WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5879

M-dimension hybrid algorithm for scientific workflow in cloud computing

Zahraa Mehssen Agheeb[1] and Sayyed Majid Mazinani[2,*]

[1]Department of Electrical Engineering, College of engineering, Misan university, Misan ,Iraq
[2]Imam Reza International University, Mashhad, Iran

*Corresponding Author: Sayyed Majid Mazinani

DOI: https://doi.org/10.31185/wjcm.98
Received: January 2023; Accepted: March 2023; Available online: June 2023

ABSTRACT: Cloud computing is developing with growing ubiquity in workflow scheduling, particularly for scientific workflow. With the emergence cloud computing, can benefit from virtually unlimited resources with minimal hardware investment. In the field of cloud computing, scheduling the operations of the Scientific Workflow Application (SWFA) to the available computing resources at the cost of optimizing the implementation of SWFA is one of the most challenging processes of the Workflow Management System (WfMS). Some strategies of cost optimization have been offered for developing SWFS economic nature in cloud computing. Basic paper aim is providing novel M-dimension hybrid algorithm that applies meta-heuristic algorithm like Hybrid Cost-effective Hybrid-Scheduling (HCHS), Completion Time Driven Hyper-Heuristic (CTDHH), genetic algorithm (GA), particle swarm optimization (PSO) applying heuristic algorithms like algorithms of IC-PCPD2 and IC-Loss. Given experimental comparison outcomes, presented technique has proven to show the most efficient outcomes of performance for whole considered experimental scenarios.

Keywords: Cloud computing, Cost optimization, Scheduling, Meta-Heuristic and Heuristic, Workflow.

1. INTRODUCTION

Heterogeneous framework has advanced as the around the world foundation for electronic computing apps' future immersing heterogeneous resources. Heterogeneous framework contains systems of grid, clusters, cloud systems and so on. For keeping complicated technical tests, distributed resources such as scientific instruments, apps, computational devices require being grouped when observing workflow task on heterogeneous systems [1]. Computing of Cloud is a type of novel paradigm of computing which works to solve novel issue that includes various computers for including large system of computing for running several big operations. One of the main topics in cloud computing refers to scheduling of task and effective cloud resources usage [2].

Workflow is the most repeatedly applied model for providing apps and has been widely applied in domains of scientific computing like astronomy, physics, bioinformatics. Scientific workflows' operations are computation-intensive/data-intensive, so need the high-performance computing area for presenting resources of computing. Cloud computing is the big-scale platform of distributed computing that resources of computation are accessible on demand for such apps of workflow [3]. This is popular which scheduling of cloud workflow is NP-complete problem, workflow of scheduling in multi cloud area is more complex [4]. It is due to that in multi cloud area, services are proposed from multiple autonomous cloud IaaS platforms, resources of computing will be gathered as one or more compound services. After that, choosing the optimum services' integration from several platforms of IaaS for meeting needs of QoS is a challenging issue.

Imposing apps of workflow on Heterogeneous system proposes some advantages [1]:

*Corresponding author: smajidmazinani@imamreza.ac.ir

1. being able to build active apps using heterogeneous resources.
2. Full use of resource which happens in particular area for increasing results and costs of running.
3. intervals of running different fields of administrator for getting particular operations of processing.

Different domains of scientific apply workflows for analyzing big data numbers and to effectively execute complicated simulations and tests. The process could be formed as the workflow sharing that in smaller and simpler processes of sub (like operations). Such operations could be shared to several resources of calculation for a more effective and quicker running. Workflow operations' scheduling in shared platforms has been broadly studied over the years [5].

SWFS cost optimization challenge in computing of cloud is the multi-objective cost-aware issue which needs 3 basic natures consideration [6]: (1) various users that often compete for resources in computing of cloud for satisfying limitations of QoS, (2) inter-dependencies between operations of workflow, (3) high communication cost because of inter-dependencies among tasks. Although, taking the whole issue of cost optimization associated to the nature makes process of SWFS complex and needs high calculation resources amount in terms of computational time. The present study shows a novel technique applying 6 algorithms at the same time (4) meta-heuristic CTDHH [7], HCHS [6], PSO, GA, two heuristic IC-PCPD2, IC-Loss).

This paper is formed as follows: In section 2, describes the related works. In section 3 describes the Hybrid Cost-effective Hybrid-Scheduling (HCHS), heuristic algorithms such as IC-Loss and IC-PCPD2 algorithms, and Completion Time Driven Hyper-Heuristic (CTDHH). Next, Section 4 provides the proposed method. Then, in Section 5, the results and discussion of this research paper are provided and finally, Section 7 provides a conclusion.

