

Workflow scheduling and reliability improvement by hybrid intelligence optimization approach with task ranking

S. Khurana^{1,*}, R.K. Singh²

¹I. K. Gujral Punjab Technical University, Jalandhar, Punjab, India

²SUS Institute of Computer, Tangori, Mohali, Punjab, India

Abstract

Workflow scheduling is one of the most challenging tasks in cloud computing. It uses different workflows and quality of service requirements based on the deadline and cost of the tasks. The main goal of workflow scheduling algorithm is to optimize the time and cost by using virtual machine migration. This algorithm computes the subset problem and decision problem in NP time. It works on the decision-making process to reduce the time and cost of computation on the server side. This paper proposes hybrid optimization to optimize the virtual machine locally and globally. The PEFT algorithm is used for initialization and worked as a heuristic algorithm. This algorithm reduces the error of random initialization of optimization. The proposed algorithm based upon Flower pollination with Grey Wolf Optimization using hybrid approach shows significant end effective results than flower pollination with genetic algorithm. The proposed approach also considered the reliability parameter on different workflows.

Keywords: Workflow Scheduling, Reliability, Cloud, Virtualization, Hybrid Optimization.

Received on 01 September 2019, accepted on 01 November 2019, published on 06 November 2019

Copyright © 2019 S. Khurana *et al.*, licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eai.13-7-2018.161408

*Corresponding author. Email: savu.khurana30@gmail.com

1. Introduction

Cloud computing is a platform which provides the virtualized access to the services and application to the clients. The physical location of the client it doesn't matter in the cloud computing services; the user can access the services and resources from anywhere and anytime. Sometimes our system is hanged up due to a large number of processes in the queue in which resources are acquired by some process. This process is also similar to the cloud and enhances the load on the cloud [1].

The load balancing on the cloud is important issues in cloud computing which affects the performance, reliability and scalability of the cloud. Load balancing is basically a process in which load is distributed among all nodes so that each node works efficiently and the performance of the cloud does not degrade. Load Balancing can be done by scheduling task, propose resource allocation and task

migration. Load balancing is done by using the device called Load Balancer[26]. It increases the capacity and reliability of the applications. It also improves the performance of the system by dividing the burden equally. Load Balancer works on the two layers that are network layer and transport layer. It assigns the data to the servers according to the data found in the application layer [2]. Load balancing in the cloud is mainly used in the cloud to the proper allocation of resources, reduce the resources consumption and maximize the throughput with minimum response time. On the basis of the systemstate, load balancing is divided into two types that are Static load balancing and Dynamic load balancing [3]. Static load balancing is needed when there is no variation in load. Task allocation or load distribution depends on the load at the time of selection of a node or based on the average load of the server. Performance of the server is taken into consideration for the selection of the server. This work is assigned on the basis of the incoming time of the job, execution time and inter-processes communication.

Dynamic load balancing is performed at the runtime. Distribution decisions are based on the current load information. In this balancing scheme, any advance information is not available. It works on the current situation and load balancing. It also works on the heterogeneous system's communication plays an important part to improve the performance of the system. It is also divided into two parts distributed and Non-distributed [4].

1.1 Motivation

The concept workflow scheduling is considered as the projecting issues in the mechanism of workflow scheduling which attempts to outline work process undertakings to the VMs dependent on various useful/functional and non-practical/non-functional necessities. Despite the fact that few planning concerns have been broadly examined, for example, the vulnerabilities delivered by framework failure, PC heterogeneity, asset versatility, cutoff time requirements, and budget restrictions among others, still there are some different uncertainties that have not pulled in the consideration they justify. These workflow or jobs will register with less time and less assets. Separately, it computes the process in a right manner and do not overlap with one another. This paper addressed a new meta-heuristic hybrid algorithm namely Flower Pollination with Grey Wolf Optimization (FPA_GWO). This algorithm has been implemented using different Scientific Workflows datasets like Sight, Ligo, and Genome and in most of the cases it gives better results.

2. Literature Survey

For solving the issue of load balancing we need some effective algorithm which effectively balances the load. The ant colony optimization algorithm is used for the load balancing of the cloud. This is a Metaheuristic algorithm and provides the optimal solution to the problem [26]. The response time of the resources is optimized by the ACO by distributing the load dynamically [5]. To distribute the load efficiently on the cloud fuzzy row penalty method is presented in [6] for cloud computing. The fuzzy approach is used to solve the problem of uncertainty in the response time in cloud resources. The fuzzy row penalty approach is used for both balanced and unbalanced fuzzy load on the cloud. The response time and space complexity are two parameters on which performance is measured.

The main goal of load balancing is to distribute the equal amount of load to each processor on the cloud so that the performance and scalability of the cloud don't affect by more load. The hybrid optimization algorithm is used for the optimization of a complex network of the cloud. The hybrid Ant Colony Optimization (ACO) algorithm is used for an optimal solution [7]. The load balancing is also based on the dynamic threshold in cloud computing. This is done by organized the virtual resources of data centers efficiently.

The proposed approach reduces the makespan and enhances the resource utilization with minimum usage of energy [8].

To find the best virtual machine on the cloud by considering the minimum makespan and power resource utilization [33]. The effective virtual machine is selected by using a chaotic Spider algorithm and it tackles the problem of task scheduling. The overall makespan during the scheduling is minimized by using a weight-based random selection method. The best virtual machine is finding out by using global intelligent searching [9].

S. Chenthara et al., proposed an ensure privacy and security of electronic health records (EHRs) in the cloud. The main storage for the data whether it is related to healthcare, computational based data, scientific calculation based data etc is cloud servers [27][30]. In this paper author highlights privacy aspects of data and address the cryptography and non-cryptography security techniques. Hua Wang et al., proposed Role-base access control(RBAC) algorithm which implement access control and its policies, authorization, grant and revoke permission to access the outsource data with security in cloud[29].

