Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EFFICIENT TASK SCHEDULING STRATEGY TOWARDS QOS AWARE OPTIMAL RESOURCE UTILIZATION IN CLOUD COMPUTING

[1]V. MURALI MOHAN, [2]K. V. V. SATYANARAYANA

[1, 2]Department of Computer Science & Engineering
Koneru Lakshmaiah Education Foundation (KLU), Vaddeswaram, Guntur Dist., A.P., India

E-mail: [1]muralimohan.klu@gmail.com , [2]kopparti@kluniversity.in

ABSTRACT

QoS (Quality of Service) aware task scheduling in cloud computing is a continuous practice due to the divergent scope of user needs. Henceforth the current research is moving in a direction to find optimal solutions for efficient task scheduling towards QoS aware resource utilization in cloud workflow management. Much of the existing solutions are specific to one or two QoS factors mainly task completion and bandwidth. According to the real-time practices, the QoS assessment by one or two factors is impractical. Moreover much of the existing approaches are delivering the computational complexity as $O(n^2)$, which is due to the magnification of the increment in number of tasks due to overwhelmed users and their requirements. In this context here we devised an explorative statistical approach, which is based on metrics called resource optimal value ($ropt$) and coupling between tasks (cbt), which enables to assess the optimal order of tasks to utilize desired cloud resource. The other key factor of the proposal is to stabilize the computational complexity to O(n*log(n)). The experiment results are indicating the significance of the proposed model towards scalable and robust QoS- aware task scheduling towards optimal utilization of the cloud resource.

Keywords: *Cloud Computing, Qos, Resource Scheduling, Multi Criteria Decision, Priority Scheduling, Task Management, ROPT, CBT*

1. INTRODUCTION:

Cloud computing model offers a convenient on-demand network access to computing resources accessed with a cloud interface by clients for performing various computing tasks remotely. The Cloud computing technology is based on distributed and virtualized design of shared pool of configurable resources. The cloud model offers resources [1] such as applications, processing power, memory, storage and networking over the net [2] and does not require the client to deploy any physical applications [3] or equipment. The cloud mechanism automates the process of resource scheduling, task scheduling and backup enabling rapid provisioning and release of resources. The management of the automatic resource allocation and tasks computation is done with cloud management software, reducing the work of the client or service provider. This approach of minimal service configuration and management assures reliable service as a result of decreased computation complexity and increased computing power and storage. The cloud service providers offer a pay as you go model to the users who pay the provider based on the resources used and avoid the hardware, software, and other services related huge costs.

Cloud computing web based services are, SaaS (Software as a Service), PaaS (Platform as a Service) and IaaS (Infrastructure as a Service). The SaaS model provides cloud versions of various applications and their functions, developed for the computing needs of the client. The PaaS model provides to the clients a computing platform for designing, developing and managing applications which comprises of, an OS, environment for executing software programs and database. The IaaS model provides on demand off-site or virtualized resources of servers, storage and networking.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The scheduling of resource is one of the important processes in resource management which makes different types of resources available to different consumer requirements. Task scheduling is another important process of mapping the user tasks to different resources practically and efficiently. The process of resource and task scheduling is based on the QoS requirements of the service provider as well as the users. Unlike other distributed computing models available, resource scheduling in cloud computing is based on several QoS parameters such as CPU speed, bandwidth availability, stability of the resources, memory etc. The QoS qualitative measures generally used are, completion time, latency, execution price, packet loss rate, throughput and reliability measure which decide the levels of service performance and user satisfaction [4].

As computing model, cloud computing is a fast developing technology, transforming every business model. The advantages of cloud computing is it enables cost effective automatic resource allocation and tasks execution with flexibility in deployment and scalability, where clients do not have the problems of managing the infrastructure rented.

Every year with increasing demand for computing, cloud computing faces challenges of migration, integration, implementation, execution, availability, reliability, governance, performance, and data security. A problem typical of cloud computing is availability of resources in terms of real time fluctuations in requirements. The major challenge [5] however is the implementation of tasks scheduling in cloud computing. Several researchers have focused their study to solve these existing scheduling problems. The research has developed a number of heuristic algorithms, such as Simulated Annealing, Particle Swarm Optimization, min-min, A*, ACO, Genetic Algorithm, etc. [6] and the demanding performance requirements has led to the design of many enhanced algorithms [7], [8] based on the existing algorithms.

