Resource Planning and Allocation in Distributed Cloud Networks using Voids in Scheduled Intervals

V. Murali Mohan, K.V.V. Satyanarayana

Abstract: ***The significant objective of research in the distributed cloud networks is resource planning and allocation using voids in scheduled intervals, which grabbed several researchers' attention in contemporary literature. Minimum failures of resource scheduling, completion of a robust task and fair usage of the resource were the important parameters of resource scheduling schemes. Therefore, this paper projected scalable-resource scheduling method for distributed CC environments, which aimed to attain the metrics of scheduling. The projected method is known as "Resource Planning and Allocation in Distributed Cloud Networks using Voids in Scheduled Intervals (RPA-DCN)," where resources are scheduled towards the corresponding task so that, the optimum using of idle time of resource is attained. The projected method performs scheduling in sequential order, and they were allocation of optimum resource when no single resource is identified to allocate, then optimum manifold idle resources are allocated with a considerable filling of schedule intervals. This paper discusses (a) pre-requisite of resource scheduling schemes, (b) current scheduling methods found in recent literature, (c) methods & materials utilized and method of projected resource scheduling scheme (d) and its significance over other standard methods. The analysis of performance for the proposal conducted by metrics cross-validation such as load vs. loss, optimality of task completion & process-overhead in the resource-scheduling. Here, the simulation outcomes exhibit that the projected method is robust and scalable under modified metrics.***

Keywords: ***RPA-DCN, Cloud Computing, Hadoop cluster, DPSA, MGBA.***

1. INTRODUCTION

With the vigorous developments in the IT infrastructure solutions, essential systems such as solutions of Cloud Computing (CC) has modified business dynamics. There were prominent advantages, which envisioned in cloud-based solutions application. The number of improvements occurred in the CC environment could be phenomenal. The work [1], [2] presents that globally, there were several companies relying on solutions based on the cloud for their managerial networks systems.

A considerable amount of inputs produced from the intense information sources and organization of network is becoming a crux for the organization related to decision-making. In instance, application of solutions of big data and such other business analytic methods requires maximum processing data speed in a continuous way and is based on real-time [3].

The significant benefit related to the resource capabilities of computing and easy data accessibility were the important parameters, which lead to the maximum demand of solutions based on the cloud. Besides, on the other hand, with enhancing demand for the solutions based on cloud, different levels of services based on the cloud are providing. The work [4] presents that in an instance, public, hybrid & private cloud services cater towards distinct demands level for the space of storage.

Even though CC inherent process is more parallel to the grid computing procedure and the cluster-computing methods, here, still virtualization usage for the management of the resource is definitely as prominent development [5], which might assist in the effective services also in instance of utility methods [6]. The work [2] presents that accessibility & computing towards the data became simpler and also managing operational costs of IT infrastructure is lessened by a deliberate percentage that might not be adaptable in the conventional environment of computing.

Different researches have conducted on CC solutions and their variation in a vivid range of information management systems. With enhancing CC trends, the virtualization parameters are attaining more significance. In service stream utility based on the market such as CC is very prominent for the concentration of resources scheduling in an optimum manner. It assures possible result & win-win case for the important stakeholders in the procedure [7].

Mainly, the most significant issues, which attain the stakeholder's attention in CC service SLA, could be regarding the optimum resources scheduling, since it could be one of the important deliverables, which decides effectiveness and weight of solution based on the cloud [8].

The outcomes of CC complexities are of load factors under-estimating, resources provision & planning unrealistic. In managing the optimal outcome dimension; it addresses effective power pre-requisites, assuring operational effectiveness of methods are significant and became an important area related to QoS pre-requisites [9].

Recognizing the appropriate resources, which are required for load scheduling within specified time and assuring continuous processing of data according to SLAs, were the integral requirements in managing solutions based on the cloud more efficiently. On another dimension, the amount of the resources availed in offering service is minimum related

Revised Manuscript Received on August 19, 2019.

V. Murali Mohan, Research Scholar, Department of Computer Science and Engineering, KoneruLakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India. Orcid Id:0000-0001-8310-0565 (muralimohan.klu@gmail.com)

K.V.V. Satyanarayana, Professor, Department of Computer Science and Engineering, KoneruLakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India. Orcid Id: 0000-0001-5115-5545, (kopparti@kluniversity.in)

to workload. It also assures that pre-requisite service quality levels are also another parameter. To assure the effective scheduling of resources, utilizing quality tools and methods for mapping workload resource is very pre-requisite. The other important parameter, which plays a crucial role in the procedure of resource scheduling, is regarding detection of suitable & appropriate workload, which might facilitate in the scheduling manifold workloads. It can address the pre-requisites of QoS, such as the utilization of CPU, management of process, reliability, and server's up-time circumstance and security [10]. By Deliberating such parameters, it might be reported that scheduling of resource has maximum importance level regardless of whether it could be homogenous/ heterogeneous workloads [11].