Md Enamul Kabir et al., proposed micro aggregation method to secure the microdata of the cloud computing [28]. Kang, Seungmin et al. addressed the issue of scheduling in multi-cloud systems. Prediction technique is used to deal with the issue of uncertainty of node. Dynamic scheduling strategy is proposed for multi-cloud systems. The total processing time of loads is minimized by using scheduling techniques. It considers the availability and heterogeneity of computing nodes. By using the proposed DSS method, divisible load theory and node availability give high performance [10].

Ajay et al., presented five different met heuristic algorithms and compare their performance and express the performance using different benchmark functions [32].

The author also proposed meta-heuristic optimized GACO algorithm for load balancing in cloud environment. In this paper author compare the performance using 17 different benchmarks functions. The results of proposed algorithm show better result than existing GA, PSO, ABC algorithms [34].

3. Research Problem

The logical work process scheduler details the planning issues as a weighted optimized issue of two targets (runtime and monetary cost). It allocates tasks to VMs to limit execution time and money related cost dependent on client necessities. To figure this issue, the considered undertakings or tasks are disseminated among the queues of VM, $[vm_1^{queue}, \dots, vm_p^{queue}]$. A queue (line) is characterized as disintegration of a set into incoherent (disjointed) subsets whose association is the primary set [31]. In light of this model, the problem of scheduling is characterized as finding the relating components of each VM queue to expand the accompanying augmented (amplified) objective function represented as follows:

$$F_a = y_1 \frac{(max^{time} - Runtime)}{(max^{time} - min^{time})} + y_2 \frac{(max^{cost} - Runtime)}{(max^{cost} - min^{cost})}$$

The variables min^{time} , max^{time} , min^{cost} , and max^{cost} are updated continuously at the time of workflow scheduling procedure mirroring the worst and the best configurations of VM's. These values generally represent the minimum and maximum values of monetary cost and runtime.

Runtime is characterized as the most extreme time taken by the least or slowest incredible VM to implement the present queues of occupations, communicated as:

$$Runtime = Max_{j=1}^{|VM|} [vm_j^{time}]$$

Here, vm_j^{time} represents the execution time to execute vm_j^{queue} on vm_j

$$vm_j^{time} = \sum_{j=1}^{|vm_j^{queue}|} t_1$$

Monetary or cost is generally defined as the summation of runtimes of each and every VM multiplied with its corresponding cost, represented as:

$$Cost = \left[\sum_{j=1}^{|VM|} [Runtime] vm_j^{cost} \right]$$

Since the owners of the application do not have devices to precisely measure complete execution time or on the other monetary cost related expense for their work processes, our methodology offers a unique and adaptable approach to pick a specific configuration of scheduling driven by y_1 and y_2 presenting the time of execution and the monetary-based cost optimization, where the weights reply over the following condition:

$$y_1 + y_2 = 1$$

Thusly, the owner of the work process provides a weight-based percentage to constraints dependent on the need.

4. Proposed Approach

Hybrid Intelligence Optimization Approach is used for scheduling of the tasks and reliability. In this paper Flower pollination algorithm (FPA) and Grey wolf optimization (GWO) algorithms are used as hybrid using PEFT algorithm for global and local optimization. The details of algorithm are following:-

FPA: Flower Pollination Algorithm is based on the concept of the transferring of pollen from one plant to other. The pollens transfer agents are mainly insects, bats and birds. Pollination process is divided into two parts that are Biotic Pollination and abiotic pollination. The most common pollination is biotic pollination and it occurred 90% in the

flowers [11]. The abiotic pollination occurs 10%. The biotic pollination needs a pollinator agent to complete the pollination process but abiotic pollination does not need an agent to complete the pollination process. In pollination, process insects go to different flowers and insects bypass some species of flower and this process is called as Flower consistency [12].

Flower pollination process is classified into two parts that are cross-pollination and Self-pollination. In cross-pollination, process pollens are transferred to different plants by the pollen agents. In the self-pollination process pollens of the same flower is responsible for fertilization.

The cross and biotic pollination are called as global pollination and follow the Levy Distribution.

The self and abiotic pollination are called local pollination [14, 15].

The reproduction ration can be computed by using the consistency of flower and it is proportional to the degree of similarity between two flowers.

Due to physical conditions like wind local pollination has an advantage over global pollination

The General steps followed by the Flower Pollination Algorithm are the following:

- I. Minimize or maximize the objective function and then create the random population with a specific size.
- II. Identify the best solution and then select the pollination process among cross and self-pollination.
- III. After this apply Levy Distribution and global pollination approach and calculate the fitness function.
- IV. Replace old solution if the new solution is better and finds the current best position for the next generation.

PEFT: Predict Earliest Finish Time (PEFT) is a scheduling algorithm which is based on the list and worked on the bounded on the number of heterogeneous processors. This algorithm works in two phases in which the first related to Task Prioritization (for computing task prioritization) and the second phase is Processor selection in which best processor is selected for executing the current task[16].

GWO: Grey Wolf optimization algorithm is a bio-inspired algorithm which is based on the leadership and hunting behavior of the wolves in the pack. The grey wolves prefer to live in the pack which is a group of approximate 5-12 wolves. In the pack, each member has social dominant and consisting according to four different levels. The below-given figure shows the social hierarchy of the wolves which plays an important role in hunting [17].