2. REVIEW OF PREVIOUS WORKS

In this section, we aim to study the work that exists in this area. Most algorithms concentrate on creating near/ near-optimum solutions. For suboptimal group, we recognize 2 techniques applied by researched algorithms. First 2 are heuristic and meta-heuristic strategies.

2.1 HEURISTICS BASED TECHNIQUES

Heuristics. Generally, heuristic is the laws' collection which have target in recognizing the solution for specific issue. These laws are particular to issue and are modeled then the near solution is recognized in the acceptable time frame. For scenario of scheduling mentioned, the heuristic strategy applies information on cloud features and task of workflow for identifying the plan which faces QoS needs of user. They are simpler for implementing and predictable than the techniques based on meta-heuristic [5].

In [8], writers address scientific workflow scheduling issue in actively provisioned commercial cloud areas. Here, it uses scheduling of workflow for obtaining lower cost with good times of response in areas of cloud. They defined novel algorithms, Proportional Deadline Constrained (PDC), Deadline Constrained Critical Path (DCCP). PDC acts with increasing parallelism in workflow by dividing that in logical classes and proportionally sub sharing whole workflow deadline over them. Algorithm of DCCP is like PDC, basic difference is that specifies limited crucial way via workflow doing co-locate tasks which connect on similar example.

In [9] has been presented the active cost-minimization and deadline limited heuristic, JIT-C, to schedule scientific apps in area of Cloud. For sustaining low cost of running, resources are provisioned only before they are required. Deadline meeting aim is obtained via continuous execution tasks controlling and actively making economic decisions of scheduling for subsequent tasks so deadline limitation is not violated. Tests of simulation performed on 4 popular workflows illustrate that compared with other new heuristics, RTC, IC-PCP, RCT, presented algorithm shows the highest hit rate to face deadline.

In [10], writers defined 4-laye system of workflow scheduling. New CWSA scheduling policy was presented to plan apps of workflow in multitenant area of cloud computing. Various performance metrics analysis was performed. The wide simulations were carried out for assessing presented scheduling policy performance. CWSA performance was compared with various policies of scheduling for highlighting performance and presented solution robustness. Achieved outcomes illustrate that CWSA performs better than other policies of scheduling. Significantly, CWSA was illustrated for using resources of computation accurately decreasing idle cloud resource nodes time.

In [11], writers presented the single approach of scheduling for workflows in area of cloud for addressing problems associated to time and cost. Presented algorithm reduces tasks' execution time at various steps according to tasks' height shaping sets of task. Time of completion is decreased to the feasible minimal level in every height, thus, additional resources allocation takes less time and overall completion workflow time is decreased. Presented algorithm needs less time for allocating workflow tasks to virtual machines. Workflows' makespan is importantly decreased compared to other present algorithms and it reduces sum price for task execution taking extra resource just when this is required.

In [12], for resource scheduling in Infrastructure as a Service (IaaS) area of cloud and workflow proposed an augmented shuffled frog leaping algorithm (ASFLA) built technique. ASFLA outcomes shows that last one performs better than other techniques to decrease all regarded workflows performance cost.

2.2 META-HEURISTICS BASED TECHNIQUES

When heuristics are modeled to work best on the particular issue, meta-heuristics are general-aim algorithms modeled for solving issues of optimization. They are higher level approaches which use issue specified heuristics for recognizing near-optimum solution to the issue. In comparison with algorithms based on heuristic, meta-heuristic strategies are more computationally intensive in general and take longer to execute; although, they want to identify more desirable plans as they recognize various solutions applying guided search. Applying meta-heuristics for solving workflow scheduling issue in clouds includes issues like shaping theoretically unbound number of resources, describing issues for preventing invalid solutions (like data dependency violations) for making convergence simple, pruning space of search applying heuristics on cloud resource model base [5].

In [13], writers provided GA-ETI, planner for scientific apps for systems of cloud to optimize the execution makespan and monetary cost concurrently. GA-ETI applies increased crossover for integrating genes clusters than shared chromosomes randomly; this uses increment/decrement mutations for adding or eliminating virtual machines from the provided chromosome. The two modifications show decreased inherent randomness in comparison with basic GA. GA-ETI solutions had lower monetary cost and makespan in comparison to solutions presented by HEFT.