The work [2], [12] presents that wide-range of research has taken in the domain of CC. Also, still, several researchers are concentrating on the solutions of CC, optimal resource scheduling parameters, which lead to efficient cloud environment utilization.

The work [13], [14] presents that non-deterministic polynomial optimization issue is the main problem in the procedure of resource scheduling of cloud, since NP-hard problem result to increase the utilization by distinct variables, which are depending on deterministic algorithm aimed at exhaustive-search.

Reduction in the dimension has envisioned in an instance of algorithms, which address comparatively routine type of issues of cloud scheduling. The problems compounded while there could be a definite type of ambition, complexities & proliferation, which envisioned in CC.

The work [15] presents that adapting evolutionary algorithm of computation for managing resource scheduling in an environment of the cloud has attained researcher's attention since the model might offer a quality solution towards intricate problems in scheduling the resource [15]. Here, the effectiveness of such resource scheduling systems at diverse grid computing ranges has attained prominence in an instance of the resource-scheduling [16].

Surveys are conducted on productive scheduling in CC [9], [17], [18], and definite models such as interconnected CC [4], represents that in spite of improvements, which occurred, still there could be scope for the development.

By Deliberating evolving developments and research scope, the projected study is regarding the assessment of contemporary solutions in cloud-resource scheduling and opportunity for enhancing solutions.

The projected research has concentrated on the following parameters:

- Assessment of taxonomic structure by concentration on the hierarchy levels of scheduling.
- Examine the suggested schemes for managing cloud schedule by reviewing the cons & pros of such type of solutions.
- Concentrate on possible solutions, which might be modified at a distinct range of cloud resource scheduling hierarchy to attain optimum resource scheduling.
- Examine the challenges related to the possible direction of research, present solutions and distribution schedule

2. RELATED WORK

The existing methods associated with scheduling the resource in the CC environment are detained in this segment. The complete review of scheduling schemes associated with CC is discussed in the contribution [19]. The adaptive and dynamic based scheduling resource has conducted. The work [20] projected "community aware resource scheduling" method for enhancing average slowdown time of hob and increase the waiting time without previous real-time knowledge on the processing of distinct nodes performing in the decentralized scheduling way. Here, in this technique, initially, the job could be distributed and allotted to resources, where the job could reschedule when present resources were not capable of fulfilling the requirements of the job.

The work [21] explained the queuing theory based on a resource scheduling method to enhance the average time interval aimed at the dynamic non-interactive workload. There are three workload scheduling rules raised for separating appropriate jobs towards execution. The work [22] projected failure & power-aware resource scheduling method for the independent workloads in lessening the consumption of power and accomplish SLA. The proactive fault-tolerance methods based on the decision-making schemes have utilized for handling failures of systems towards controlling shared resources in an effective way by utilizing VM strategy. The work [23] presented that decision is made by feedbacks for scheduling resources with less contention. Here, this method depends on resources preemption from jobs scheduled, and it reschedules for attaining resource scheduling that is free of contention.

The work [24] projected resource scheduling method based on a Hadoop cluster for a computing arrival rate of job and its performance time in making the correct decision for optimal scheduling. Here, the Hadoop method contains a cluster that is a cluster of resources linked. Data in Hadoop method have structured into the files. Here, consumers submit tasks towards the system, while every job contains some of the tasks. Every task could be either reduce or map task. The algorithm deliberates minimum satisfaction in sharing the pre-requisites of overall consumers and fairness amid overall consumers in the system.

The work [24] projected algorithm of resource scheduling based on priority known as "DPSA (dynamic priority scheduling algorithm)" in CC, where scheduling issue of service request has solved. Here, in DSPA, the consumer requests were examined, classified & received based on their definite pre-requisites into units of tasks for directly scheduling on sufficient resources, and effective service is provided based on consumer request. Explained resource scheduling method based on priority and flexibility that applied in the platform of Xen virtualization. Using hypervisor monitoring for observing the requirements of the user and scheduling the resources based on pre-requisites, and it lessens the time of overhead. The work [25] projected dynamic resource allocation technique based on the virtualization for enhancing the utilization of the server. The

Retrieval Number: B11660882S819/2019©BEIESP
DOI:10.35940/ijrte.B1166.0882S819

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

algorithm of skewness has utilized for predicting disproportion in various dimensions of processor utilization by mitigation of hotspot. This method functions better in the migration of hotspot and balancing the load, yet live migration could not be possible.

The work [26] projected task scheduling rule based on QoS by deliberating task priority to lessen the time of completion. This policy of scheduling calculates the priority of task based on the attributes of the task for detecting the tasks precedence relation and is stored in the arranged task queue based on recognized priority and was also detected the performance time of each task and assigns the task towards the resources that performs a task in minimal time.