1.1 Problem Description

Let consider a cloud-resource available to the set of interconnected tasks, which demands the utilization of the resource with max QoS objectives such as reliability, availability, cost, response time, iterative scope, completion time, which is significant issue for research and many of the researchers delivered considerable solutions for QoS aware task scheduling in recent years. Let consider the constraints explored below are influencing these set of tasks:

i. Task1 is independent and not influencing any other task
ii. The task2 and task3 should complete in the same order,
iii. Task 5 is dependent of Task 4, and conditionally recursive
iv. The task 5 is essentially need to wait for the instruction from task 4
v. The task 6 and task 7 are independent, can be initiated in parallel
vi. The conditional result of the task6 can revert the process to the task 2
vii. The task 8 is follower of the task 6 and task 7

The observations above listed are constraints to schedule the given tasks. The issue of scheduling task2 is sensitive since this and task1 should perform together. This constraint togetherness is one of the significant tasks scheduling constraint, which can be referred as “collectiveness” for further reference. The task5 is dependent of task4 and initiates conditionally, which is based on the result of the task4. Hence these two tasks are recursive on specific response of the task4 and this constraint can be referred further as “dependent”. The cloud-resources considered for task6 and task7 performs parallel, which we further referred as QoS connection constraint “parallel”. The cloud-resource selected for task is dependent of cloud-resources selected for task 6 and task 7. Another significant constraint can depicted from the activity of task6. The conditional result of the task6 rollbacks the job to task2. This constraint can be referred as “revert” for further references.

Any of these task scheduling constraints can be simplified by achieving coupling between tasks. This coupling between tasks is maximal and reliable, if both of the tasks are either in the expected order or at least in reliable order in scheduling.

Here in this paper we devised a new task scheduling metric called coupling between tasks (cbs). The model devised here in this paper is defining a statistical assessment strategy that assesses the robustness of the workflow scheduling under multiple QoS metrics and the coupling between tasks.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. RELATED WORK:

Task scheduling is one of the prime factors of QoS aware workflow management, since the QoS optimization during tasks scheduling to a specific resource elaborates the overall QoS scalability of the workflow management. The task scheduling is a significant issue for research since the era of distributed computing. This took new phase while scheduling in cloud environment. This is due to the sensibility of virtualization and magnified growth in usage.

The majority of the significant task scheduling strategies [9][10][11][12] found in literature are aiming to evaluate system design alternatives and capabilities aware task scheduling for large-scale data processing on accelerator-based distributed systems. These models are accelerated the MapReduce programming model by utilizing multiple types of computational accelerators. However, none of these focuses on multi-objectives of the QoS such as reducing the processing cost and completion time. Resource rescheduling [9] is identified to be one of the significant strategies that increase the reliability of job completion. The model devised here is a combination of physical and virtual infrastructure. Initially scheduling is done with physical infrastructure (Grid resources), if tasks are overwhelmed and physical infrastructure is alone not scalable, then virtualization (cloud) of the resources done and the same will be used to fulfill rescheduling the tasks in queue. The model devised here is also using the benchmarking heuristics to schedule or reschedule the tasks, which indicates the computational complexity is remaining NP-Hard. The other QoS aware Task scheduling strategy [10] is aimed QoS optimality under the mutually conflicting directions called consumer's and the provider's perspective. In regard to this the model in [10] is relied on genetic Algorithm to schedule the resource towards the tasks in queue. Here again this model also relied on a heuristic strategy, which leads to the computational complexity as NP-Complete [11][12].

The majority of the existing models are keen to bind the resources to the tasks towards QoS aware Workflow management. But, to the best of our knowledge, ordering tasks in queue towards utilizing a specific resource in the context of optimal QoS is overlooked in literature. Most of the existing models are dealing the issue of resource allocation towards task scheduling by heuristic strategies. These heuristic strategies are finding to be computationally complexes unless devised an efficient fitness evaluation strategy to limit the number of evolutions.