1. The wolves on the first level are called alpha wolves (α) and they are leaders in the hierarchy. Wolves at this level are the guides to the hunting process in which wolves seek

Existing approach flowchart

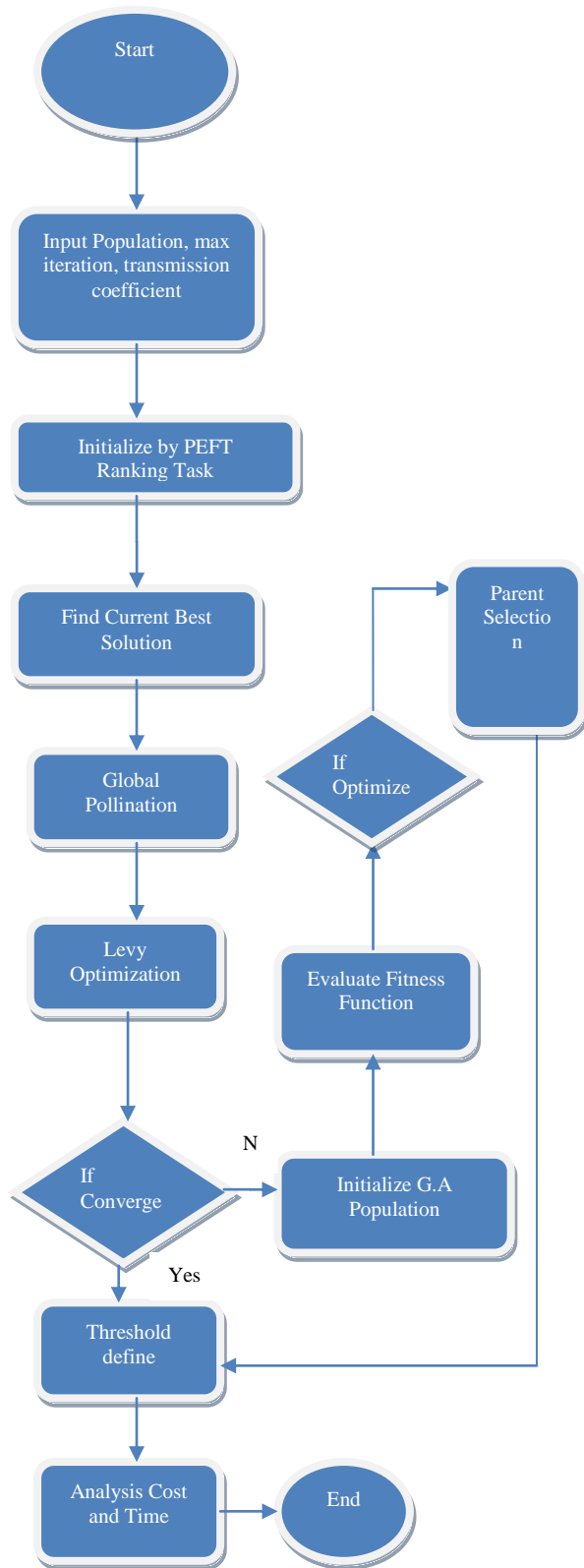


Figure 1. Existing FPA_GA optimization algorithm

Proposed Hybrid FPA_GWO Algorithm

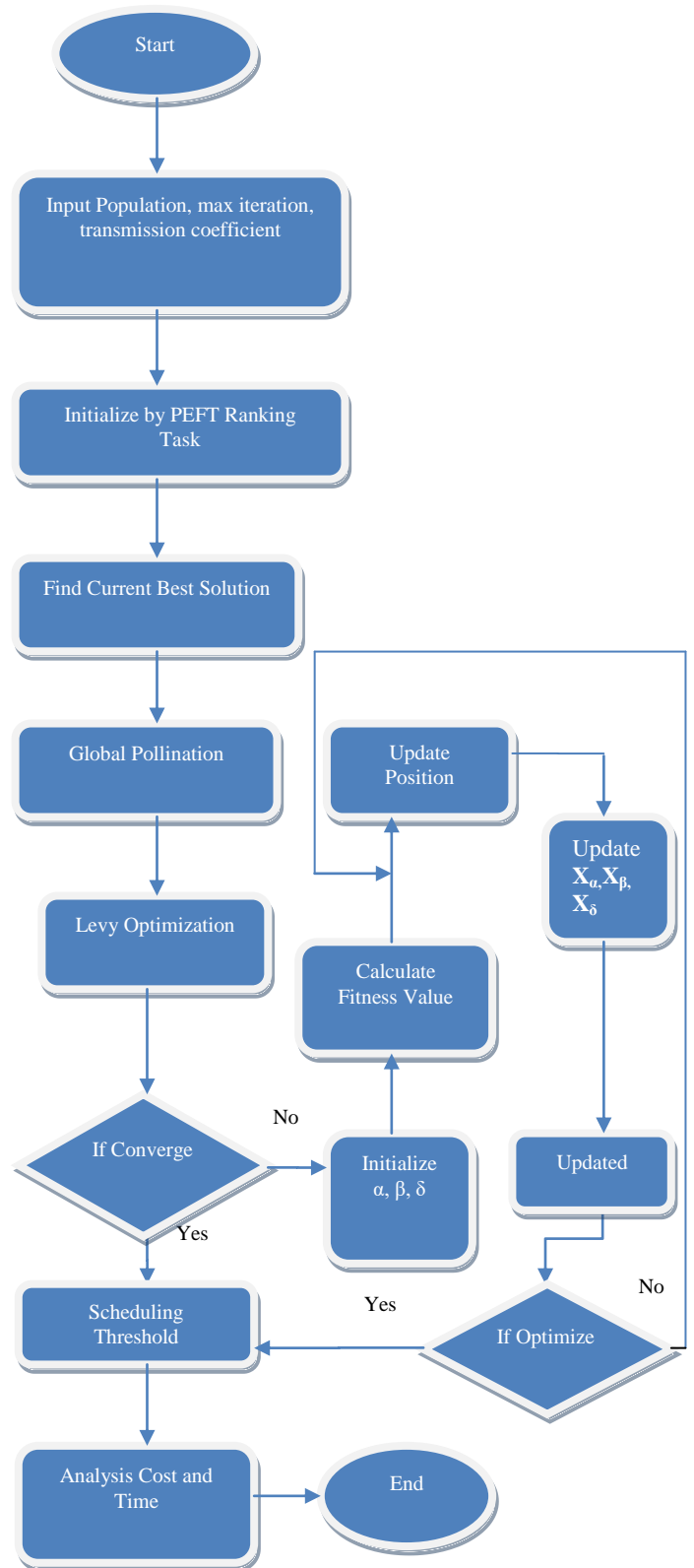


Figure 2. FPA_GWO hybrid optimization proposed algorithm

follow and hunt and work as a team. Decision making is the main task that is performed by the alpha wolves and order by the alpha wolves is followed by all members of the pack [18].

2. Second level wolves are called beta (β). These wolves are called subordinates and advisors of alpha nodes. The beta wolf council helps in decision making. Beta wolves transmit alpha control to the entire packet and transmit the return to alpha.

3. The wolves of the third level are called Delta wolves (δ) and called scouts. Scout wolves at this level are responsible for monitoring boundaries and territory. The sentinel wolves are responsible for protecting the pack and the guards are responsible for the care of the wounded and injured [19].

4. The last and fourth level of the hierarchy is called Omega (ω). They are also called scapegoats and they must submit to all the other dominant wolves. These wolves follow the other three wolves.

4.1 Proposed Hybrid algorithm FPA_GWO

Algorithm FPA_GWO
Step 1: Input Population, maxiteration, the transmission coefficient
Step 2: Initialize by PEFT Ranking.
Step 3: Initialize the Flower Pollination algorithm for the best solution and global Pollination
I. Initialization
II. Exploration process for global pollination
III. Exploitation process for local Pollination
IV. Update solution
Step 4: Check the convergence. If converged the scheduling threshold otherwise initialize the Grey Wolf Optimization Algorithm.
Step 5: Calculate fitness function for every search agent
$A_\alpha \leftarrow$ best search agent