The optimization based on resource scheduling has conducted. The work [27] proposed algorithm resource scheduling based on the market for performing deadline workloads sensitive within the boundary and their budget in huge clusters of computing. The important objective of the algorithm could be to increase revenue by utilizing the assumptions of Bayesian for the experiments on the job traces datacenter. The work [28] projected "hierarchical distributed loop self-resource scheduling policy" for the scientific implementations in balancing the load utilizing weighted autonomous scheduling method in a distinct environment of the cloud. To lessen the overhead of communication and the output data is distributed effectively. This policy of scheduling utilized two scientific implementations for proving effectiveness. The work [29] projected scheduling policy based on stochastic-integer programming known as "MGBA (Minimized Geometric Buch Berger Algorithm)" by deliberating the SLA to NP-hard issue in lessening the time of execution. The MGBA resolved the stochastic integer programming issue by utilizing a Grobner basis concept for solving the issue of resource scheduling.

The work [30] explained, "multi-objective based task-scheduling scheme" for workloads to lessen the time taken for search and overhead scheduling. The ordinal optimization model strategy has been utilized for fulfilling the pre-requisites of the virtual environment of cloud that contain distinct serves of divergent data-centers. This scheduling scheme lessens overhead scheduling.

Sequencing the applications based on the web in the data center based on the cloud are discussed in the placement controller application [31] & MUSE [32]. Here, MUSE duplicates web implementations deployed and utilizes communication algorithm in the frontend for serving every request to lessen servers infrequently. This method, known as the placement controller of application balances the load.

The work [33] proposed a method, which combines load remitting & server allocation for services based on a connection like the messenger of windows. The work [32], [31], [33] were not relied on virtual-machines. Therefore, they utilize a multi-tier structure for organizing the implementations and also utilize the dispatch algorithm. When compared to these methods, our implementation virtualizes the infrastructure application rationally and schedules virtual machines in the form of resources.

The CC context is coordination of implementations of optimum data tracking that also referred to in the form of data locality. Here, the work [34] presents cloud service known as Map-reduce, which often utilized in this framework of optimum data-tracking.

The work [35] introduced scheduling scheme known as Quincy, where tasks are scheduled under metrics such as maximization of data locality and resources allocation. The same contribution is detection in [36], which aimed in balancing the time taken for execution of data tracking procedure. The work [37] devised a scheduling algorithm under active priorities in attaining fair resources allocation.

The work [32], [33], [36], [37], [38], [39], [40], [41], presented identical less cost solutions that is resource scheduling scheme at application level. In contrast to contemporary methods noticed in the literature, the algorithm of resource scheduling has projected in this paper, which is tried and materialized successfully [42].

Another important parameter is that the method projected here could be the set of tasks composition with identical resource requirement in the form of window request. The knowledge attained from the contemporary review of literature could be the primary contribution, which studied resources scheduling among identical tasks in the form of 1 unit. Here, the theory of collecting identical tasks in the form of 1 window for optimizing the utilization of the resource is adapted in this paper.

3. RESOURCE SCHEDULING WITH SCHEDULE INTERVAL FILLING FOR CLOUD COMPUTING

The Resource Scheduling with Schedule Interval Filling (RPA-DCN) proposed in this manuscript functions as a frontend to Resource Allocation Controller. Initially, the set of similar tasks triggered pooled as a window. The schedule interval filling can define as the usage of the interval time between the pair of resource scheduled times in sequence. The scheduling strategy performs the search for an optimal resource for a given tasks window in a hierarchical order. The hierarchical order of the search for an optimal resource is as follows:

- A control frame respective to each triggered task window (hereafter referred as a window) carries the requirements such as expected resource, time to engage that resource, the size of the window, window arrival time and its completion time.
- The arrival time of the request window is the aggregate value of time required to reach the resource allocation controller, the volume of time required to process a control frame (see Eq1).

$$\tau_{w_i} = t_{cf}(w_i) + p_{cf}(w_i) + t_{w_i} + \beta \quad \text{(Eq1)}$$

// the aggregate value of arrival time $t_{cf}(w_i)$ of the control frame cf, process time $p_{cf}(w_i)$, time t_{w_i} required for the window to reach the resource allocation controller, and elapsed threshold β defined.

The requirements and priorities obtained from the control frame, the proposed RPA-DCN schedules the resources, which explored in the following sections.

RPA-DCN Scheduling Strategy

Resource Allocation Controller executes RPA-DCN to perform resource allocation to the window that represented by the control packet arrived, which is as follows:

The adaptable to the requirements and idle time of the resource that suits to accomplish the completion of the tasks window are two standards followed by a proposed resource scheduling strategy. RPA-DCN, upon failure to identify an individual resource that meets the scheduling criteria, then pools a minimal set of resources to meet scheduling criteria, if failed then selects one or more resources with maximal scheduling intervals (idle time between pair of schedule times in sequence) and schedule them to fulfill the requirements of the window to be arrived. If either of these cases succeeds, then segments the window into a minimum number of windows such that resource scheduling succeeds under specified factors. The resource allocation to the target window at scheduling intervals, which is the third level of the proposed scheduling hierarchy, is explored in the following steps.

