advances in Intelligent Systems and Computing , (2014) , pp.457-465.

## Authors



**Geetinder Kaur** , Research Scholar , Department of Computer Science and Engineering , CT Institute of Technology & Research , Jalandhar,India , geetinder810@gmail.com