$A_\beta \leftarrow$ second best search agent
$A_\delta \leftarrow$ Third best search agent
While ($T < \text{Max iterations}$)
For (X_i in every pack)
Update current position of wolf by Eq. (1)
Update x, X and Y
Calculate the fitness function for all search agents

Update A_α , A_β , and A_ω
End for
For best pack insert migration (m_i)
Evaluate fitness function for new individuals
selection of best pack
New random individuals for migration
End if
End while
Step 6: Analyze the cost and time

4.2 Simulation Parameters

Table I. Simulation Parameters

Parameters	Value
Number of tasks	25-1600
Number of workflows	2-20
Number of VM	2-20
MIPS	500
RAM	1024 (MB)
BW	500 (Mbps)
Number of Processors	1
VM Policy	Time Shared

4.3 Calculate Fitness Function

Time

$$T_t = \sum T_R + \sum T_P + \sum T_W \quad (1)$$

$$T_t \leftarrow \text{Total Time}$$

$$T_R \leftarrow \text{Receiving or passing time of task}$$

$$T_P \leftarrow \text{Processing Time of task}$$

$$T_W \leftarrow \text{Waiting Time of task}$$

$$\text{Total Cost: } \frac{MF+CF}{2} \quad (2)$$

MF: Movement Factor

CF: Cost Factor

MF =

$$\frac{1}{\text{Number of Host / Datacenter}} \left[\sum_{X=1}^{VM_X} \left(\frac{\text{Number of Migration}}{\text{Used VM}} \right) \right] \quad (3)$$

$$CF = \sum_{X=1}^{VM_X} \left(\frac{\text{Cost to Process} + \text{MEMORY OF TASK}}{VM \times \text{Datacenter}} \right) \quad (4)$$

Actual Cost

$$\text{Actual Cost} = \text{Under Deadlined Total Cost} + \text{Deadline cost} \quad (5)$$

Fitness Function

$$F_x = \frac{1}{\text{Datacenter} \times VM_i} \left[\sum_{i=1}^{DC_i} \sum_{i=1}^{VM_i} \frac{1}{VM} \frac{\text{CPU Utilization}}{CPU_{ij}} + \frac{\text{Memory Utilized}}{\text{Memory } i_j} + \frac{\text{Time Utilized}}{\text{Time } i_j} \right] \quad (6)$$

5. Results

This section presents the results on the different workflows of the cloud that are LIGO, Genome, and Sight. The results analysis is done on the two parameters that are Cost and Time.