- Schedules a resource to the windows w_k and w_l expected to arrive at different times, if available with scheduling interval $i_{w_k \to w_l}$, such that
  - $b(i_{w_k \to w_l}) < \tau_{w_i}$ //begin time $b(i_{w_k \to w_l})$ of the scheduling interval $i_{w_k \to w_l}$ is less than the arrival time τ_{w_i} of the window w_i
  - $e(i_{w_k \to w_l}) > (c_{w_i} + \beta)$ // end time $e(i_{w_k \to w_l})$ of the interval $i_{w_k \to w_l}$ is greater than the completion time $(c_{w_i} + \beta)$ of the tasks in the window w_i, here β is the elapsed completion time offset defined.
- If failed to meet the above criteria, then selects a minimal set of resources, which already scheduled and having scheduling intervals such that,
  - Scheduling Interval begins time of all the selected compatible resources are identical and less than the arrival time of the window, and a sum of the scheduling intervals is greater than the completion time of the tasks found in a given window. If found pools all the selected resources and schedules to the target window.
- If failed to meet the above criteria, then segments the target window into two and executes RPA-DCN on each window.

The strategy of RPA-DCN has explored with mathematical notations and algorithm flow in the following Sections.

Pseudo representation of scheduling algorithm

$RS(w_i, cf)$ Begin

1. Let cf be the control frame of the respective window w_i,
2. $\bar{r} \leftarrow \phi$ // vector of optimal resources, which is empty initially.
3. $\bar{r} = RPA - DCN(cf, R)$ //invoking a method that tracks optimal resource under three levels of RPA-DCN that meets the criteria of requirements found in cf respective to the window w_i, here R is the set of resources available
4. If ($\bar{r} \neq \phi$) Begin
   a. Partition the w_i into two windows $\{\overleftarrow{w_i}, \overrightarrow{w_i}\}$ and apply RPA-DCN on each such that control frame cf represents both windows.
   b. $RS\left(\overleftarrow{w_i}, cf\right)$ // invoking the main method for the first part of the window
   c. $RS\left(\overrightarrow{w_i}, cf\right)$ // invoking the main method for the second part of the window
5. End // of line 4
6. Else Begin // of line 4
   a. if the size of the $\bar{r}$ is one then schedules that resource
   b. Else pools the all resources as one unit and schedules to the window w_i represented cf .
   c. Exit
7. End // of a condition inline 6
8. End // of the function

Pseudo representation of optimal resource selection algorithm

$RPA - DCN(cf, R)$ Begin

1. $er \leftarrow \phi$ // an empty vector that contains eligible resources identified during the process
2. $\bar{r} \leftarrow \phi$ //, an empty vector contains optimal resources to schedule found in the process.
3. For-each $\{r \exists r \in R\}$ begin
4. If $\left\{ \begin{matrix} (b(nit_r) + \alpha) < (\tau_{w_i}) \wedge \\ (e(nit_r) - b(nit_r)) > c_{w_i} + \beta) \end{matrix} \right\}$ Begin// The begin of the next idle time frame $b(nit_r)$ that summed up with elapsed threshold α defined is less than the arrival time τ_{w_i} . Besides, the total idle time of the resource (which is the absolute difference between end and beginning of the idle time) r is greater than the expected completion time c_{w_i} of the given task window that summed up with completion elapsed offset β defined.
5. $\bar{r} \leftarrow r$
6. Break the loop // in line 3.

