Critical Path Based Ant Colony Optimization for Cloud Computing using meta heuristic approach

Rupinder Singh[1], Dr. Kanwalvir Singh Dhindsa[2]
[1]*Research Scholar,*
[2]*Professor (CSE),*
[1-2]*Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib, I.K.G. Punjab Technical University, Jalandhar, PUNJAB*

Abstract: Workflow tasks computing environment when we are dealing scientific computation using scientific workflow scheduling environment under deadline constraint, QoS is one of the most challenging task for any system used in scientific computing systems. In the local computing environment, while dealing with scientific computation using scientific workflow scheduling environment under deadline constraint, QoS is one of the most challenging task for any system used in scientific computing systems. Because while focusing on minimizing the workflow execution cost as well as time, the user-defined quality of service requirements should not forget to consider while minimizing the workflow execution cost and time. Therefore, to reduce the cost and time cloud environment is used. In cloud environment, resources will increase but its utilization is another challenge. Therefore, in this paper, Ant Colony Optimization (ACO) has been used, which is initialized by Pareto Distribution. ACO is used to converge the decision of Virtual Machine (VM) migration by its convergence to minima of cost and time. In this experiments, Total Execution Time (TET) and Total Execution Cost (TEC) is used in which ACO shows significant performance components when are compared to existing Genetic algorithm.

Keywords: Cloud Computing, Workflow scheduling, Load Balancing, Deadline Constraint, Total Execution Time, Total Execution Cost

I. INTRODUCTION

These days, Cloud computing is a developing zone in distributed computing that conveys progressively versatile administrations on demand over the web through the equipment and programming virtualization. The greatest preferred standpoint of the cloud is its adaptability to rent and discharge resources according to the client necessity. Besides, the cloud supplier offers two sorts of plans to be specific on demand short-term plan and long-term reservation plan. It has good framework i.e. Scalability, Transparency, Security and Monitoring [1].

Load Balancing :In computing, load balancing is the approach with which strings, methodology or straming information are offering access for framework assets (e.g. time processor takes for processing, correspondences data transmission) [5]. Normally, this is done for load adjusting and share framework assets adequately or accomplishes a target administration nature. The necessity for a planning calculation rises up out of the essential for most forefront frameworks for performing multitasking (executing more than one technique at once) and multiplexing (various information streams transmit all the while over a solitary physical channel).

Mainly the scheduler is concerned with:

- Throughput The processes number in total which completes their execution as per time unit.
- Latency
  a) Turnaround time - the total time among a process submission and its end.
  b) Response time - the amount of time it takes from the request submission time till the first response production.
- Fairness Each process is given equal CPU time (or generally more appropriate times in accordance to workload and every process priority).
- Waiting Time The time when the process remains in the ready queue [6].

A workflow management system (WFMS) gives an infrastructure to the set-up, performance, and checking of a characterized grouping of tasks, arranged as a work process. Work process can be automated with software devices that utilization business guidelines to choose when one stage has been finished effectively and the following stage can start. Some work process management software programs can also coordinate ward relationships between individual strides, an idea known as workflow orchestration.

II. LITERATURE REVIEW

Buyya et al. [1] characterized Cloud computing and gives the design for making Clouds with resource allocation which is market-oriented by using innovations, such as, Virtual Machines (VMs). Likewise, giving a knowledge bits on market-based resource administration techniques which includes both client driven service computational and administration risk administration for maintaining the Service Level Agreement (SLA)- oriented resource allocation. Moreover, highlighted the distinction among High-Performance Computing (HPC) workload and services workload based on internet and portraying a meta-negotiation framework for establishing Cloud trades and markets worldwide, and delineate a contextual tackling `Storage Clouds' investigation for high-performance content conveyance. At last, concluded with the convergence need of going after IT ideal models for conveying their vision of 21st-century.