In [14], aim is reducing workflow makespan and cost under limitation of reliability. Because of various cloud platforms failure coefficients, this is essential for cloud users to consider mapping kinds of VM to tasks for whole reliability of workflow. For solving previous issue, presented algorithm of MOS, given the optimization of particle swarm, takes location of tasks execution and tasks data transmission order at the same time. Outcomes of simulation, on real-world scientific workflow structures base, illustrate that algorithm of MOS performs better than algorithms of RANDOM and CMOHEFT on whole multi-objective metrics of performance.

In [15], writers take static scheduling problem for real-time workflows in systems of cloud. They try to reduce cost of execution when promising limitation on makespan. The algorithm could set computing resources adaptively occupied when process of planning in the iterative way. PSO is used in every iteration and iteration stops while VMs number applied is fixed. Algorithm gets rid of basic resource pool dependence. Moreover, fine-grained billing technique is used related to trends of market.

In [16], developed WFSACO version provided for planning workflow applying ACO reduces makespan and decreasing WFSACO complexity. At first, tasks in every workflow class are organized given the length of task and child tasks number. Organized tasks are mapped to resources applying ACO. For decreasing makespan, classes of pheromone and heuristics data are updated for every machine, given the feasibility of transition as each assumption of presented strategy. Presented strategy empirical outcomes' comparison with that of other present algorithms is being worked out with other metrics like resource cost, planning time.

In [17], new meta-heuristic technique based on Discrete binary cat swarm optimization (DBCSO) is offered which tries to achieve the optimized makespan when running apps of workflow in area of cloud. Outcomes of Simulation illustrate that DBCSO provides optimized makespan and this is observed which carries out better for big tasks' number than standard particle swarm optimization (PSO) and binary particle swarm optimization (BPSO).

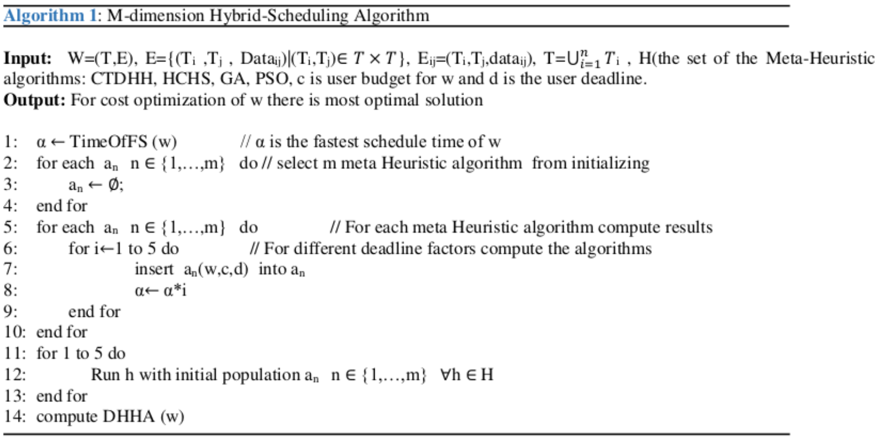
In [18], writers present algorithm of multi-objective gene expression programming based on indicator. For every resource pair and task, amount of superiority is computed by the operation of superiority. The pair with the highest superiority is chosen at every stage. To look for the best function of superiority, some low-level heuristics are described at first and applied as construction blocks to build last heuristics. After that, they combine the technique multi-objective optimization with a recent publish GP variant (called SL-GEP) based on indicator for looking for several heuristics which have various trade-offs among workflow execution time and cost.

In [19], writers have presented the novel strategy of PEFT genetic algorithm for decreasing time of execution on this model. One approach is deployed for allowing GA to concentrate on chromosomes aim optimization for getting the best appropriate mutated kids. After achieving the possible answer, GA concentrates on execution time optimization. Outcomes illustrate that PEFTGA outperforms operations' plans on virtual machines in makespan terms. Time of completion (makespan) for presented algorithm of PEFTGA is decreased by medium 25% in comparison with standard GA.

In [20], writers provide the adaptive privileged multi-objective workflow scheduling algorithm for area of cloud computing. This optimizes aims such as cost and makespan. Most present task takes a single function of fitness; they count on function of fitness given the two sum cost and makespan. This is monitored that adaptive privileged multi-objective workflow scheduling algorithm obtains better outcomes than several present algorithms of workflow scheduling. This balances load on resources of compute spreading functions on accessible resources.