7. End //of the condition in line 4
8. End of the loop in line 3
9. If $\left(\bar{r}\, is\, not\, empty\right)$ return $\bar{r}$ // completion of the method at first level of the hierarchy
10. For-each $\{r \exists r \in R\}$ begin
11. if $\left((b(nit_r)+\alpha)<(\tau_{w_i})\right)$ begin
    a. $er \leftarrow r$
12. End // of line 11
13. End // of line 10
14. If $\left(er\, is\, not\, empty\right)$ Begin
    a. Sort the er as $\overline{er}$ in descending order of their idle time
    b. $snit=0$ // aggregate of the idle times observed for selected resources in $\bar{r}$
15. For each $\left\{r \exists r \in \overline{er}\right\}$ Begin
    a. $\bar{r} \leftarrow r$
    b. $snit{+}{=}\left(e(nit_r)-b(nit_r)\right)$
    c. If $\left(snit>(c_{w_i}+\beta)\right)$ Begin
    d. Return $\bar{r}$ // completion of the method at the second level of the hierarchy
16. End //of line 15
17. End// of line 14
18. $er \leftarrow \phi$ // empty the vector er
19. For-each $\{r \exists r \in R\}$ begin
20. if $\left((b(si_r)+\alpha)<(\tau_{w_i})\right)$ begin // if the begin of the schedule interval si_r of the resource r is less than the arrival time τ_{w_i} of the window w_i
    a. $er \leftarrow r$
21. End // of line 19
22. End//line 18
23. If $\left(er\, is\, not\, empty\right)$ Begin
    a. Sort the er as $\overline{er}$ in descending order of their scheduled intervals
24. For-each $\left\{r \exists r \in \overline{er}\right\}$ Begin
    a. $\bar{r} \leftarrow r$
    b. $snit{+}{=}\left(e(si_r)-b(si_r)\right)$
    c. If $\left(snit>(c_{w_i}+\beta)\right)$ Begin
    d. Return $\bar{r}$ // completion of the method at the third level of the hierarchy
25. End //of line 24
26. End// of line 23
27. Return $\bar{r}$
28. End //of the Method

To perform the resource scheduling, RPA-DCN initiates to track possible optimal resource, if failed, then attempts to segment the window into two windows and performs the scheduling of the resources for each partition. Here the process of segmenting the window is on-demand; hence, the segmentation process claims minimal overhead.

4. EXPERIMENTAL SETUP AND EMPIRICAL ANALYSIS& RESULTS

The proposed method RPA-DCN performance evaluated by the experimental study has executed on the planet-lab [43] utilized for simulating distributed cloud networks environment with tasks stream, and manifold resources are virtualized rationally. The RPA-DCN performance evaluated by the task of metrics load vs. failure of resource allocation, a load of task vs. optimal task completion processes overhead scheduling. Here, the values noticed for the metrics from experimental study of the proposed method RPA-DCN is compared with the outcomes noticed for the metrics of other standard methods known as “A Framework for Resource Allocation Strategies in Cloud Computing Environment (FRAS)” [40] & “analytic hierarchy process for task scheduling and resource allocation in cloud computing environment (AHP)” [41]. Here, the parameters utilized are in the following (Table 1).

Table 1: Utilization of Parameters in Simulation

Count of users	125
Count of Resources with their virtualizations	155
The number of tasks included forming Request window	11 - 25 identical tasks
The count of million instructions for a request window	0.1 - 1
Number of priorities of task	5 - 15
elapsed values of the threshold used	0.05% of the actual

The same tasks are combined in the form of a window that is in a range of 11-25 tasks in every window. Here, the projected RPA-DCN is applied in the java & deployed in the form of simulation frontend. The scheduled resources and status of tasks completion logged besides with the execution flow of RPA-DCN. The logs execution flow utilized for predicting process-overhead and scheduled resources logs and state of tasks completion utilized for evaluating tasks load vs. failures of resource allocation and optimal tasks completion. The outcomes exhibited for these metrics at distinct tasks load are compared with outcomes attained from other existing methods FRAS & AHP. Here, comparison of the load vs. failures of resource scheduling noticed for the RPA-DCN, FRAS & AHP are examined and depicted in the figure-1 in the form of line-chart. It concluded that the projected method could be 39%, 28% of the scheduling failures are lessened when compared to AHP & FRAS in respective order.

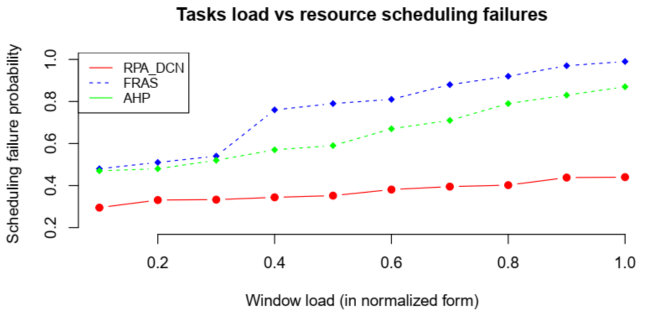

Figure 1: Request Load window versus failures of resource allocation

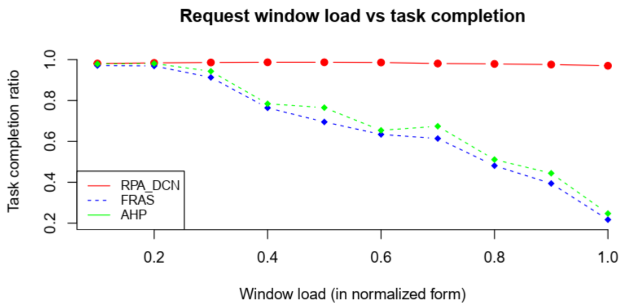

Figure 2: Request Load window versus optimal task completion

Table 2: Failure of Resource Scheduling ratio in averse to window load in the form of instructions set in millions