The Berger model based job scheduling strategy for cloud is found in [13], which was using dual fairness constraints as metric to schedule the tasks to the specific resource. In regard to this the said model is initially categorizing the tasks, which is based on their QoS preferences and then assess the resource fairness towards each group of tasks. Further the tasks has scheduled to the optimal resources. This is a motivated version of the tasks scheduling. The said model is keen to identify the fair relation between resource and specific group of tasks, which again ignoring the order of these tasks to use that specific resource. As we discussed in the problem description section the order of tasks to utilize a specific resource is also a significant towards improving QoS optimality of the workflow management

The research article [8] was devised a reliability-driven scheduling architecture, which aimed to define the order of tasks to be coupled to fulfill the job. Hence this model devised a reliability priority ranking (RRank) strategy to conclude the preference of the order of the tasks to be scheduled. This model [14] is another motivating strategy observed in literature, but it limited to identify the dependency and its impact to schedule the tasks, but overlooked other QoS constraints.

Though these models are significant in regard to the context obtained, but all of these models are considering one or two QoS factors to assess the scalability of the QoS aware workflow management, which is impractical in reality. Moreover the computational complexity of task scheduling is $O(n^2)$, since the increment in number of tasks in a workflow management is magnifying the evolution complexity of the heuristic models opted. Hence, we proposed an explorative statistical analysis model that is not specific to particular QoS factors and quantity of factors also considering the coupling between tasks. The said model assesses the impact of each task by the measuring their resource utilization optimality, which is based on multiple QoS factors.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. RESOURCE UTILIZATION OPTIMALITY ($ropt$) AND COUPLING BETWEEN TASKS (cbt): METRICS FOR QOS AWARE TASK SCHEDULING IN CLOUD COMPUTING

Let us consider a set of m cloud resources $S = \{s_1, s_2, s_3, ..., s_m\}$ and each resource s can be allocated to any individual task among available T, and T is
$T = \{ts_1 = \{t_{11}, t_{12}, ..., t_{1i}\},$
$ts_2 = \{t_{21}, t_{22}, ...t_{2j}\}, ...$
$ts_m = \{t_{m1}, t_{m2}, ...t_{mp}\}\}$
The tasks in set $ts_i = \{t_1, t_2, t_3, ...t_x\}$ are x number of tasks in queue for scheduling of resource s_i. Hence the scheduling of the cloud-resource to all tasks in queue, such that the resource utilization is optimal and robust. Thus, the objective of our proposal is which order of tasks in queue should be considered for each set of x tasks queued for a specific resource.

The order of tasks queued for a specific resource can influence the QoS. Hence, it is essential to pick optimal order of tasks scheduling. The explorative statistical model proposed here in this paper is based on the characteristics of tasks towards cloud-resource utilization, which are described as follows:

- A cloud-resource usage can be rated best when it allocates to a task, such that only that task is waiting for scheduling. But might fail to deliver the same optimality of cloud-resource usage while scheduling to the set of tasks in queue, which is in order of first come first serve basis during workflow scheduling.
- A cloud-resource usage by a specific task can be rated divergently with respect to its various QoS factors. As an example, cloud-resource s utilization by a task t can be best with respect to resource occupancy, but might be moderate in terms of cost, worst in the context of computational cost.
- The importance of the QoS factors might vary from scheduling to one task to other in a given set of tasks.

According to the characteristics of the cloud-resource utilization described, it is evident that the optimal cloud-resource allocation strategy (first come first serve) that applied to one independent task is not always the optimal towards set of tasks in queue of a specific workflow. But at the same moment, verification of the cloud-resource scheduling with all possible order of tasks in queue is also not scalable and robust. In regard to this the said explorative statistical model in its first stage, finds the optimality of the cloud resource towards each independent task in scheduling queue, which is based on primary QoS factor opted by each task in the queue. This process is labeled as resource scheduling optimality evaluation. Further tasks are ranked according to their optimal resource utilization state and will be used in the same order to finalize the order of tasks to utilize cloud-resource.

3.1 Exploration Of The Inputs Used In The Proposed Model

Let a set of cloud-resource utilization level QoS metrics $M = \{m_1, m_2, m_3, m_4, ...m_{|M|}\}$ of each task in the given tasks set
$T = \{ts_1 = \{t_{11}, t_{12}, ...t_{1i}\},$
$ts_2 = \{t_{21}, t_{22}, ...t_{2j}\}, ...$
$ts_m = \{t_{m1}, t_{m2}, ...t_{mp}\}\}$
Let $E = \{e_1, e_2, e_3, ..., e_n \forall [e_i : t_j \to t_{j+1}]\}$ be the set of n edges such that each edge connecting two tasks in workflow scheduling sequence. Let $CT = \{[t_i, t_j, t_k, ...], [t_x, t_y, t_z, ...], ...\}$ be the set of task-sets such that the connections between tasks of each tasks-set is influenced by any of the connection QoS constraint called reliant, analogous, revert or collectiveness. This can be defined as, each tasks-set of CT is expecting coupling between tasks. The term coupling can be defined under our proposed model as the cloud-resource used for these tasks should be of the expected order.