Results with LIGO

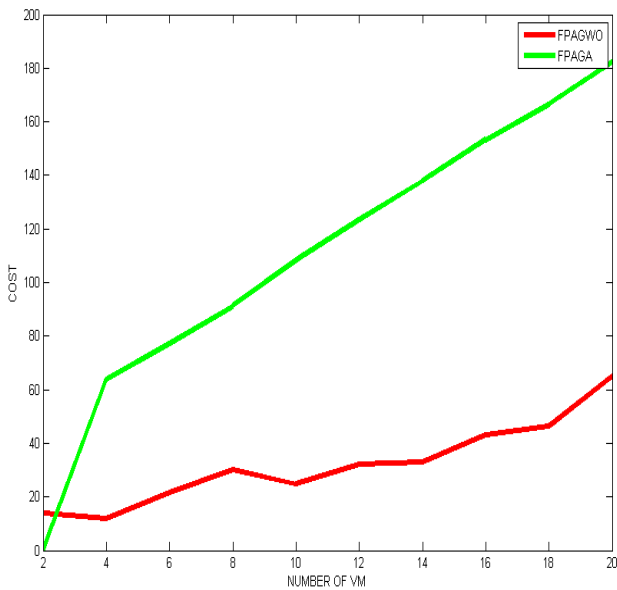


Figure 3. Cost on LIGO workflow for FPA_GWO and FPA_GA

Figure 3 depicts the cost of LIGO workflow for FPA_GWO and FPA_GA. The green curve shows the cost of FPA_GA and the red curve shows the cost of FPA_GWO. The cost of proposed FPA_GWO is less than FPA_GA. In Figure 3 comparison of flower pollination algorithm with genetic algorithm (FPGA) and flower pollination algorithm with grey wolf optimization (FPAGWO) on Cost parameter. The cost analysis is done

by using equation 3 and 4 on the basis of VM migrations. VM migration is basically a process in which task is transferred to one VM to another VM and this decision is taken by the optimization algorithm. This algorithm works on the reduction of cost and time. In this research experiment, hybrid optimization is used and initialized by the PEFT algorithm [21]. This algorithm is a heuristic algorithm which initializes and start mapping of VM. The migration decision in this work is taken by FPA_GWO. The above-given Fig 3 depicts the cost analysis of FPA_GWO and FPA_GA. The analysis is done in the following way.

- The decision of a number of migrations

In this analysis, task migration depends on the number of workflows because it increases the preparation tasks and respectively VM will also increase. So, the analysis point considered when the number of workflow and types of workflow increase and what will the effect on the decision of optimization. In Figure 3 FPA_GA increase the cost which clearly depicts that complexity enhances the time shown in Figure 4. The decision time is also increased when the time is increases and VM task are not migrated. The VM host continues as given by the server and it reduces the utilization. If this analysis is performed in the different workflow like genome in Fig 5 and Fig 6 for cost and time respectively. In GENOME workflow zigzag in the cost curve shows the complexity which varies when the workflows are increased and time is also increased as compared to LIGO workflow with FPA_GWO optimization.

- The decision of Balancing Time of computation and Cost

This point is very challenging in this research because when the cost is increased the number of VM is also increased for few numbers of tasks and workflow at the time of computation. The waiting time is reduced but the cost is enhanced. So, the decision of optimization plays a vital role to balance the VM and time of computation and needs the global and local optimization. For both local and global optimization algorithms like particle swarm optimization (PSO) [22], Ant colony optimization (ACO) [24] is used but they increase the time because of two optimizations. In this work, hybrid optimization is used in which one is used for local and meantime other for global according to local optimization output.

Figure 4 depicts the time of LIGO workflow for FPA_GWO and FPA_GA. The green curve shows the cost of FPA_GA and the red curve shows the cost of FPA_GWO. The time of proposed FPA_GWO is less than FPA_GA.

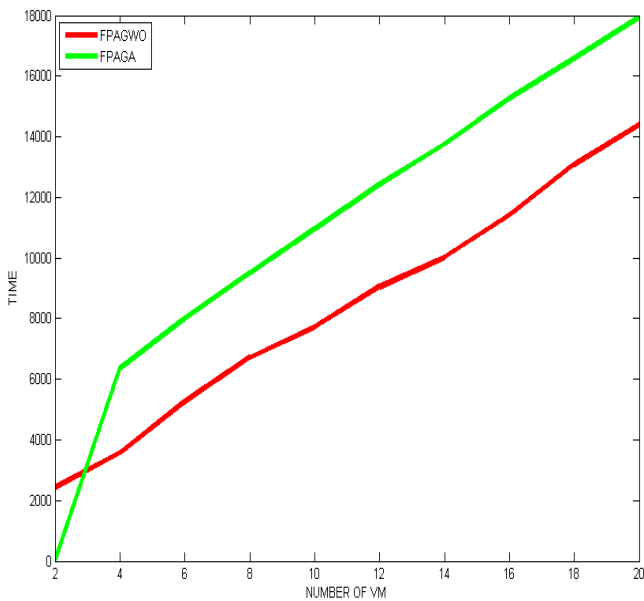


Figure 4. Time on LIGO workflow for FPA_GWO and FPA_GA

• **Selecting Proposed algorithm**

In this paper, the hybrid optimization is used because of the reasons mentioned in the above-given section but to decide which algorithm will hybrids discussed in this section under this point. First of all, two optimization algorithms that are based on bio-inspired and swarm intelligence are considered in which FPA is hybrid with GWO. In FPA_GWO algorithm, FPA is used because it is a Local optimizer and locally optimizes the task in VM and GWO global optimizer which take the decision of VM migration. In other hand use, FPA_GA is also swarm based and bio-inspired method.

Results with Genome

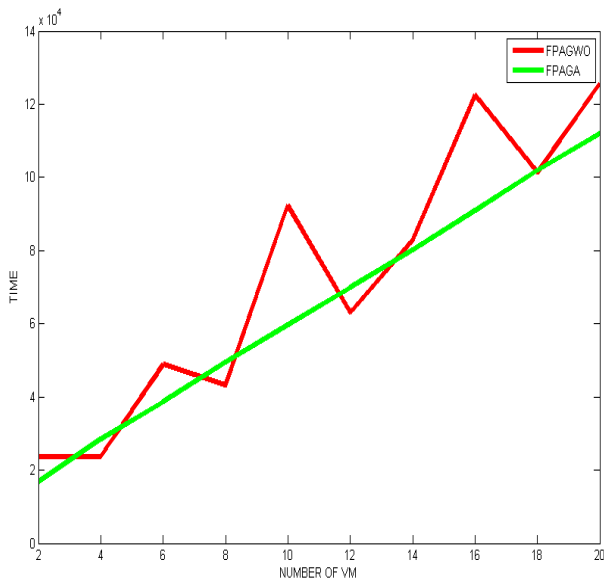


Figure 5. Time of Genome workflow for FPA_GWO and FPA_GA

Figure 5 depicts the time of Genome workflow for FPA_GWO and FPA_GA. The green curve shows the time of FPA_GA and the red curve shows the time of FPA_GWO. The time of proposed FPA_GWO is less than FPA_GA.