	FRAS	AHP	RPA-DCN
0.1	0.48	0.47	0.295
0.2	0.51	0.48	0.331
0.3	0.54	0.52	0.333
0.4	0.76	0.57	0.344
0.5	0.79	0.59	0.352
0.6	0.81	0.67	0.381
0.7	0.88	0.71	0.395
0.8	0.92	0.79	0.402
0.9	0.97	0.83	0.438
1	0.99	0.87	0.44

The optimal task completion noticed for the RPA-DCN, and the other two methods are compared and examined in figure 2. Here, comparison of optimal task completion noticed for entire these methods show that RPA-DCN is enhancing the optimal task completion by 28%, 31% respectively for the AHP & FRAS.

The processing overhead noticed in averse to request load window (Figure 3) is exhibiting in the form of linear in an instance of RPA-DCN method, while in the other two instances, process-overhead could be non-linear.

Table 3: Ratio of Task completion in averse to window load in the form of instructions set in millions

	FRAS	AHP	RPA-DCN
0.1	0.971	0.979	0.981
0.2	0.969	0.981	0.984
0.3	0.913	0.943	0.986
0.4	0.764	0.784	0.987
0.5	0.695	0.765	0.987
0.6	0.634	0.654	0.986
0.7	0.614	0.674	0.981
0.8	0.481	0.511	0.979
0.9	0.394	0.444	0.976
1	0.217	0.247	0.97

Figure 1 presents the Ratio of loss of request window in averse to request load window. Request load window is normalized towards value between zero & 1 that is the count of tasks pooled in the form of a window for a time unit. Here, the simulation study signifying that RPA-DCN prominently defused the loss of window when compared with the other two methods (table 2 & figure 1). Therefore, high task achievement noticed for RPA-DCN (Table 3 & Figure 2). Here, the conditional performance of hierarchical levels orders followed by the RPA-DCN and resources allocation towards tasks pool and also pooling context more than one resource for fulfilling the requirement of window tasks shown process-overhead in the form of linear (Figure 3).

Table 4: Ratio of process-overhead noticed in averse to window load in the form of instructions for a window in the millions

	FRAS	AHP	RPA-DCN
0.1	0.257	0.301	0.251
0.2	0.396	0.337	0.303
0.3	0.494	0.413	0.345
0.4	0.613	0.452	0.375
0.5	0.535	0.504	0.375
0.6	0.644	0.563	0.395
0.7	0.717	0.607	0.43
0.8	0.781	0.649	0.463
0.9	0.748	0.709	0.47
1	0.807	0.753	0.507

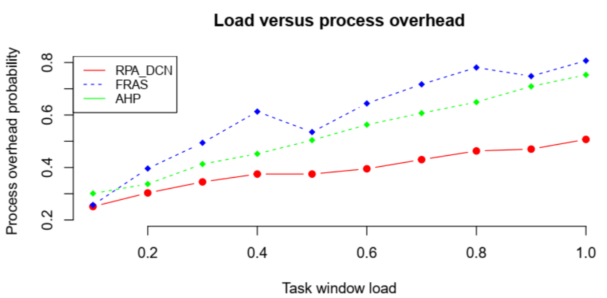

Figure 3: Process-overhead versus request load window

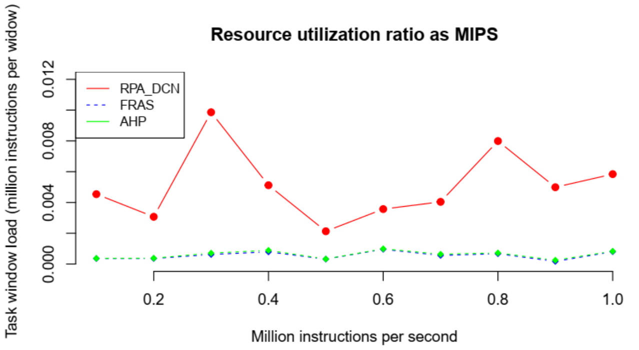

Figure 4: Ratio of Resource utilization in a million instructions for a second

The ratio of resource utilization is evaluated (Figure 4 & Table 5) for the projected method and the other two methods deliberated for the simulations. Here, the ratio of utilization could scale as manifold instructions for a second (MIPS) that are projected by the benchmark performance assessment corporation [44].

