Optimal Workflow Scheduling In Cloud Computing Using AHP Based Multi Objective Black Hole Algorithm

Fatemeh Ebadifard
Department of Computer,
University of Kashan, Kashan, Iran
Ebadifard.fatemeh@gmail.com

Seyed Morteza Babamir
Depertment of Computer
University of kashan, Kashan, Iran
Babamir@kashanu.ac.ir

***Abstract*— Cloud computing is one of the fields that has attracted a lot of attention in recent years. Task scheduling meaning the proper permutation of user requests on virtual machines is one of the most important challenges in the cloud environment due to the increase in virtual machines as well as the diversity of users with different service quality requirements and therefore task scheduling is a NP-hard problem. This becomes more complicated when quality of service objectives conflict with each other; therefore, service providers for the proper use of cloud environment capabilities require an optimal trade-off between the various objectives and in such cases heuristic algorithms can be used for optimal scheduling. To this end, we extended a recent heuristic algorithm called Black hole and considered dependency graph of workflow tasks. The proposed method combines the heuristic algorithm and decision-making method (AHP) to solve the multi-objective workflow scheduling problem on virtual machines. We converted the single-objective Black-hole algorithm into a multi-objective by using the AHP relationship, and then it is used to solve the scheduling problem. We have implemented our proposed method using the Workflowsim tool and have compared the results with multi-objective algorithms SPEA2 and NSGA2 based on the parameters of Makespan and cost and resource utilization using a balanced and unbalanced workflow.**

Keywords- ***Cloud Computing; Workflow; Scheduling; Multi objective; Decision-Making method ;AHP.***

I. INTRODUCTION

Workflow is a common model for modeling most scientific programs which contains a number of tasks and data dependencies between tasks. Since most tasks in applications are a set of workflows; in recent years, extensive research has been done on the workflow scheduling in the cloud environment. Workflow scheduling on resources is to select the appropriate virtual machine for a task, so that its related tasks are already executed. This selection of resources and assignment of task on them depends on the requirements of quality of service for different users and since there are different permutation of requests on virtual machines, task scheduling is a NP-hard problem [1].

A great related work has already been done on the workflow scheduling problem in the cloud environment; in most classic work, it is tried to reduce finish time, but in recent ways; in addition to Makespan, there are other objectives, such as resource utilization and cost for scheduling. And they seek to provide an optimal permutation of requests on virtual machines, in light of these objectives [2]. This problem is presented as multi-objective scheduling, and there are different approaches to solve it. One of these methods is the use of Pareto optimal solutions. This solution allows users to have the best choices from the set of appropriate solution and thus provide a set of ultimate solutions that include optimal tradeoff between some of the QoS objectives.

In this paper we have expanded our previous work [3] which is based on the black hole algorithm [4] and it is a heuristic optimization method which can be used as an appropriate solution to the scheduling problem due to its simple structure and lack of dependence on external parameters and its high performance.

The extension is carried out in 2 aspects: (1) making our previous work an AHP-based algorithm and using this feature to select the optimal solutions. To this end, we apply combination of multi-objective black hole approach and Decision-Making method (AHP) [5] for workflow scheduling optimization. So, we have been able to develop the domination relationship in the multi-objective algorithm by using AHP method, as a result we could choose better solution according to the preference of users in the Pareto front.

(2) Presenting resource utilization objective to consider provider’s preferences in selection of the best solution among the optimal solutions. So that the proposed method can consider the quality of service requirements for service provider and the client simultaneously.

Our main contributions can be summarized as follows:

1: Presentation of the multi-objective Black hole algorithm using the single-objective Black hole algorithm and Pareto optimizer and applying it for the multi-objective workflow scheduling in the cloud environment

2: Using the AHP technique in multi-objective domination relationship and taking into account the preferences of users in the solution of workflow scheduling

3: Targeted evaluation and analysis to illustrate the effective combination of the black hole algorithm with AHP technique to reduce Cost of resources and makespan and increase resource utilization.