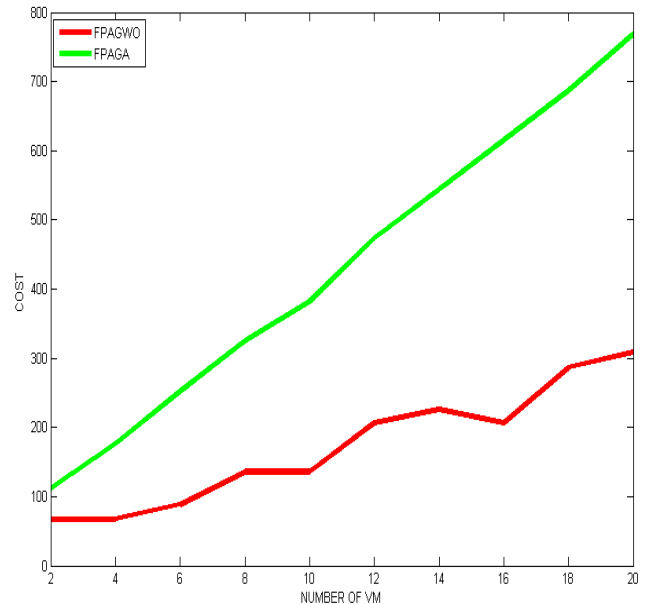


Figure 6. Cost on Genome workflow for FPA_GWO and FPA_GA

Figure 6 depicts the cost of Genome workflow for FPA_GWO and FPA_GA. The green curve shows the cost of FPA_GA and the red curve shows the cost of FPA_GWO. The cost of proposed FPA_GWO is less than FPA_GA.

Results with Sight

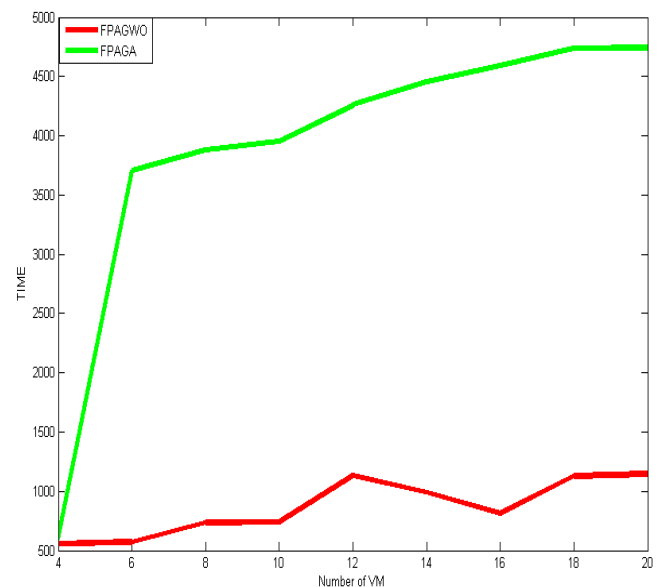


Figure 7. Time on Sight workflow for FPA_GWO and FPA_GA

Figure 7 depicts the cost of Sight workflow for FPA_GWO and FPA_GA. The green curve shows the cost of FPA_GA and the red curve shows the cost of FPA_GWO. The cost of proposed FPA_GWO is less than FPA_GA.

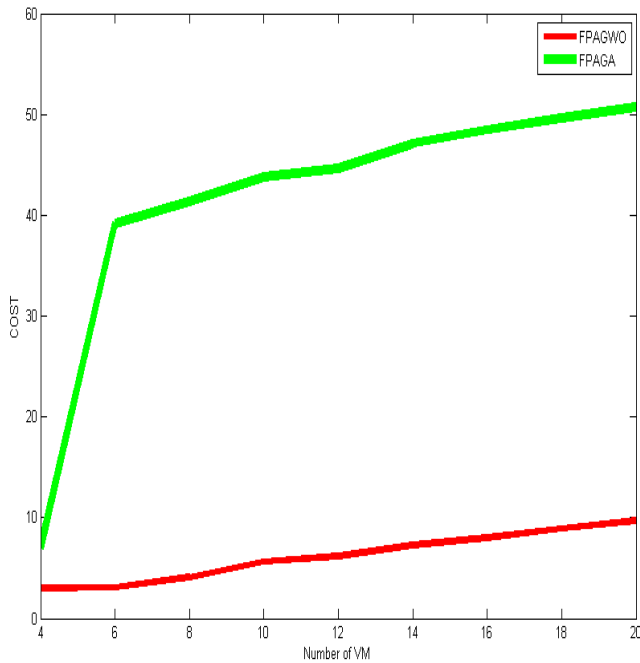


Figure 8. Cost of Sight workflow for FPA_GWO and FPA_GA

Figure 8 depicts the Sight of Genome workflow for FPA_GWO and FPA_GA. The green curve shows the cost of FPA_GA and the red curve shows the cost of FPA_GWO. The time of proposed FPA_GWO is less than FPA_GA.