Table 5: Ratio of Resource Utilization

	FRAS	AHP	RPA-DCN
0.1	0.00036	0.00037	0.00454
0.2	0.00036	0.00038	0.00307
0.3	0.00062	0.00071	0.00986
0.4	0.00079	0.00089	0.00512
0.5	0.00032	0.00033	0.00213
0.6	0.00096	0.00099	0.00357
0.7	0.00056	0.00064	0.00404
0.8	0.00067	0.00072	0.00799
0.9	0.00017	0.00025	0.00499
1	0.0008	0.00083	0.00584

5. CONCLUSION

This paper proposed an algorithm of resource scheduling for distributed cloud network environment. Here, the projected method is RPA-DCN that considers the identical tasks in the form of 1 window unit and resources are scheduled as per the priority's availability of every task in window tasks and resource idle-time availability. Here, the proposed method schedules resources in a sequential manner. At the first hierarchy level, idle resource is tracked, which fulfills the tasks window priorities, when it is unsuccessful, then idle resources set is tracked and combines them for fulfilling the requirement, and when it failed, then traces for 1 or several resources with the compatible intervals of scheduling and combines them and later schedules towards corresponding window tasks. When it is unsuccessful to attain any of criteria, then tasks window is reformed, so that resources, which are available in an existing context, might fulfill the pre-requisites of window tasks. The simulation study exhibits that, the projected method is robust and scalable in scheduling the resource with the optimum completion time of the task and minimum failures of resource allocation. As the allocation scheme executed in a sequential way and performance of every hierarchy level is conditional, it recognized that computational-overhead is in the form of linear. Here, the maximum utilization of resource with minimum virtual machines & minimum computational-overhead noticed as the resources allotted for pooling of the tasks, as an alternative of individual-task. Further research direction could deliberate for composing resources towards distinct pooling jobs incorporated in every task in attaining optimum resource scheduling.

REFERENCES

1. Dragland, Asa. "Big Data–for better or worse." SINTEF. no. 22 May 2013. Web. 27 Oct (2013).
2. Toosi, Adel Nadjaran, Rodrigo N. Calheiros, and RajkumarBuyya. "Interconnected cloud computing environments: Challenges, taxonomy, and survey." ACM Computing Surveys (CSUR) 47.1 (2014): 7.
3. Zhang, Zehua, and Xuejie Zhang. "A load balancing mechanism based on ant colony and complex network theory in open cloud computing federation." Industrial Mechatronics and Automation (ICIMA), 2010 2nd International Conference on. Vol. 2. IEEE, 2010.
4. Xu, Fei, et al. "Managing performance overhead of virtual machines in cloud computing: A survey, state of the art, and future directions." Proceedings of the IEEE 102.1 (2014): 11-31.
5. Grandison, Tyrone, et al. "Towards a formal definition of a computing cloud." Services (services-1), 2010 6th World Congress on. IEEE, 2010.
6. Buyya, Rajkumar, et al. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility." Future Generation computer systems 25.6 (2009): 599-616.
7. Armbrust, Michael, et al. "A view of cloud computing." Communications of the ACM 53.4 (2010): 50-58.
8. Morshedlou, Hossein, and Mohammad Reza Meybodi. "Decreasing impact of sla violations: a proactive resource allocation approach for cloud computing environments." IEEE Transactions on Cloud Computing 2.2 (2014): 156-167.
9. Baliga, Jayant, et al. "Green cloud computing: Balancing energy in processing, storage, and transport." Proceedings of the IEEE 99.1 (2011): 149-167.
10. Singh, Sukhpal, and InderveerChana. "QoS-aware autonomic resource management in cloud computing: a systematic review." ACM Computing Surveys (CSUR) 48.3 (2016): 42.
11. Singh, Sukhpal, and InderveerChana. "Cloud resource provisioning: survey, status and future research directions." Knowledge and Information Systems 49.3 (2016): 1005-1069.
12. Heilig, Leonard, and Stefan Voß. "A scientometric analysis of cloud computing literature." IEEE Transactions on Cloud Computing 2.3 (2014): 266-278.
13. Genez, Thiago AL, Luiz F. Bittencourt, and Edmundo RM Madeira. "Workflow scheduling for SaaS/PaaS cloud providers considering two SLA levels." Network Operations and Management Symposium (NOMS), 2012 IEEE. IEEE, 2012.
14. Xu, Jielong, et al. "Enhancing Survivability in Virtualized Data Centers: A Service-Aware Approach." IEEE Journal on Selected Areas in Communications 12.31 (2013): 2610-2619.
15. Zhan, Zhi-Hui, et al. "Cloud computing resource scheduling and a survey of its evolutionary approaches." ACM Computing Surveys (CSUR) 47.4 (2015): 63.
16. Jennings, Brendan, and Rolf Stadler. "Resource management in clouds: Survey and research challenges." Journal of Network and Systems Management 23.3 (2015): 567-619.
17. Kaur, Harpreet, and Maninder Singh. "Review of various scheduling techniques in cloud computing." International

Journal of Networking & Parallel Computing 1.2 (2012).