We have used the WorkflowSim tool [6], which is an extension of CloudSim [7] open source tool, to evaluate our proposed method. We have developed the initial core of this tool to provide our algorithm and then compared our proposed method with previous Pareto-based algorithms such as SPEA2 [8], NSGA2 [9].

The rest of the paper is comprised of the following sections: Section 2 presents the related work of workflow scheduling. In Section 3, the mathematical model of the workflow scheduling problem and the details of the optimization objectives used have been provided and in the fourth section, we first introduced the single-objective black hole algorithm and the AHP technique and then we present the details of the proposed AHP-based multi-objective algorithm, in the fifth part, we have described the details of the evaluation of the proposed method and analyzed the results. Finally, in the final section, conclusions and future work have been presented.

II. RELATED WORK

Many previous work have been done to solve the problem of task scheduling for independent workload and workflow in the cloud environment; since our proposed approach has been provided for the workflow, we investigate the methods of workload scheduling. There are different types of classification for workflow scheduling methods; for example, static or dynamic scheduling or other categorization based on single-objective or multi-objective workflow scheduling. In this section, we examine the multi-objective workflow scheduling methods. In multi-objective scheduling, several objective which are often in contradictory have been considered as optimization objectives and provide the optimal tradeoff of solutions. Multi-objective scheduling methods are in the following categories:

A. Aggregation approach

One of the multi-objective scheduling methods is to convert a multi-objective problem to a single objective which is done by weighting objectives. In Li et al. [10], have converted the multi-objective problem into a single objective by using the heuristic method CCSH and have provided a scheduling method with cost and Makespan reduction objectives. Dongarra et al. [11] have proposed a scheduling method by using aggregation technique to increase performance and reliability. They have provided (RDLS) a reliable dynamic level scheduling algorithm based on the DLS algorithm [12]. Dogan et al. [13] have improved their method using genetic algorithm and BDLS scheduling method.

B. Pareto based approach

Unlike the previous method, in which only one definitive solution is presented as the result of the algorithm. In these methods, a set of non-dominated solutions is provided that allows users to choose a solution based on their expected QoS. Yu et al. [14] have used multi-objective evolutionary algorithms (MOEAs) to solve the workflow scheduling problem. This approach is aimed at reducing two conflicting objectives of cost and runtime. In addition to these two purposes, they also considered deadline and budget constraints in the algorithm and therefore provided fitness functions corresponding to objectives and constraints. The population-based algorithms SPEA2 and NSGAII [8, 9] and local search-based algorithms MOEA, PAES [15] have been provided for workflow scheduling based on different objectives and constraints. Various scheduling methods have been proposed for cost and Makespan reduction using heuristic algorithms and Pareto optimization; such as Udomkasemsub et al. [16] using ABC and WU [17] using the RDPSO algorithm. Most of the previous methods have only considered two objectives of cost and makeapn reduction as optimization objectives; Khalili et al. [18], in their proposed method, in addition to Makespan and cost, also considered the purpose of resource efficiency and using a gray wolf algorithm, they have introduced a new Pareto-based multi-objective scheduling method. In our proposed approach, in addition to reducing the cost, Makespan that benefits of the users are supplied; increased resource utilization for the benefit of service providers has been considered. The difference of our work with previous methods is to use the new Black Hole heuristic algorithm, which is more efficient than other algorithms and converting it to a multi-objective algorithm is done using the Pareto optimizer. Also, in the proposed algorithm by combining black hole algorithm with the decision-making method (AHP), the QOS preferences are considered by the user during the execution of the algorithm and this will reduce the search space of the problem and choose the more optimal solutions with increasing requests and the variety of virtual machines.

III. PROBLEM FORMOULATION

A set of tasks and edges for communication between requests is defined as workflow and in this workflow child request is not allowed to be execute as long as the parent is not executing. Figure 1 illustrates an example of a workflow in which each node represents a task and edges are the connection between tasks and the number above each edge shows the estimated cost for data transfer between the corresponding tasks. In the following, each of the objectives that we have used in the workflow scheduling problem will be explained.