Vockler et al. [2] depicted their encounters in the cloud, the running of a logical work process application. The applications were produced for processing astronomy information discharged with the Kepler extend, a NASA mission for looking to planet like Earth circling different stars. This work process was conveyed over different clouds utilizing the Workflow Pegasus Management System. The clouds utilized incorporate a few destinations inside the Future Grid, Amazon EC2, and NERSC's Magellan cloud. They portray how the application was sent, assess its performance executing in various clouds (based on Eucalyptus, EC2, and Nimbus), and discuss the difficulties of conveying and executing work processes in a cloud environment. They additionally demonstrate how Pegasus could bolster sky computing by executing a solitary work process over various cloud foundations at the same time.

Liu et al. [3] proposed an adaptive penalty function in this paper for the strict constraints compared with other genetic algorithms. In addition, the co-evolution approach is utilized to adjust the mutation and hybrid probability, which is able to accelerate the convergence and keep the pre-maturity. Also, compared our algorithm with baselines, for example, Random, HEFT (Heterogeneous Earliest Finish Time), genetic algorithm, and particle swarm optimization in a Workflow Sim simulator on four representative scientific workflows. The outcomes demonstrates the superior performance of it to the other state-of-the-art algorithms in the criterion of both the total cost of execution and probability meeting deadline constraint. The technique in this paper, a CGA with adaptive penalty function (CGA2) for scientific constrained workflow scheduling in clouds is proposed. The common drawback of existing evolutionary algorithms is the need of defining problem-specific parameters of penalty function for constrained optimization issue. And these algorithms are also static and lead to premature convergence. To address these issues, their proposed algorithm plans an adaptive penalty function with no parameter tuning and is easy to execute.

Future work will utilize a multi-objective evolutionary algorithm for taking care of the cloud resource scheduling issue and will take into account the load balance and task failures. Meanwhile, we will stretch out the resource model for considering the data transfer cost among data focuses.

Foster et al. [4] endeavored for comparing and contrasts Cloud Computing along with Grid Computing from different angles and give insights are given into both important characteristics. This paper demonstrates which Grids and Cloud shares a ton commonality in their innovation, vision, and architecture, however varies in different aspects, for example, programming model, security, compute display, business show, applications, data model, and abstractions. Also, recognize openings and challenges in the two fields. A comparison, for example, this can enable the two communities for sharing, understanding, and advancement in infrastructure and innovation across and within, and accelerates the Cloud Computing from early models for production frameworks.

Future scope hazards a couple of predictions which are based on our convictions that the computing economics will look increasingly like those of vitality. Neither the vitality nor the tomorrow computing grids will resemble yesterday's electric power grid. Both will move towards a blend of miniaturized scale production and large utilities, with small-scale makers number increase (the biomass, solar, wind, and so forth., for vitality; for computing, local groups, and installed processors—in walls and shoes) co-existing with regional large-scale makers, and the load being dynamically distributed among them.

Liu et al. [5] showed study of workflow cloud application and the cloud workflow based architecture is presented for Smart City. Then, a workflow scheduling algorithms diversity is assessed. The motivation behind this paper is for making taxonomy of workflow scheduling and management in a cloud environment, and also applying this workflow based cloud architecture for Smart City environments, further several research challenges is presented in this field.

The further research related work challenges, with the workflow complexity and scale being increased greatly, already an individual cloud can’t satisfy the prerequisite of it. The already existed algorithms greater part is suitable only for the individual cloud environment. Later on, it can be researched in workflow management framework running on Inter-Cloud. The challenges primarily should have been explained is finding an approach appropriately in Inter-Cloud environment for service orchestration that ought to consider regarding several criteria for meeting QoS prerequisites and orchestration service optimization under the deadline and cost constrained as well.

Pandey et al. [6] displayed a heuristic based particle swarm optimization (PSO) for planning the applications for the cloud resources taking into account both the computational cost and transmission data cost and explore different avenues regarding a workflow application with the variation of its cost of communication and costs of computation. The cost savings is compared while utilizing 'Best Resource Selection' (BRS) and PSO algorithm. The approach comes about demonstrate which PSO can be achieve: a) as much as three times cost savings in comparison to BRS, and b) greater workload distribution onto resources.