3. THE PROPOSED ALGORITHM

The idea provided in [6, 7] is using several meta-heuristic algorithms and has selected one of such algorithms for execution in every step. One of the issues in [7] refers to high order algorithm times because of lack of attention to basic algorithms population. For solving the issue, presented method is using heuristic algorithms for developing basic algorithms population for obtaining better outcomes at shorter order times. For such aim, algorithms of IC-PCPD2 and IC-Loss have been applied for generating basic population. General presented technique algorithm is as:



Algorithm 1: M-dimension Hybrid-Scheduling Algorithm

Input: W=(T,E), $E=\{(T_i, T_j, Data_{ij}) | (T_i,T_j) \in T \times T\}$, $E_{ij}=(T_i,T_j,data_{ij})$, $T=\bigcup_{i=1}^{n} T_i$, H(the set of the Meta-Heuristic algorithms: CTDHH, HCHS, GA, PSO, c is user budget for w and d is the user deadline.
Output: For cost optimization of w there is most optimal solution

```
$\alpha \leftarrow$ TimeOfFS (w)          // $\alpha$ is the fastest schedule time of w
for each $a_n$  $n \in \{1,\ldots,m\}$  do // select m meta Heuristic algorithm  from initializing
    $a_n \leftarrow \emptyset$;
end for
for each $a_n$  $n \in \{1,\ldots,m\}$  do            // For each meta Heuristic algorithm compute results
    for $i \leftarrow 1$ to 5 do            // For different deadline factors compute the algorithms
        insert  $a_n(w,c,d)$  into $a_n$
        $\alpha \leftarrow \alpha * i$
    end for
end for
for 1 to 5 do
      Run h with initial population $a_n$  $n \in \{1,\ldots,m\}$  $\forall h \in H$
end for
compute DHHA (w)
```

Presented algorithm needs 2 basic incomes for the accurate working. Incomes include: (i) W=(T,E), that T (vertex) is functions' collection and E (edges) is directed edges collection among functions and (ii) H(Heuristic algorithms collection: GA, CTDHH, HCHS, particle swarm optimization (PSO)). As illustrated in Algorithm 1, the first presented technique basic step is executing every 4 used meta-heuristic algorithms (like PSO, CTDHH, HCHS, GA). On the other hand, presented algorithm executes every (h) developed Heuristic algorithms (H) for planning submitted workflow functions given the accessible VMs for 4 times for every specific scenario, that (h) shows algorithm of Low Level Heuristic (LLH) that is a part of (H) collection.

Firstly, basic population is generated applying 2 algorithms: IC-PCPD2 (with user deadline limit), algorithm of IC-LOSS (with user budget limit). By generating basic population, the run time of algorithms are decreased in presented algorithm.

3.1 USING IC-PCPD2 ALGORITHM FOR INITIAL POPULATION

IC-PCPD2 [21] is 2-step algorithm which has the same structure to PCP algorithm with 2 basic differences. At first, 3 way determining policies have been replaced with unique novel policy for adapting to novel model of cost pricing. Secondly, step of scheduling is changed in a way for planning the function, at first this attempts in using left present computation services time samples, in a case this fails, it takes launching as the novel sample for running a function before the sub deadline. In other words, ICPCP is the 1-step algorithm that applies an approach like PCP deadline distribution step, however instead of determining sub deadlines to partial critical path tasks, it attempts in planning them by identifying (novel/current one) computation service sample that could run whole way before its latest last time.

3.2 USING THE IC-LOSS ALGORITHM FOR INITIAL POPULATION

IC-LOSS is the changed Loss algorithm version [22] for area of cloud. As in the IaaS clouds replanning personal task on a cheaper machine might raise sum cost of execution, replanning Loss algorithm step must be adapted. IC-Loss attempts in replanning whole sample tasks to a cheaper present/novel sample, for decreasing sum cost of execution. According to that algorithm of IC-LOSS is the algorithm which presents optimum planning according to budget provided by user, the algorithm has been applied for generating basic population.