18. Chawla, Yogita, and MansiBhonsle. "A study on scheduling methods in cloud computing." International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 1.3 (2012): 12-17.
19. Mohan, V. Murali, and K. V. V. Satyanarayana. "The Contemporary Affirmation of Taxonomy and Recent Literature on Workflow Scheduling and Management in Cloud Computing." Global Journal of Computer Science and Technology 16.1 (2016).
20. Huang, Ye, et al. "Exploring decentralized dynamic scheduling for grids and clouds using the community-aware scheduling algorithm." Future Generation Computer Systems 29.1 (2013): 402-415.
21. Le, Guan, KeXu, and Junde Song. "Dynamic resource provisioning and scheduling with deadline constraint in elastic cloud." Service Sciences (ICSS), 2013 International Conference on. IEEE, 2013.
22. Sampaio, Altino M., and Jorge G. Barbosa. "Dynamic power- and failure-aware cloud resources allocation for sets of independent tasks." Cloud Engineering (IC2E), 2013 IEEE International Conference on. IEEE, 2013.
23. Li, Jiayin, et al. "Adaptive resource allocation for preemptable jobs in cloud systems." Intelligent Systems Design and Applications (ISDA), 2010 10th International Conference on. IEEE, 2010.
24. Rasooli, Aysan, and Douglas G. Down. "An adaptive scheduling algorithm for dynamic heterogeneous Hadoop systems." Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research. IBM Corp., 2011.
25. Xiao, Zhen, Weijia Song, and Qi Chen. "Dynamic resource allocation using virtual machines for cloud computing environment." IEEE transactions on parallel and distributed systems 24.6 (2013): 1107-1117.
26. Wu, Xiaonian, et al. "A task scheduling algorithm based on QoS-driven in cloud computing." Procedia Computer Science 17 (2013): 1162-1169.
27. Jain, Navendu, et al. "Near-optimal scheduling mechanisms for deadline-sensitive jobs in large computing clusters." ACM Transactions on Parallel Computing 2.1 (2015): 3.
28. Han, Yiming, and Anthony T. Chronopoulos. "A hierarchical distributed loop self-scheduling scheme for cloud systems." Network Computing and Applications (NCA), 2013 12th IEEE International Symposium on. IEEE, 2013.
29. Li, Qiang. "Applying Stochastic Integer Programming to Optimization of Resource Scheduling in Cloud Computing." JNW 7.7 (2012): 1078-1084.
30. Zhang, Fan, et al. "Multi-objective scheduling of many tasks in cloud platforms." Future Generation Computer Systems 37 (2014): 309-320.
31. Tang, Chunqiang, et al. "A scalable application placement controller for enterprise data centers." Proceedings of the 16th international conference on World Wide Web. ACM, 2007.
32. Chase, Jeffrey S., et al. "Managing energy and server resources in hosting centers." ACM SIGOPS operating systems review 35.5 (2001): 103-116.
33. Chen, Gong, et al. "Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services." NSDI. Vol. 8. 2008.
34. Guo, Zhenhua, and Geoffrey Fox. "Improving mapreduce performance in heterogeneous network environments and resource utilization." Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on. IEEE, 2012.
35. Isard, Michael, et al. "Quincy: fair scheduling for distributed computing clusters." Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles. ACM, 2009.
36. Bobroff, Norman, AndrzejKochut, and Kirk Beaty. "Dynamic placement of virtual machines for managing sla violations." Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on. IEEE, 2007.
37. Das, Tathagata, et al. "LiteGreen: Saving Energy in Networked Desktops Using Virtualization." USENIX annual technical conference. 2010.
38. Savage, YuvrajAgarwal Stefan, and Rajesh Gupta. "Sleepserver: A software-only approach for reducing the energy consumption of pcs within enterprise environments." Power (KW) 100.150 (2010): 200.
39. Setty S, Braun B, Vu V, Blumberg AJ, Parno B, Walfish M. Proceedings of the 8th ACM European Conference on Computer Systems, EuroSys 2013. In8th ACM European Conference on Computer Systems, EuroSys 2013 2013.
40. Arfeen, M. Asad, Krzysztof Pawlikowski, and Andreas Willig. "A framework for resource allocation strategies in cloud computing environment." Computer Software and Applications Conference Workshops (COMPSACW), 2011 IEEE 35th Annual. IEEE, 2011.
41. Ergu, Daji, et al. "The analytic hierarchy process: task scheduling and resource allocation in cloud computing environment." The Journal of Supercomputing (2013): 1-14.
42. Mohan, V. Murali, and K. V. V. Satyanarayana. "Efficient task scheduling strategy towards Qos aware optimal resource utilization in cloud computing." Journal of Theoretical and Applied Information Technology 80.1 (2015): 152.
43. Chun, Brent, et al. "Planetlab: an overlay testbed for broad-coverage services." ACM SIGCOMM Computer Communication Review 33.3 (2003): 3-12.
44. Benchmarks, S. P. E. C. "Standard performance evaluation corporation." (2000).