Zhu et al. [7] highlighted such troubles, and modelling the workflow scheduling issue that enhances both cost and make span for the Cloud environments as a Multi-target Optimization Problem (MOP). The procedure proposed an algorithm based on Evolutionary Multiobjective Optimization (EMO) for tackling this issue of workflow scheduling on an IaaS (Infrastructure as a Service) platform. The novel plans to issue particular initialization of population and encoding, wellness evaluation, and this algorithm propose genetic operators aswelll. Broad tests on randomly generated workflows and real-world workflows demonstrate that the timetables delivered by their evolutionary algorithm display greater stability on a large workflows portion with IaaS computing based on instances and pricing models. Furthermore, the overheads time and monetary costs of both storage and communication is included in the considerations.

Ostermann et al. [8] showed an assessment of the handiness of the present services of cloud computing to scientific computing. Analyzing the performance of the Amazon EC2 platform by utilizing miniaturized scale benchmarks and pieces. While clouds yet changing are the cloud, their outcomes indicate that the present cloud services require a magnitude request in performance change to be valuable for the scientific community. The procedure work having additional other services analysis offered by Amazon: database (SimpleDB), Storage (S3), Private Cloud, line service (SQS), and their inter-connection and also expand the performance evaluation comes about with the running of similar trials on other IaaS suppliers and also clouds on other large-scale real platforms, for example, grids and commodity and grids bunches.

Alkhanak et al. [9]: depicted that workflow scheduling (WFS) predominantly focus is on task allocation for accomplishing the pined for workload adjusting by seeking after ideal use of accessible resources. In the meantime, pertinent execution criteria and appropriation system structure which must be considered for handling specific WFS issues in cloud computing by giving particular services on pay-as-you-go and on-request premise to cloud customers. In the writing, different challenges of WFS influencing execution cost of WFS have been talked about. This paper principle objective is to encourage researchers in choosing suitable cost-aware WFS come closer from the accessible choices pool. For accomplishing these objectives, they have directed a wide study to explore and break down the fundamental ideas of the significant methodologies. The cost-mindful pertinent difficulties of WFS in cloud computing are arranged in light of Quality of Service (QoS) execution, structure usefulness, and system design, that at last outcome in a scientific categorization set. Some research open entryways have additionally talked about that help with distinguishing future research headings in the cloud computing zone. This review discoveries give a roadmap to creating cost-mindful models, that persuade researchers for proposing a better cost-aware methodologies for service buyers as well as utility providers in cloud computing.

Shukla et al. [10] Clarified all difficulties, the services quality is the most detectable test and influences the services of cloud computing. The service quality can be upgraded with consideration of the few elements, workload scheduling for appropriate resources of cloud computing one of them. If the cloud computing resources are reserved precisely, it influences the services reaction time, add up to cloud resources cost, the imperativeness utilization decreased, diminish the CO2 discharge and upgrade the execution of entire cloud system.

On the base of past considerations; they have given that algorithms built up as indicated by the customer require parameters. Parameters that upgrades security issues, the aggregate cost issues, imperativeness utilization issues, execution issues, QoS issues in the Multi-target workflow territory.

III. PROPOSED WORK

Proposed Workflow Scheduling ACO Algorithm:

With the above given ant algorithm characteristic utilization, the task can be scheduled. Similarly, new task can be carried out with the utilization of previous task scheduling result. The basic ideas of ACO algorithm is inherited in Workflow Scheduling-ACO algorithm for the reduction of execution time and cost.

Firstly, input the workflow to the workflow simulator and parse the task from this workflow. Pareto distribution is followed by the task. On VMs, there is a pareto distribution of ants at the beginning and then, VM_i pheromone values are initialized:

$$\tau_i(0) = p\_NUM_i \times p\_MIPS_i + VM\_b_i$$

Where

p_NUM_i ← number of VM_i processor

p_MIPS_i ← million instructions per second of each VM_i processor

VM_b_i ← VM_i communication bandwidth ability

Choosing VMs rule for next task: For next task, VM_i choose by k-ant with probability defined as:

$$P_i^K(T) = \begin{cases} \frac{[\tau_i(T)]^\alpha [c_i]^\beta [lb]^\gamma}{\sum[\tau_K(T)]^\alpha [c_K]^\beta [lb]^\gamma} & if\ i \epsilon 1, 2 \ldots\ldots\ldots\ldots n \\ 0 & otherwise \end{cases}$$