3.3 COMPLETION TIME DRIVEN HYPER-HEURISTIC (CTDHH ALGORITHM)

CTDHH [7] has been provided for SWFS cost optimization concern in area of cloud. This algorithm is taken as novel developed method which is able to accelerating meta-heuristic algorithm execution time. CTDHH applies High Level Heuristic (HLH) approach using 4 popular meta-heuristic algorithms based on population that work as Low Level Heuristic (LLH). Basic HLH approach aim is to intelligently lead search process given the employed meta-heuristic LLH algorithms performance. CTDHH strategy performance has been widely assessed compared to that with 4 strategies based on population (like GA, PSO, HIWO, IWO) and the present hyper heuristic strategy known as Hyper-Heuristic Scheduling Algorithm (HHSA). Given the lowest obtained completion time, presented strategy actively leads searching processes for identifying the optimum solution by continuously saving scores of computed time (like last runs completion times) of entire LLH algorithms for every taken scenario and after each execution. Accordingly, completion time algorithm based on hyper heuristic becomes more efficient in a way to let reusing and using max employed LLH algorithms strengths to search for optimum targeted cost optimization issue solution. From sum computational cost outcomes, this strategy has obtained the cheapest sum computational cost compared to baseline strategies. Such outcomes are influenced by size and kind of SWFA. It is basically due to complicated and big submitted SWFA workflow tasks size that finally make strategies of SWFS for taking longer time for such tasks' execution. It is because of the reality that Montage SWFA tasks have less precedence limits. So, strategy of CTDHH has obtained the most optimum outcomes for most of the SWFA datasets and for most taken scenarios in comparison to strategies of baseline and HHSA. In our presented technique, we applied the mentioned algorithm building the suitable basic population.

3.4 HYBRID COST-EFFECTIVEHYBRID-SCHEDULING (HCHS ALGORITHM

Strategy of Hybrid Cost-effectiveHybrid-Scheduling (HCHS) [6] is for providers of service for leasing samples of VM from providers of cloud to process hybrid workloads, when facing needed response time for interactive services and batch jobs' deadlines. This paper basic aim is providing the novel strategy that applies a meta-heuristic algorithm like Completion Time Driven Hyper-Heuristic (CTDHH) also by applying 2 heuristic algorithms like algorithms of IC-PCP and IC-Loss, presented technique is using heuristic algorithms for developing basic such algorithms population for obtaining better outcomes at a shorter order times. For this aim, algorithms of IC-PCP and IC-Loss have been applied for generating basic population.

3.5 GENETIC ALGORITHM (GA)

Strategy of GA [23] is one of the basic Evolutionary Computation domain levels. This evolves from population genetics for solving complicated kind optimization issues kind and uses Mendel's genetics structure law for organizing chromosomes, genes, alleles. This includes 2 tasks that are crossover and mutation for recognizing developed solutions. Population includes 2 basic individuals that are genotype and phenotype. Carrying out such tasks in iterated manner outcomes in the adaptive-fit solution. GA aim is increasing candidate solutions payoff in population against the cost function from field of issue. GA approach is repeatedly used surrogates for mutation genetic mechanisms and recombination on candidate solutions population that cost function (called function of objective/fitness) used for decoded candidate governs representation probabilistic contributions provided candidate solution could make to subsequent candidate solution creation.

3.6 PARTICLE SWARM OPTIMISATION (PSO ALGORITHM)

Strategy of PSO [24] is one of the basic swarm intelligence environment levels. This is according to social foraging manner of bird. For instance, birds flocking and fish schooling. For recognizing better solution, 2 basic approaches have been presented: (i) global position, (ii) local position. Swarm maintains whole history of location for future reference. PSO is targeted in identifying optimum solution given whole swarm location info in multi-dimensional hyper-volume. Optimum solution could be achieved by randomly assigning locations and velocities to whole swarms. Strategy of PSO proceeds by incrementing each swarm position given the velocity. Important PSO strategy opinion is looking for a problem space for assigning the popular global location that is known to particle after sampling for every novel update associated to location. Accordingly, particles set together/converge in developed solution connecting better locations exploration and use in search space.

.

4. EVALUATION OF RESULTS

To evaluate and compare the proposed method has been used the algorithms of the paper [6, 7], because this scheduling algorithm has the same criteria with the proposed algorithm. For these comparisons have been tested five workflows Montage, Inspiral, Epigenimics, SIPHT and Cybershake with the available classes in Table 1.

Table 1. Workflow classes

Small	Large	Workflow
25-50	1000	Montage
30-50	1000	Inspiral
24-46	997	Epigenimics
30-100	1000	Cybershake
30-50	1000	SIPHT

As shown in Figures. 1-5, for all Inspiral, Epigenimics, Montage, Cybershake and SIPHT workflows, the performance of proposed algorithm is compared with the basic algorithms (CTDHH, HCHS, GA, PSO).