3.2 Evaluation Strategy Of Resource Scheduling Optimality Exploration

Let a QoS metric m_{opt} is said to be the prime metric prioritized to schedule the tasks in queue, which is used to rank the tasks towards resource scheduling. The QoS metrics of the cloud-resource utilization by a specific task can be classified as positive and negative metrics. The metrics that desire higher values are said to be positive metrics and the metrics that are optimal with minimal values are said to be negative metrics.

Henceforth the values of negative and positive metrics are normalized as follows:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

For each tasks set $[ts_i \exists ts_i \in T]$ Begin

For each task $[t_j \exists t_j \in ts_i]$ begin

For each metric $[m_k \exists m_k \in M_{s_j}]$ Begin // here M_{t_j} is the set metrics of resource utilization by task t_j

If m_k is positive metric then $norm(m_k) = 1 - \frac{1}{val(m_k)}$

Else if m_k is negative metric then $norm(m_k) = \frac{1}{val(m_k)}$

End

End

End

Then the set of cloud-resources possible to schedule to a specific task are ranked by their normalized values from maximum to minimum, such that each cloud-resource gets different rank for different metrics.

Further these ranks will be used as input to measure the resource scheduling optimality.

Let rank set of a task $[t_j \exists t_j \in ts_i \wedge ts_i \in T]$ is $rs(t_j) = [r(m_1), r(m_2), ...r(m_n)]$, then resource scheduling optimality $ropt$ of this task can be measured as follows.

$$\mu(t_j) = \left(\frac{\sum_{i=1}^{|M_{t_j}|} r(m_i \exists m_i \in M_{t_j})}{|M_{t_j}|} \right)$$

Here in this equation, $\mu(t_j)$ represents the mean of the all QoS metric ranks of the metrics M_{t_j}.

$$\sigma(t_j) = \sqrt{\frac{\sum_{k=1}^{|M_{t_j}|} \left(\mu(t_j) - \{ r(m_k) \exists m_k \in M_{t_j} \} \right)^2}{|M_{t_j}|}}$$

The above equation is calculating the standard deviation $\sigma(t_j)$ of the QOS metric ranks assigned to a task t_j.

$$g(t_j) = \frac{\sum_{k=1}^{|M_{t_j}|} \left(\mu(t_j) - \{ r(m_k) \exists m_k \in M_{t_j} \} \right)^3 \Big/ |M_{t_j}|}{(\sigma(t_j))^3}$$

$$g(t_j) = \sqrt{g(t_j) * g(t_j)}$$

Here in the equation $g(t_j)$ is skewness [15] of the QOS metric ranks distributed for task t_j

The skewness can be negative or positive, and near to 0 represents that the ranks of all QoS metrics are nearer, equal to 0 represents that the all QoS metrics with same rank.

In regard to get only positive skewness we find the square-root of the square of the resultant skewness.

According to ANOVA [16],

(i) The less skewness represents the uniform distribution of the ranking, which may be best, moderate or worst ranks (not the blend of these three).

(ii) The mean of the ranks distributed, reflects the centrality of the ranks distributed

(iii) The standard deviation of these ranks represents the how they deviated from each other.

(iv) The less variance between skewness, mean and standard deviation indicates the distribution of ranks with less skewness, less deviation and moderately average of near ranks. Hence the optimality of the cloud-resource utilization by task t_j can be measured as follows:

$$\mu(g(t_j), \mu(t_j), \sigma(t_j)) = \frac{g(t_j) + \mu(t_j) + \sigma(t_j)}{3}$$

Here in the above equation the mean $\mu(g(t_j), \mu(t_j), \sigma(t_j))$ of the resultant skewness ($g(t_j)$), mean ($\mu(t_j)$) and standard deviation ($\sigma(t_j)$ of cloud-resource t_j is measured.

$$\sigma^2(g(t_j), \mu(t_j), \sigma(t_j)) = \frac{\begin{bmatrix} (\mu(g(t_j), \mu(t_j), \sigma(t_j)) - g(t_j))^2 + \\ (\mu(g(t_j), \mu(t_j), \sigma(t_j)) - \mu(t_j))^2 + \\ (\mu(g(t_j), \mu(t_j), \sigma(t_j)) - \sigma(t_j))^2 \end{bmatrix}}{3}$$

$$ropt(t_j) = \frac{1}{\sigma^2(g(t_j), \mu(t_j), \sigma(t_j)) + 1}$$

Here in the above equation,

(i) $ropt(t_j)$ is indicating the resource usage optimality of the task t_j.