Where

$\tau_i(T)$ ← pheromone value of VM_i at time T

c_i ← VM_i computing capacity

C_i can be defined as:

$$c_i = p\_NUM_i \times p\_MIPS_i + VM\_b_i$$

lb_i ← VM_i load balancing factor for minimizing the degree of imbalance defined as:

$$lb_i = 1 - \frac{et_i - Avg\_et}{et_i + Avg\_et}$$

Where

Avg_et ← virtual machine average execution time in the optimal path last iteration

et_i ← expected execution time of VM_i task

et_i is defined as:

$$et_i = \frac{total\_TL}{c_i} + \frac{Input\_FS}{VM\_b_i}$$

Where

$total\_TL$ ← total length of task submitted to VM_i

$Input\_FS$ ← task length before execution

$\alpha, \beta\ and\ \gamma$ ← parameters controlling the relative weight of pheromone trail along with VMs computing capacity and load balancing.

Once heavily loaded are some VMs becoming bottleneck in cloud influences the given task set makespan. The load balancing factor lb_i is defined in the ant algorithm for improving the capacity of lead balancing. Bigger the lb_i, higher will be the probability of choosing means VM_i comprehensive ability is greater now.

Updating Pheromone: Ant Let $\tau_i(T)$ at any time T be the VM_i pheromone intensity. The update of the pheromone is given by:

$\tau_i(T+1) = (1-\rho) \times \tau_i(T) + \Delta\tau_i$

Where

$\rho \epsilon [0,1] \leftarrow$ decay coefficient of pheromone trail

The past solution impact will be less if value of ρ is greater. The $\Delta\tau_i$ value is defined as:

After the completion of ant tour, updating the local pheromone on VM visited and $\Delta\tau_i$ value is given as:

$\Delta\tau_i = 1/t_{iK}$

Where

$t_{iK} \leftarrow$ K-ant searched shortest path length at i^{th} iteration

In case, the current optimal solution is found by the ant while completing its tour, larger intensity pheromone is laid on its tour and updating the global pheromone on VM visited and $\Delta\tau_i$ value is given as:

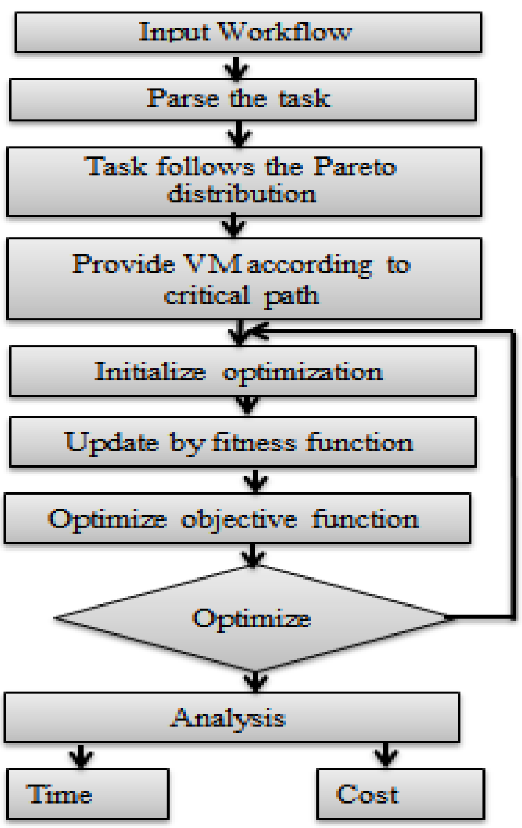
$\Delta\tau_i = d/t_{op}$

Where

$t_{op} \leftarrow$ current optimal solution

$d \leftarrow$ encouragement coefficient

If the function is optimized then we analysis the cost and time of that function.