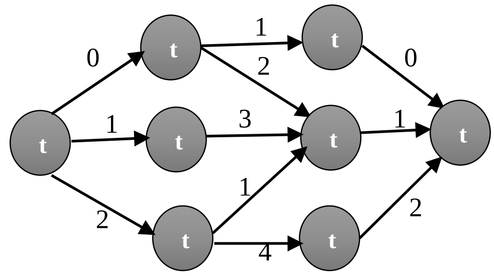


Figure. 1. The workflow sample

A. Makespan

To formulate the problem we have denoted a set of tasks Task = $\{t_1, t_2, ..., t_n\}$ where $i \in \{1, 2, ..., n\}$. We have defined a set of m virtual machines, VM = $\{VM_1, VM_2, ..., VM_j\}$ interconnected by network where $j \in \{1, 2, ..., m\}$. The tasks will be processed on virtual machines. Completion time of task T_i on virtual machine VM_j are denoted as CT_{ij} respectively. Overall task completion time is called makespan and is defined by Eq. (3-1)

$$\text{(3-1)} \quad Makespan = \max_{1 \le j \le m} \sum_{i=1}^{n} CT_{ti}^{VM_j} \times x_{ij}$$

If the request t_i is running on VMj, value of x_{ij} is equal to 1 otherwise is zero. Figure. 2 epitomizes a sample scheduling makespan for 6 tasks and 3 VMs.

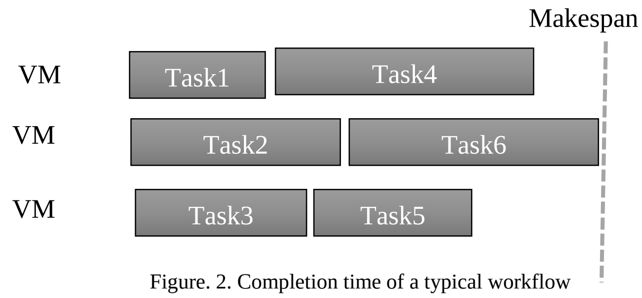


Figure. 2. Completion time of a typical workflow

B. Cost

In cloud computing, computational cost for each customer is calculated based on the timespan they use of resource at any time. Total cost for each request include:

B.1. Computational cost:

This cost is calculated based on execution time of the request t_i on the VM_j and cost per processing in VM_j by Eq.3-2 and the execution time of the request t_i is calculated by Eq.3-3.

$$(3\text{-}2)\quad C_p(t_i) = ET_{ti}^{VMj} \times \mathrm{Cos}\,tPer\,\mathrm{Pr}\,occes\,\mathrm{sin}\,ginVmj$$

$$(3\text{-}3)\quad ET_{t_i}^{VM_j} = \frac{MI(t_i)}{MIPS(VM_j)}$$

While MI (t_i) is the number of instructions of request i and MIPS (VM_J) is the number of million instructions that the machine j executes per second.

B.2. Cost per storage:

This cost is calculated based on the time that this task is placed on the virtual machine and it is calculated according to Eq.3-4.

$$(3\text{-}4)\quad C_s(t_i) = (ET_{ti}^{VMj} + WT_{t_i}^{VMj}) \times \mathrm{Cos}\,tPerStoragInVMj$$

$WT_{t_i}^{VMj}$ has been the waiting time of request t_i on the virtual machine VMj which has depended on the time for providing required files from its parent and is calculated according to Eq. 3-5.

$$(3\text{-}5)\quad WT_{ti}^{VMj} = \frac{\max input(t_i)}{BW}$$

B.3. Cost per transfer:

This costs is dependent on the cost that request ti should pay for the transfer of files to their children which is calculated according to Eq. 3-6.

$$(3\text{-}6)\quad C_T(t_i) = \frac{\sum Output(t_i)}{BW} \times \mathrm{Cos}\,tPerTransfer$$

The total cost is calculated by Eq. 3-7. Cloud provider calculates the cost of request ti based on the total cost presented below.