(ii) The resultant variance σ^2 is normalized to the value between 0 and 1, such that

Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

more variance gives less resource usage optimality of the task t_i.

(iii) To avoid the divided by zero error, we added 1 to the variance.

3.3 Measuring Coupling Between Tasks (cbt)

Let $Q = \{q_1, q_2, q_3, ..., q_z\}$ be the set of possible order of tasks towards specific resource utilization Then the coupling between tasks (cbt) of tasks order q_i of a resource to be scheduled s_i, which indicates the number of connections having coupling (expected order of connection formed between dependent tasks) against number of connections required coupling (see section 4.1**)**. Then the coupling between tasks $cbt(c_i)$ will be measured as follows

$$cbt(q_i) = \left[\sqrt{\frac{(\mu(cbt(q_i), cbtR) - cbt(q_i))^2 + (\mu(cbt(q_i), cbtR) - cbtR)^2}{2}} + 1\right]^{-1}$$

Here in the above equation:

(i) Let $cbtR$ be the coupling count that represents total number of edges between tasks with required order of tasks.

(ii) $cbt(c_i)$ is the coupling between tasks of the resource scheduling order q_i, which is measured by normalizing the standard deviation observed from $cbt(q_i)$ and $cbtR$ to $0 \le cbt(q_i) \le 1$

(iii) To avoid the divide by zero error, the resultant standard deviation is increased by 1.

(iv) $\mu(cbt(q_i), cbtR)$ is the mean of the $cbt(q_i)$ and $cbtR$

3.4 Measuring ($ropt$) of the Cloud-Resource Scheduling towards the Order of Tasks.

Then the overall resource scheduling optimality tasks order c_i will be measured as follows:

(i) Initially, the mean of fitness values of the cloud-resources involved in workflow scheduling is measured

$$\mu(q_i) = \frac{\sum_{j=1}^{|q_i|} \{ropt(t_j) \forall t_j \in q_i\}}{|q_i|}$$

- Here in the above equation $\mu(q_i)$ is the mean of the resource scheduling optimality of the cloud-resources involved in workflow scheduling

(ii) The standard deviation of the Inverse of the resource scheduling optimality of the cloud-resource in the order of tasks

$$\sigma(q_i) = \sqrt{\frac{\sum_{k=1}^{|q_i|} (\mu(q_i) - \{ropt(t_k) \exists t_k \in q_i\})^2}{|q_i|}}$$

- Here in the above equation $\sigma(q_i)$ indicates the standard deviation of the resource scheduling optimality distributed across the tasks in order q_i

(iii) The skewness of the $ropt$ of the tasks in order q_i

$$g(q_i) = \frac{\sum_{k=1}^{|q_i|} (\mu(q_i) - \{ropt(t_k) \exists t_k \in q_i\})^3 \Big/ |q_i|}{(\sigma(q_i))^3}$$

$$g(q_i) = \sqrt{g(q_i) * g(q_i)}$$

- Here in the above equation $g(q_i)$ is the skewness observed across the resource utilization optimality $ropt$ distributed over tasks in order q_i

(iv) Variance of the mean, standard deviation and skewness of the resource utilization optimality of the tasks in order q_i

$$\sigma^2(g(q_i), \mu(q_i), \sigma(q_i)) = \frac{\begin{bmatrix} (\mu(g(q_i), \mu(q_i), \sigma(q_i)) - g(q_j))^2 + \\ (\mu(g(q_i), \mu(q_i), \sigma(q_i)) - \mu(q_j))^2 + \\ (\mu(g(q_i), \mu(q_i), \sigma(q_i)) - \sigma(q_j))^2 \end{bmatrix}}{3}$$

- Here in the above equation, σ^2 represents the variance between $g(q_i)$, $\mu(q_i)$ and $\sigma(q_i)$

(v) Then the variance divides 1 that results the $ropt$ of a order of tasks q_i, which is as follows:

Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$ropt(q_i) = \frac{1}{\sigma^2(g(q_i), \mu(q_i), \sigma(q_i)) + 1}$$

- Here in the above equation, the variance is incremented by 1, which is to avoid the divide by zero error.