The experiments show that for a workflow, according to the two algorithms we considered, whatever performance of the two IC- LOSS and IC-PCPD2 algorithms be better in the production of the initial population, the proposed method has a better performance than the base algorithms. For the scheduling of the cloud workflow, our proposed evolutionary algorithm is more stable and seems to be more likely to produce acceptable schedulers.

Table 2. Statistical results of completion time for Inspiral workflow

Workflows	GA	PSO	CTDHH	HCHS	proposed method
inspiral30	1.8	2.21	1.45	1.32	**1.15**
Inspiral 50	2.2	2.5	1.9	1.65	**1.48**
Inspiral 1000	2.43	2.78	2.12	1.8	**1.62**

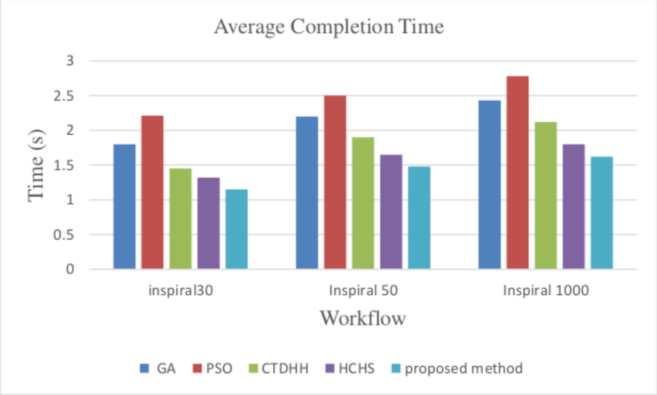


FIGURE 1. Comparison of average completion time in proposed algorithm with other algorithms (Inspiral workflow)

Table 3. Statistical results of completion time for Epigenomics workflow

Workflows	GA	PSO	CTDHH	HCHS	proposed method
Epigenomics 24	2.85	2.59	2.44	2.31	**2.12**
Epigenomics 46	2.69	2.55	2.8	2.25	**2.02**
Epigenomics 997	2.41	2.68	2.74	2.18	**1.92**

Average completion time worth of offered presented technique is better than medium CTDHH and HCHS strategy completion time for whole workflows. It is for applied heuristic algorithms (IC-PCPD2, IC-LOSS) in generating basic algorithms' population (PSO, GA, CTDHH, HCHS) taking for identifying the most optimum solution for workflow cost optimization. As it is obvious in Table 6, proposed algorithm outcomes are better than other algorithms, order of time is decreased for workflow.

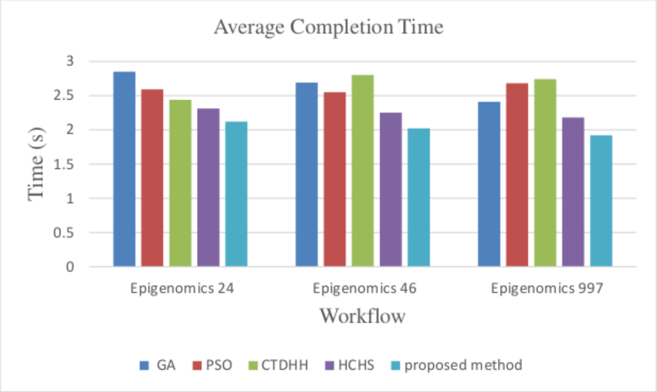


FIGURE 2. **Comparison of average completion time in proposed algorithm with other algorithms (Epigenomics workflow)**

Table 4. Statistical results of completion time for Montage workflow

Workflows	GA	PSO	CTDHH	HCHS	proposed method
Montage 25	2.77	4.5	3.1	2.54	**2.3**
Montage 50	3.24	3.94	3.37	2.69	**2.41**
Montage 1000	3.11	3.48	2.91	2.51	**2.28**

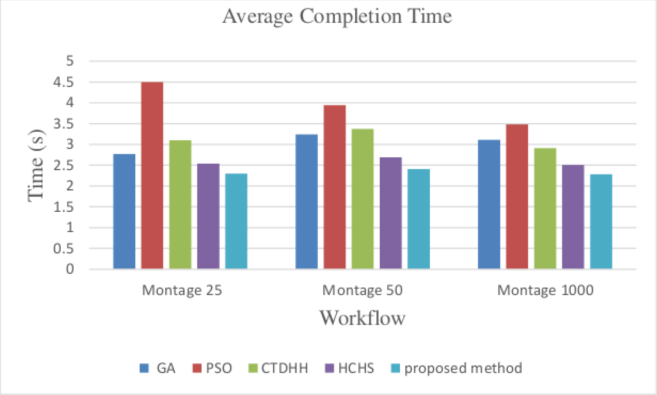


FIGURE 3. **Comparison of average completion time in proposed algorithm with other algorithms (Montage workflow)**