$$(3\text{-}7)\quad C_{total}(t_i) = C_p(t_i) + C_s(t_i) + C_T(t_i)$$

C. Resource utilization:

For each virtual machine, resource utilization was defined as the Eq.3-8. Where x_{ij} is equal to 1, if the request t_i is executing on VMj otherwise is zero.

$$(3\text{-}8)\quad Utilization_{VM_j} = \frac{\sum_{i=1}^{n} PT_{ij}}{Makespan}$$

IV. Proposed Method

Heuristic algorithms have been used effectively in most optimization issues. Among them, the Black Hole heuristic algorithms can provide more optimal solutions for workflow scheduling problem in terms of its appropriate structure and efficiency compared to the PSO and GA algorithms. In this section, we have first introduced the standard black hole algorithm and then, after expressing the AHP technique, we present the details of the AHP-based multi-objective black hole algorithm.

A. Standard Black Hole Algorithm

The black hole algorithm is a population-based heuristic algorithm which was first presented by Mr. Hamtlou In the first step, this algorithm generates an initial population from candidate solutions (stars) randomly in a sample space. In the workflow scheduling problem on virtual machines, each solution is a permutation of requests on virtual machines. The value of the fitness function is calculated for each star and the star with the best value for the objective function is selected as Black Hole. The black hole is capable of absorbing the stars that have trapped it. After the black hole absorbs the stars around it, the remaining stars move toward the black hole. The motion of the stars towards the black hole is based on Eq. (4-1).

(4-1)

$$x_i(t+1) = x_i(t) + rand \times (x_{BH} - x_i(t)) \qquad i = 1, 2, \ldots, N$$

Where x_i is the position of the star i at time t and t + 1. x_{BH} is the position of the black hole in the search space and rand is a random number in the range [0,1] and N is the number of stars (candidate solutions).

This evolution of solutions continues until the algorithm termination condition is reached and at any stage, if a star has a better fitness function than a black hole, then that star replaces the black hole. The black hole algorithm can prevent being trapped in the local optimal due to the elimination of the available stars in the range of the best solution and selection the alternate star randomly in search space. The radius of the horizons in the black hole algorithm is calculated using Eq. (4-2).

$$(4\text{-}2)\quad R = \frac{f_{BH}}{\sum_{i=1}^{n} f_i}$$

Where f_{BH} is the fitness of black holes and f_i is the fitness function of the star i and n is the number of stars. When the distance between a star and a black hole is less than R, that star is eliminated and a new star is created randomly in search space.

B. *AHP decision making method*

AHP is a multi-criteria decision-making approach which can be used to solve complex decision problems; it was presented by Saaty [5] that has been established based on using a set of pairwise comparisons. The main steps of Analytic Hierarchy Process (AHP) includes making a hierarchy, Assign weight to each metric, investigate the Consistency check of system and ultimately decision-making (determine the priorities of options)

B.1: Making hierarchy:

In the first step, the decision-making elements and the relationships between them must be known in order to build hierarchy. So the options that have an impact on decision making are at the lowest level Thus objectives that have an impact on decision making are considered at lower ranks and decision making whole objective is considered at top of the hierarchy, and finally decision-making options are placed in the lowest level. In following an example for workflow scheduling problem which QoS criteria includes Makespan, cost and resource utilization, the hierarchy structure is shown in Figure. 3. In this example Studied options are members of the Pareto front, which indeed one of them should be selected.

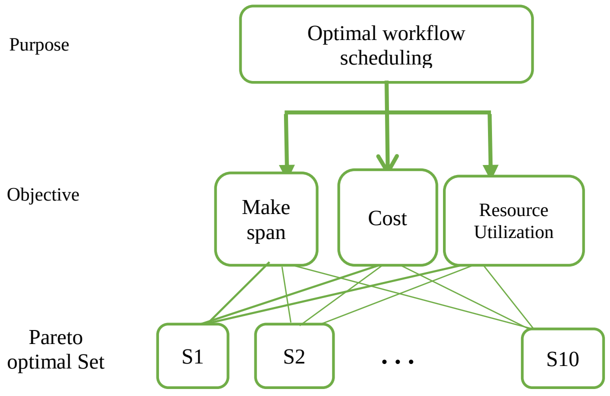


Figure. 3. The AHP hierarchy model

B.2: Weight calculation:

Objectives are compared with each other pairwise to determine the relative priority of each metric (weight), this Pairwise Comparison construct pairwise comparison matrix A. For example, comparing the importance between the Makespan of workflow and its cost indicates that which one has the higher degree of impact in finding the optimal solution. Numbers 1 to 9 are usually used to pairwise comparison. 1 means the same importance and 9 shows the highest degree of importance. Pairwise comparison matrix is shown with $A = (a_{ij})_{m\times m}$ and is as follows:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Where a_{ii}=1 and a_{ij}= $\dfrac{1}{a_{ji}}$

And m is equal to the number of objectives that in this case study is equal to the number of the optimization objectives means 3.

To calculate priority, we use the concepts of Normalization and weighted average. As first the geometric mean of each row has been calculated and then we do normalization. The values obtained from calculations that make up the priority column are called eigenvector (λ). Similarly, we perform priority determination for Pareto optimal set for each criterion.

B.3: Consistency check:

Consistency check is one of the advantages of AHP which aims to testing coordination of the important degree between each metric. The concept of consistency can be expressed as if objective A is more important compared to B and objective B is more important compared to the C, the consistency check is established if the A is more important than C . The consistency index (CI) is shown by Equation (4-3)

(4-3) $$CI = \frac{\lambda_{\max} - m}{m-1}$$

And consistency ratio is calculated by Equation (4-4)

(4-4) $$CR = \frac{CI}{RI}$$

In Equation (4-4) RI value is obtained from the random consistency index table. If value of CR is CR<0.1 consistency is acceptable otherwise contents of Matrix A should be revised.

B.4. Decision making and selecting definitive solution:

The final step of analytic hierarchy process includes determining the importance of each decision-making option in relation to the criteria and general purpose of desired problem. The final weight of each option is calculated of sum of multiplying the priority of objective in weight of options in accordance with the Equation (4-5)

(4-5) $$P_i = \sum_{i=1}^{m} W_i \times G_i$$

That G_i is weight of options and W_i is weight of the objective. In this study we have considered the same weight for options that those options are Pareto front in workflow scheduling problem. Members who value of P_i are less than other options are more appropriate than the rest of the members. It should be noted, the inverter form of resource utilization has been used because Makespan and cost should be minimized and resource utilization should be maximize in the optimization of the scheduling problem.

C. *AHP base multi objective black hole*

Pareto optimal set in heuristic multi-objective algorithms often involves non-dominated solutions and an optimal tradeoff between objectives. Users decide between the optimal solutions, based on their preferences between objectives and choose the right solution which includes permutations of requests on virtual machines. In such cases, in order to choose the right solution among the Pareto optimal set, we need to have a precise weight of the objectives corresponding to user preferences. Without the precise weight of the objectives, choosing the solution proportional with the user's preferences is difficult. That's why users after finding out the set of non-dominated solutions, use AHP-like techniques to choose the optimal solution that suits the user's preferences. We have also used the AHP technique in our proposed method. The difference in our approach with previous works is that we combine the AHP technique with heuristic algorithm but in the previous methods [19], after finding Pareto optimal set in the heuristic algorithm the AHP technique is used. With increasing requests and the diversity of virtual machines, the space of the problem increases and the number of solutions in Pareto optimal set increases. As a result, the AHP technique may not be able to differentiate between solutions [20].

In this paper, we combine the AHP technique with the Black Hole heuristic algorithm and, as a result, a new domination relationship has been proposed that, in the proposed domination relationship, the fitness value of a solution, in addition to the solution strong in dominating other solutions, is selected based on the amount of final weight of that solution obtained by the AHP technique. The details of the proposed algorithm are described in detail:

We have used the Pareto-optimization method to convert the BH algorithm to PBH algorithm and accordingly, for each star, we have considered two fitness values of R, S. As for each star, based on the number of stars in the archives and the population that this star is dominate, the power of S is calculated. Then, in the next step, we obtain the value of R for each star with respect to Eq. (4-6). If a star exceeds the upper time boundary and the upper boundary of the cost, then the power of R is given a very large number.

$$R(i) = \sum_{\substack{j=1 \\ j\in p+a \\ j\succ i}}^{n} (S_j) \tag{4-6}$$

In the above relation, j>i is the symbol of the Pareto domination and j dominate i.