3.5 Ranking The Resultant Orders Of Tasks

Further the resultant orders of tasks will be ranked by their '$ropt$' from max to min and selects "max best order of tasks ($cbest$)" of cloud resource.
Then these $cbest$ orders of tasks will be ranked further, which is based on their coupling between tasks cbt from max to min, and selects "final best orders of tasks" ($fbest$) of ranked $cbest$.

4. EXPERIMENTS AND RESULTS EXPLORATION

The performance analysis of the proposed model was done by the dataset synthesized such that the tasks reflect the need of coupling and divergent prioritization of QoS factors. The dataset that used to assess the performance of the proposal is explored in table 1. The dependency scope of the tasks to be scheduled demands the assessment of the metric called coupling between tasks (cbt). The divergent prioritization of the QoS factors is significant in measuring optimal resource utilization scope ($ropt$). Further these two metrics are used to order the tasks towards resource scheduling. The experiments were done with divergent set of tasks in the range of 70 to 250. The devised explorative statistical analysis was done using expression language called R. The performance analysis of the devised model is assessed through the computational metrics called time complexity and time taken for task completion. The other metric optimal resource utilization, which is a performance assessment metric, is also used to estimate the scalability and robustness of the proposed model. In regard to this the proposed model is compared with two other models RRank [14] and Berger Model [13], which are of the statistical assessment strategies as like as our proposed model.

Table 1: The Data Used For Experiments

Number of tasks	450
Range of tasks to be scheduled	70-250
Range of dependency scope	5-75

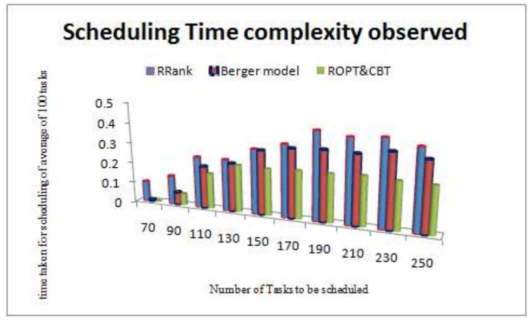
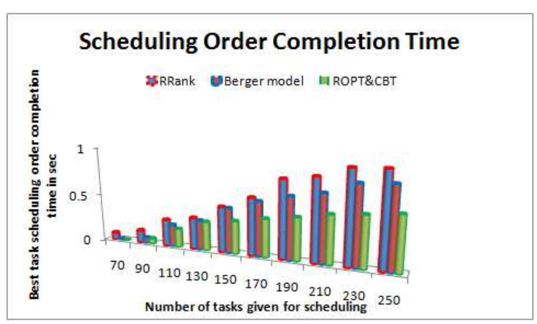


Fig 1: Time Complexity Observed To Schedule An Average Of 100 Tasks

The experiment results indicting the scalability and robustness of the proposed model, which is observed based on the performance metrics, time complexity (see fig 1), task scheduling completion time (see fig 2) and job completion time of the tasks scheduled (see fig 3). The observations from figure 1 indicating that the time taken for ratio of 100 tasks is low and stable in the case of proposed explorative statistical analysis model that compared to other two models RRank and Berger Model. The scheduling completion time for the range of tasks from 70 to 250 is also optimal for devised model over the other benchmarking models, which is visualized in fig 2. Optimal resource utilization is measured towards the job completion time for the ratio of 100 tasks scheduled. The figure 3 is exploring the scalability and robustness of the devised explorative statistical analysis model that compared to RRank and Berger model. The average scheduling time complexity of RRank and Berger model is growing as 1.1% and 0.65% respectively that compared to our proposed ROPT&CBT. Task Scheduling completion time observed with a growth of 2% and 1.2% at RRank and Berger Model respectively that compared to ROPT&CBT.