IV. RESULTS

In below given Table4.1 ,4.2and 4.3 show the behavior of GENOME workflows in different number of workflows and Virtual machines which represent by ensemble size. In results, show the Ant colony optimization and genetic algorithm optimization on total execution time, total execution cost and response time. If analysis these parameter ACO perform well in cost and time parameter because of ant colony searching time decide by adaptive pheromones and VM task migration depend on Transient problem but in genetic algorithm both is depend on candidate solution and which is static but ACO initialization is depend on pare to distribution which is depend on normal distribution. However, response time of Genetic algorithm is better than ACO because of pare to distribution take more time for mapping of VM by task. It will effect on Total cost execution because pare to VM mapping but TET always have significance improve.

Table 1: Comparison table of GA and ACO using GENOME

RESULTS OF GA AND ACO USING - GENOMEGENOME			
Ensemble	ACO		
Size	TET	TET	TET
2	23.2	23.2	23.2
4	73.24	73.24	73.24
6	133.83	133.83	133.83
8	315.65	315.65	315.65
10	745.24	745.24	745.24
12	546.78	546.78	546.78
14	317.76	317.76	317.76
16	401.95	401.95	401.95
18	791.04	791.04	791.04
20	746.6	138725.4	0.00113138

RESULTS OF ACO USING -GENOME			
Ensemble	GA		
Size	TET	TEC	Response Time
2	137.78	7549.552	0.00008362
4	201.57	42511.93	0.00063674
6	370.15	28155.45	0.00013048
8	284.46	58872.21	0.00132634
10	365.5	77034.56	0.00151176
12	423.76	66004.98	0.00141466
14	486.7	82720.56	0.00147464
16	530.13	113448.6	0.00167419
18	419.1	119935.5	0.00157754
20	746.6	138725.4	0.00113138

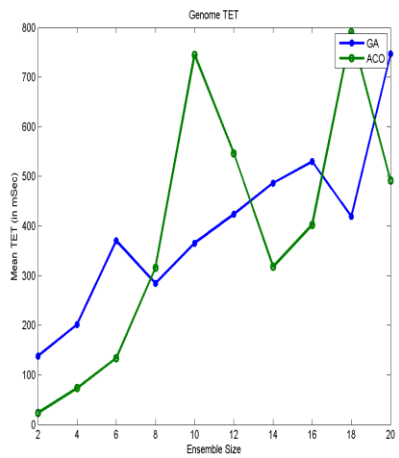


Fig. 1: Comparison graph of TET of GA and ACO using GENOME

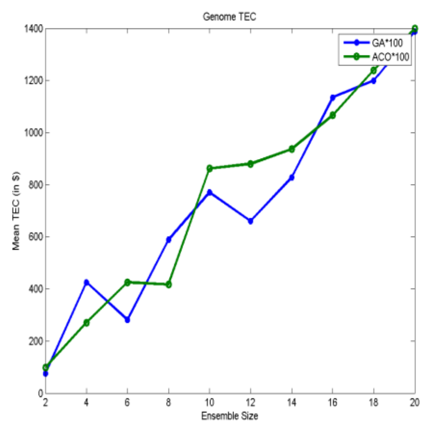


Fig. 2: Comparison graph of TEC of GA and ACO using GENOME

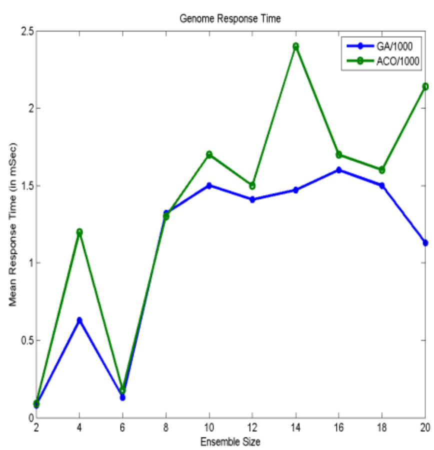


Fig. 3 Comparison of Response time GA and ACO

In results, show the Ant colony optimization and genetic algorithm optimization on total execution time, total execution cost and response time. If analysis these parameter ACO perform well in cost and time parameter because of ant colony searching time decide by adaptive pheromones and VM task migration depend on Transient problem but in genetic algorithm both is depend on candidate solution and which is static but ACO initialization is depend on pare to distribution which is depend on normal distribution. But response time of Genetic algorithm is better than ACO because of pare to distribution take more time for mapping of VM by task. It will effect on Total cost execution because pare to VM mapping but TET always significance improve.