Table 5. Statistical results of completion time for Cybershake workflow

Workflows	GA	PSO	CTDHH	HCHS	proposed method
CyberShake30	4.8	5.4	5.6	3.41	**3.17**
CyberShake100	5.2	4.3	4.87	2.89	**2.56**
CyberShake1000	3.8	3.2	3.7	2.21	**1.92**

Table 6. Statistical results of completion time for SIPHT workflow

Workflows	GA	PSO	CTDHH	HCHS	proposed method
Sipht30	1.01	1.95	1.21	0.89	**0.62**
Sipht100	1.18	1.54	1.14	1.08	**0.91**
Sipht1000	1.32	1.27	1.92	1.22	**1.01**

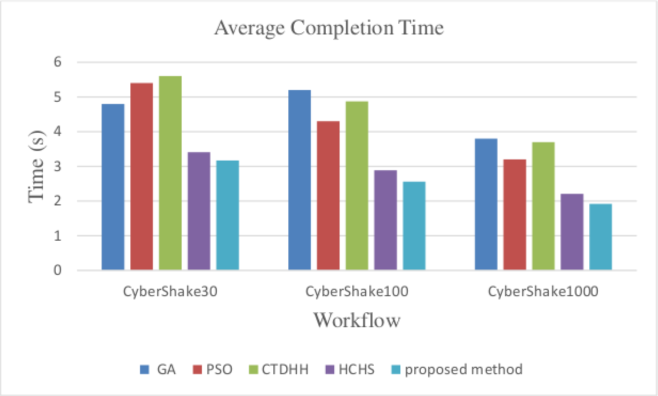


FIGURE 4. Comparison of average completion time in proposed algorithm with other algorithms (Cybershake workflow)

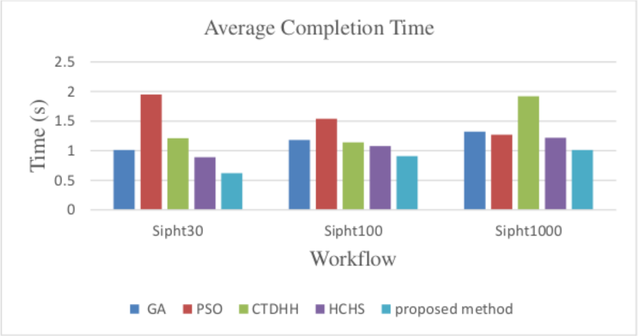


FIGURE 5. Comparison of average completion time in proposed algorithm with other algorithms (SIPHT workflow)

5. CONCLUSION

One of the scheduling approaches is to rely on the quality of service (QoS) the user's desired on scheduling. Quality of service-based workflow scheduling algorithms usually consider a number of service quality parameters, such as cost parameters, task completion deadlines (and in general, parameters in the service level agreement) for scheduling. In this paper is presented a combination of Meta-heuristic and heuristic algorithms for workflows scheduling in a cloud environment. In this way, some heuristic algorithms (IC-PCPD2, IC-LOSS) were used to create the initial input and population of the evolutionary algorithm of PSO, GA, CTDHH, and HCHS. According to the results, the proposed method is in many cases better than the initial algorithms. In the future, we plan to extend the proposed method for multiple workflows and multi-cloud environments.

FUNDING

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] A. Gupta and R. Garg, "In Computing and Communication Technologies for Smart Nation (IC3TSN," *2017 International Conference on*, pp. 319–326, 2017.

[2] V. Vinothina, “Scheduling scientific workflow tasks in cloud using swarm intelligence,” *Current Trends in Advanced Computing (ICCTAC*, pp. 1–5, 2017.

[3] H. Hu, Z. Li, H. Hu, J. Chen, J. Ge, C. Li, and V. Chang, “Multi-objective scheduling for scientific workflow in multicloud environment,” *Journal of Network and Computer Applications*, vol. 114, pp. 108–122, 2018.

[4] R. Tyagi and S. K. Gupta, “A Survey on Scheduling Algorithms for Parallel and Distributed Systems,” in *Silicon Photonics & High Performance Computing*, pp. 51–64, Springer, 2018.