+
Fig 2: Tasks Scheduling Completion Time Observed

Journal of Theoretical and Applied Information Technology
10th October 2015. Vol.80. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

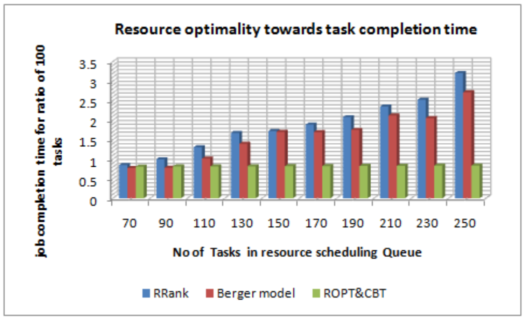


Fig 3: Optimal Resource Utilization Observed

5. CONCLUSION

The objective of the proposed model is multi objective QoS aware task scheduling towards optimal resource utilization. In regard to this an explorative statistical analysis model is used here in this paper that assesses two proposed metrics called resource optimal utilization (*ropt*) and coupling between tasks (*cbt*). The metrics *ropt* and *cbt* are used as key factors to order the tasks to be scheduled towards target resource utilization. The results explored are concluding that the devised metrics are capable to achieve scalability and robustness in ordering tasks to be scheduled. The said metrics are concluding the best fit order of the tasks towards optimal resource utilization. The considerable factors if any influencing the order of the tasks defined then this model is not confirming the alternative order of the tasks. Hence in this regard our future research would be initiated to devise a evolutionary computational strategy, which will use the proposed metrics *ropt* and *cbt* as fitness function parameters.

REFERENCES

[1]. R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, 1. Brandic. Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 2009, 25(6), p. 599-616.

[2]. Esha Bansal, Nisha Bansal, 2011. “An Analysis of Cloud Computing”. International Journal on Computing and Corporate Research (IJCCR), Vol.1, Issue 3, Manuscript 7, November 2011

[3]. An Oracle White Paper in Enterprise Architecture, 2009. ”Architectural Strategies for Cloud Computing”, August 2009

[4]. Syed Muhammad Ahsan. A framework for QoS computation in web service and technology selection. Computer Standards & Interfaces, 2006, 28(6), p. 714-720

[5]. KwangSik Shin, MyongJin Cha, MunSuck Jang, JinHa Jung, WanOh Yoon, SangBang Choi. Task scheduling algorithm using minimized duplications in homogeneous systems. Journal ofParallel and Distributed Computing, 2008, 68(8) , p. 1146-1156.

[6]. Braun TD, Siegel HJ, Beck N, etc. A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems. Joumal of Parallel and Distributed Computing, 2001, 61(6) , p. 810-837.

[7]. XiaoShan He, Xianhe Sun and Gregor von Laszewski. QoS guided Min-Min heuristic for grid task scheduling. Journal of Computer Science and Technology, 2003, 18(4) , p. 442-451.

[8]. Ching-Hsien Hsu, Zhan, J., Wai-Chi Fang, et al.Towards improving QoS-guided scheduling in grids. 2008 Third ChinaGrid Annual Conference (CHINAGRID). Dunhuang, Gansu, China, 2008, p. 89-95.

[9]. Y. C. Lee, A. Y. Zomaya, Rescheduling for reliable job completion with the support of clouds, Future Generation Computer Systems 26 (2010) 1192_1199

[10]. E. Vineka, P. P. Beranb, E. Schikutab, A dynamic multiobjective optimization framework for selecting distributed deployments in a heterogeneous environment , Procedia Computer Science 4 (2011) 166–175

[11]. V.M. Lo, "Task assignment in distributed systems", PhD dissertation, Dep. Comput. Sci., Univ. Illinois, Oct. 1983.

[12]. G. Gharooni-fard, F. Moein-darbari, H. Deldari and A. Morvaridi, Procedia Computer Science, Volume 1, Issue 1, May 2010,Pages1445-1454 ,ICCS 2010.

[13]. Baomin Xu, Chunyan Zhao, Enzhao Hua, Bin Hu. Job scheduling algorithm based on Berger model in cloud environment. Advances in Engineering Software, 2011, 42(7) , p. 419-425.

[14]. Xiaoyong Tang, Kenli Li, Renfa Li, Bharadwaj Veeravalli. Reliability-aware scheduling strategy for heterogeneous distributed computing systems. Joumal ofParallel and Distributed Computing, 2010, 70(9) , p. 941- 952.

[15]. http://www.tc3.edu/instruct/sbrown/stat/shape.htm

[16]. http://www.statsoft.com/textbook/anova-manova