V. CONCLUSION AND FUTURE SCOPE

In this, work is done on different workflows genome, cyber shake etc on TET and TEC parameter in different virtual machine or ensemble size. In this thesis, two to twenty ensembles size and optimize are used by genetic algorithm and ant colony optimization. In experiment results, ACO reduces the average TET and TEC in different workflow. So, we concluded that ACO optimize and converge workflow scheduling in cloud scenario. work on different workflows Genome, Cyber shake, LIGO, Montage and SIPHT on TET and TEC parameter in different virtual machine or ensemble size. In this paper, we use two to twenty ensembles size and optimize by genetic algorithm and ant colony optimization. In experiment results, ACO reduces the average TET and TEC in different workflow. Therefore, we concluded that ACO optimize and converge workflow scheduling in cloud scenario. As per the result analysis, we find that response time of ACO is more as compare to GA in local simulation. Hence, in-order to reduce response time of ACO, it can be executed in real-time cloud environment using SLA. In addition, this work can be extended for multi-objective algorithm to get solution for load balancing and task failures. The proposed model performance can be enhanced by increasing the VM numbers for better response time. In future this work can be enhance on hybrid optimization and multi objective optimization because both are reduce the time of computation because of single objective optimization, which use in research work in several ways and increase the computation time.

VI. REFERENCES

[1] R. Buyya , C. S. Yeo, S. Venugopal S, J. Broberg J, and I. Brandic "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility." Future Generation computer systems 25.6, 599-616, (2009)

[2] J. S. Vöckler, G. Juve, E. Deelman and M. Rynge "Experiences using cloud computing for a scientific workflow application." ACM Proceedings of the 2nd international workshop on Scientific cloud computing., 2011.

[3] L. Liu, M. Zhang, R. Buyya. "Deadline-constrained co evolutionary genetic algorithm for scientific workflow scheduling in

cloud computing." Concurrency and Computation: Practice and Experience (2016).

[4] I. Foster, Y. Zhao, I. Raicu, S. Lu, “Cloud computing and grid computing 360-degree compared", Grid Computing Environments Workshop, 2008. GCE'08. IEEE, 2008.

[5] L. Liu, M. Zhang, Y. Lin, L. Qin, "A survey on workflow management and scheduling in cloud computing." Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on. IEEE, 2014.

[6] S. Pandey, L. Wu, S. M. Guru, R. Buyya, "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments." Advanced information networking and applications (AINA), 2010 24th IEEE international conference on. IEEE, 2010.

[7] Z. Zhu, G. Zhang, M. Li and X. Liu, "Evolutionary Multi-Objective Workflow Scheduling in Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 5, pp. 1344-1357, May 1 2016.

[8] S. Ostermann, A. Iosup, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, "A performance analysis of EC2 cloud computing services for scientific computing." International Conference on Cloud Computing. Springer Berlin Heidelberg, 2009.

[9] E. N. Alkhanak, E. Nabiel, S. P. Lee, and S. U. R. Khan. "Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities." Future Generation Computer Systems 50, 3-21 (2015).

[10] S. Shukla, A. K. Gupta, S. Saxena and S. Kumar "An Evolutionary Study of Multi-Objective Workflow Scheduling in Cloud Computing." International Journal of Computer Applications (0975–8887) Volume (2016).

[11] D. S. Chawla, K. S. Dhindsa, "Comparison of job scheduling algorithms in cloud computing." Asian Journal of Multidimensional Research (AJMR) 6, (2017)

[12] D. S. Chawla, K. S. Dhindsa, " A Load Balancing Based Improved Task Scheduling Algorithm in Cloud Computing." International Journal for Research in Applied Science & Engineering Technology (IJRASET) 45.98, (2017)