[5] M. A. Rodriguez and R. Buyya, “A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments,” *Concurrency and Computation: Practice and Experience*, vol. 29, no. 8, pp. 4041–4041, 2017.

[6] A. Alhwayzee, “A Cost-effective Hybrid-Scheduling Algorithm for Scientific Workflow in cloud computing,” *Sci.Int. (Lahore)*, vol. 30, no. 4, pp. 575–580, 2018.

[7] E. Alkhanak, S. P. Nabiel, and Lee, “A hyper-heuristic cost optimisation approach for Scientific Workflow Scheduling in cloud computing,” *Future Generation Computer Systems*, 2018.

[8] V. Arabnejad, K. Bubendorfer, and B. Ng, “Scheduling deadline constrained scientific workflows on dynamically provisioned cloud resources,” *Future Generation Computer Systems*, vol. 75, pp. 348–364, 2017.

[9] J. Sahni and D. P. Vidyarthi, “A cost-effective deadline-constrained dynamic scheduling algorithm for scientific workflows in a cloud environment,” *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 2–18, 2018.

[10] B. Rimal, M. Prasad, and Maier, “Workflow scheduling in multi-tenant cloud computing environments,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 1, pp. 290–304, 2017.

[11] G. Reddy, S. Narendrababu, and Phanikumar, “A novel method for scheduling workflows in cloud computing environment,” *Science Technology Engineering & Management (ICONSTEM*, pp. 12–16, 2017.

[12] P. Kaur and S. Mehta, “Resource provisioning and work flow scheduling in clouds using augmented Shuffled Frog Leaping Algorithm,” *Journal of Parallel and Distributed Computing*, vol. 101, pp. 41–50, 2017.

[13] I. Casas, J. Taheri, R. Ranjan, L. Wang, and A. Y. Zomaya, “GA-ETI: An enhanced genetic algorithm for the scheduling of scientific workflows in cloud environments,” *Journal of computational science*, vol. 26, pp. 318–331, 2018.

[14] H. Hu, Z. Li, H. Hu, J. Chen, J. Ge, C. Li, and V. Chang, “Multi-objective scheduling for scientific workflow in multicloud environment,” *Journal of Network and Computer Applications*, vol. 114, pp. 108–122, 2018.

[15] P. Guo and Z. Xue, “An adaptive PSO-based real-time workflow scheduling algorithm in cloud systems,” *2017 IEEE 17th International Conference on*, pp. 1932–1936, 2017.

[16] V. Vinothina, “Scheduling scientific workflow tasks in cloud using swarm intelligence,” *Current Trends in Advanced Computing (ICCTAC*, pp. 1–5, 2017.

[17] B. Kumar, M. Kalra, and P. Singh, “Discrete binary cat swarm optimization for scheduling workflow applications in cloud systems,” *Computational Intelligence & Communication Technology (CICT), 2017 3rd International Conference on*, pp. 1–6, 2017.

[18] Xiao, J. Qin-Zhe, W.-N. Zhong, Z.-H. Chen, J. Zhan, and Zhang, “Indicator-based multi-objective genetic programming for workflow scheduling problem,” *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 217–218, 2017.

[19] R. Nagar, K. Deepak, R. M. Gupta, and Singh, “Time Effective Workflow Scheduling using Genetic Algorithm in Cloud Computing,” *Information Technology and Computer Science*, vol. 1, pp. 68–75, 2018.

[20] A. Singh, “A multi-objective workflow scheduling algorithm for cloud environment,” *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*, pp. 1–6, 2018.

[21] S. Abrishami, M. Naghibzadeh, and D. H. Epema, “Deadline-constrained workflow scheduling algorithms for infrastructure as a service clouds,” *Future Generation Computer Systems*, vol. 29, no. 1, pp. 158–169, 2013.

[22] R. Sakellariou, H. Zhao, E. Tsiakkouri, and M. D. Dikaiakos, *Scheduling workflows with budget constraints*. Boston, MA: Springer, 2007.

[23] Z. Zhu, G. Zhang, M. Li, and X. Liu, “Evolutionary multi-objective workflow scheduling in cloud,” *IEEE Transactions*, vol. 27, no. 5, pp. 1344–1357, 2016.

[24] M. A. Rodriguez and R. Buyya, “Deadline based resource provisioningand scheduling algorithm for scientific workflows on clouds,” *IEEE transactions on Cloud Computing*, vol. 2, no. 2, pp. 222–235, 2014.