Reliability

In general, the concept explained above can be represented verbally by an eminent term ‘reliability’, which usually refers person’s quality or entity that is trust worthy. Trust in the informational society is generally built on several distinct grounds on the basis of knowledge, calculus, or on the basis of social reasons [25]. In an organization, the notion of trust could be defined as the certainty of the customer such that the organization is able to provide the needed services infallibly and accurately. The principle of certainty that also helps in expressing the customer’s faith in soundness of its operation, in its expertise, its moral-based integrity, in the mechanism of security-based effectiveness, and in its abidance by the overall laws and regulations, on simultaneous basis, it also carries the affirmation of a minimized risk factor, by the party-based relying.

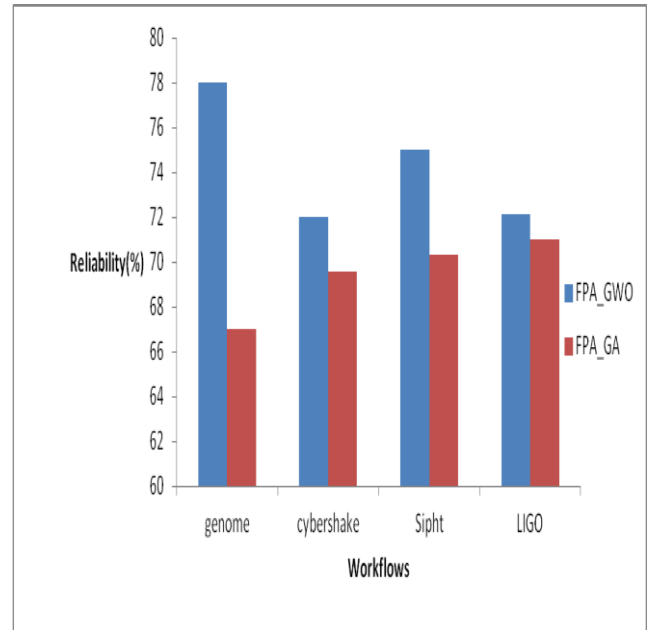


Figure 9. Comparison of Reliability in different workflows

6. Conclusion

The proposed work is based on the effective hybrid optimization approach that is a combination of FPA and GWO algorithm. In this work, two parameters are considered for scheduling that is cost and time. In this algorithm FPA is used because it is a local optimization algorithm and GWO is a global optimization algorithm which optimized the VM globally. On the other hand for comparison FPA_GA algorithm is used which is also a swarm-based algorithm. In this work, when the cost is increased the number of VM is also increased for a few numbers of tasks and workflow at the time of computation. The waiting time is reduced but the cost is enhanced. So, the decision of optimization plays a vital role to balance the VM and time of computation and needs the global and local optimization. For both local and global optimization algorithms like particle swarm optimization (PSO), Ant colony optimization (ACO) is used but they increase the time because of two optimizations.

References

[1] NUAIMI, A. KLAITHEM, ET. AL. (2012) A survey of load balancing in cloud computing: Challenges and algorithms. *Network Cloud Computing and Applications (NCCA), Second Symposium IEEE*, 2012, pp-137-142. 10.1109/NCCA.2012.29.

[2] MARTIN, R., LAMB, D., AND BENDIAB, A. (2010) A comparative study into distributed load balancing algorithms for cloud computing. *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference*.

- [3] KHERANI, FORAM F., AND JIGNESH V. (2014) Load Balancing in cloud computing. *International Journal of Engineering Development and Research*, Vol-2 Issue-1, pp. 907-912.
- [4] VOLKOVA, VIOLETTA N., ET AL. (2018) Load balancing in cloud computing. *Young Researchers in Electrical and Electronic Engineering (EIconRus)*, 2018 IEEE Conference of Russian. IEEE, pp. 387-390.
- [5] DAM, S., ET AL. (2018) An Ant-Colony-Based Meta-Heuristic Approach for Load Balancing in Cloud Computing. *Applied Computational Intelligence and Soft Computing in Engineering*, IGI Global, pp. 204-232.
- [6] KUMAR, N., AND SHUKLA, D. (2018) Load Balancing Mechanism Using Fuzzy Row Penalty Method in Cloud Computing Environment. *Information and Communication Technology for Sustainable Development*. Springer, Singapore, pp. 365-373.
- [7] TRIPATHI, A., SHUKLA, S., AND ARORA, D. (2018) A Hybrid Optimization Approach for Load Balancing in Cloud Computing. *Advances in Computer and Computational Sciences*. Springer, Singapore, pp. 197-206.
- [8] MISHRA, S. K., ET AL. (2017) Time efficient dynamic threshold-based load balancing technique for cloud computing. *Computer, Information and Telecommunication Systems (CITS)*, IEEE Conference, 2017.
- [9] XAVIER, VM ARUL, AND ANNADURAI, S. (2018) Chaotic social spider algorithm for load balance aware task scheduling in cloud computing. *Cluster Computing*, pp. 1-11.
- [10] KANG, S., VEERAVALLI, B., AND AUNG, K. M. (2018) Dynamic scheduling strategy with efficient node availability prediction for handling divisible loads in multi-cloud systems. *Journal of Parallel and Distributed Computin*, Vol. 113, pp. 1-16.
- [11] MAKHLOUF, S. A., AND YAGOUBI, B. (2018) Clustering Strategy for Scientific Workflow Applications in IaaS Cloud Environment. *International Conference Europe Middle East & North Africa Information Systems and Technologies to Support Learning*, pp. 482-491. Springer.
- [12] LIU, L., ZHANG, M., BUYYA, R. AND FAN, Q. (2017) Deadline-constrained coevolutionary genetic algorithm for scientific workflow scheduling in cloud computing. *Concurrency and Computation: Practice and Experience* 29, no. 5 (2017): e3942.
- [13] YAO, G., DING, Y., JIN, Y., AND HAO, K. (2017) Endocrine-based coevolutionary multi-swarm for multi-objective workflow scheduling in a cloud system. *Soft Computing*, Vol. 21, no. 15 (2017): 4309-4322.
- [14] YE, X., LIU, S., YIN, Y., AND JIN, Y. (2017) User-oriented many-objective cloud workflow scheduling based on an improved knee point driven evolutionary algorithm. *Knowledge-Based Systems*, Vol. 135, (2017): 113-124.
- [15] HILMAN, M. H., RODRIGUEZ, M. A. AND BUYYA, R. (2017) Budget-constrained Resource Provisioning and Scheduling Algorithms for Scientific Workflows in Cloud Environments. *In School of Compung and Informa on Systems 5th Annual Doctoral Colloquium 19 July 2017*, pp. 16.
- [16] SAMADI, Y., ZBAKH, M., AND TADONKI C. (2017) Workflow Scheduling Issues and Techniques in Cloud Computing: A Systematic Literature Review. *International Conference of Cloud Computing Technologies and Applications*, pp. 241-263. Springer.
- [17] ZHOU, X., ZHANG, G., SUN, J., ZHOU, J., WEI, T., AND HU, T. (2019) Minimizing cost and makespan for workflow scheduling in cloud using fuzzy dominance sort based HEFT. *Future Generation Computer Systems* 93 (2019): 278-289.
- [18] SHAW, R., HOWLEY, E., AND BARRETT, E. (2017) Predicting the available bandwidth on intra cloud network links for deadline constrained workflow scheduling in public clouds. *International Conference on Service-Oriented Computing*, pp. 221-228. Springer, Cham.
- [19] GOYAL, M., AND AGGARWAL, M. (2017) Optimize workflow scheduling using hybrid ant colony optimization (ACO) & particle swarm optimization (PSO) algorithm in cloud environment. *Int. J. Adv. Res. Ideas Innov. Technology*, 3, no. 2.
- [20] CHANG, Y.S., FAN, C.T., SHEU, R.K., JHU, S.R. AND YUAN, S.M. (2018) An agent-based workflow scheduling mechanism with deadline constraint on hybrid cloud environment. *International Journal of Communication Systems*, 31(1), p.e3401, 2018.
- [21] LATIFF, M.S.A. (2017) A checkpointed league championship algorithm-based cloud scheduling scheme with secure fault tolerance responsiveness. *Applied Soft Computing*, 61, pp.670-680.
- [22] GUPTA, I., KUMAR, M. S., AND JANA, P. K. (2018) Efficient Workflow Scheduling Algorithm for Cloud Computing System: A Dynamic Priority-Based Approach. *Arabian Journal for Science and Engineering*, pp. 1-16.
- [23] MIGLANI, N., SHARMA, G. (2018) An adaptive load balancing algorithm using categorization of tasks on virtual machine based upon queuing policy in cloud environment. *International journal of grid and distributed computing*, 11(11), pp.1-12.
- [24] MANASRAH, A., M., AND ALI, H. B. (2018) Workflow scheduling using hybrid GA-PSO algorithm in cloud computing. *Wireless Communications and Mobile Computing*, Vol-2018, <https://doi.org/10.1155/2018/1934784>.
- [25] SHISHIDO, H.Y., ESTRELLA, J.C., AND ARANTES, M.S. (2018) Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds. *Computers & Electrical Engineering*, Vol. 69, pp. 378-394.
- [26] MIGLANI, N., SHARMA, G. (2019) Modified particle swarm optimization based upon task categorization in cloud environment. *International journal of engineering and advanced technology(TM)*, Vol. 8(4).
- [27] CHENTHARA, S., AHMED, K., WANG, H., WHITTAKER, F. (2019) Security and Privacy-Preserving Challenges of e-Health Solutions. *IEEE access*, Vol 7, PP. 74361-74382, 2019, DOI: 10.1109/ACCESS.2019.2919982
- [28] WANG, H., YI, X., BERTINO, E., AND SUN, L. (2014) Protecting outsourced data in cloud computing through access management. *Concurrency and Computation: Practice and Experience*, Concurrency Computat.: Pract. Exper. (2014), Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.3286.
- [29] KABIR, M.E., MAHMOOD, A.N., MUSTAFA, A.K., WANG, H. (2015) Microaggregation Sorting Framework for K-Anonymity Statistical Disclosure Control in Cloud Computing. *IEEE Transactions on Cloud Computing*, DOI 10.1109/TCC.2015.2469649, pp-1-22.
- [30] SUN, L., WANG, H. ET AL. (2012) Semantic access control for cloud computing based on e-Healthcare. *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design*, pp.512-518.
- [31] KAUR, S., BAGGA, P., HANS, H., KAUR, H. (2019) Quality of Service (QoS) Aware Workflow Scheduling (WFS) in

- Cloud Computing: A Systematic Review. *Arabian Journal for Science and Engineering*, 44:2867–2897, <https://doi.org/10.1007/s13369-018-3614-3>.
- [32] KUMAR, A., AND BAWA, S. (2019) A comparative review of meta-heuristic approaches to optimize the SLA violation costs for dynamic execution of cloud services. *Soft Computing*, <https://doi.org/10.1007/s00500-019-04155-4>.
- [33] OLIVEIRA, D., ET AL. (2019) Performability Evaluation and Optimization of Workflow Applications in Cloud Environments. *J Grid Computing*, <https://doi.org/10.1007/s10723-019-09476-0>.
- [34] KUMAR, A. AND BAWA, S. (2018) Generalized Ant Colony Optimizer: swarm-based meta-heuristic algorithm for cloud services execution. *Computing*, <https://doi.org/10.1007/s00607-018-0674-x>.