In fact, fitness R for a member is determined by the power of its dominators in both the population and archive and the lower R value for each star in Eq. (4-6), the greater fitness of the star because that star is dominated by stars that have less power. In the next step the final weight of each star obtained according to the user's objectives and preferences for each objective, which is achieved using the AHP technique. Then the members of the archives and population set are arranged in ascending order based on its R value and stars with the same R value are arranged by AHP weight and star with larger AHP weight is selected. Therefore The members of population and archive sets are primarily arranged based on R, and secondly, based on AHP weight.

The star that has the lowest R amongst all stars in the population and archives is considered as the Black Hole star and then the position of the stars in the initial population is updated according to the position of the black hole. In the next step, equal to the number of members in the archive, members are selected from the top of the arranged list and transferred to the archive set. This cycle of procedures is repeated until the achievement of finish conditions. Non-dominated solutions obtained from solving the multi-objective optimization problem (archives) are often referred to as Pareto fronts. None of the Pareto front solutions are superior to the other and depending on the circumstances, one can consider each as the optimal decision. Algorithm 1 depicts the pseudocode of the AHP Base Multi objective Black hole algorithm.

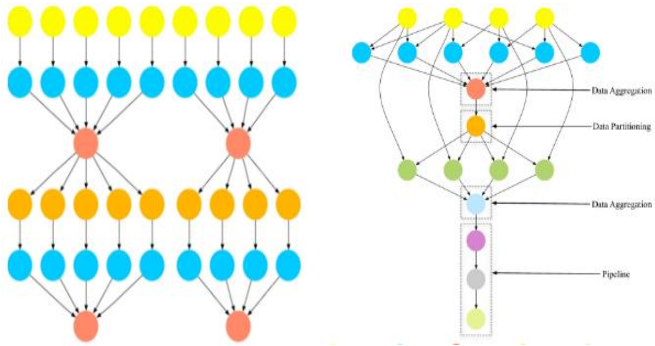
Algorithm1: AHP base multi objective black hole algorithm pseudo code

1. ***Create an initial population of star randomly and an empty archive***
2. ***While (t< Max number of iterations)***
3. ***Calculate the fitness function for all stars based on Eq.3-1, 3-7, 3-8***
4. ***Calculate Pareto based fitness function according to Eq.4-6***
5. ***Calculate final weight of each star by using the AHP technique.***
6. ***Copy top ten non-dominated stars in population to the archive which has the smallest R value and the highest AHP weight value***
7. ***Select the star with the best Pareto base fitness function from the archive (Black hole)***
8. ***For each star***
9. ***Update the position of current star according to Eq.4-1***
10. ***Calculate fitness of Pareto according to Eq. 4-6***
11. ***Calculate the distance between each star and the black hole according to the 4-2***
12. ***If(R< distance)***
13. ***Remove star and create a random star in search space***
14. ***End while***

V. RESULT EVALUATION

To evaluate the proposed methods we have used Open source, WorkflowSim tools, then we have compared our simulation results of the proposed method with known radiation-optimizer-based algorithms SPEA2 [8] and NSGA2 [9].

For evaluation the proposed method we have used a real workflow library presented by Bahraini et al. [21]. We have used balanced (Ligo) and unbalanced workload (Montage) to evaluate the proposed method. Figure.4 shows a small sample of these two workflows.



(a) (b)

Figure. 4. The structure of two realistic scientific workflows[22].(a: Ligo) and (b:Montage)

We have used a data center consists of twenty hosts and 60 virtual machine. Host specification are according to Table 1 and characteristics of virtual machine are presented in Table 2. The input parameters for algorithms PBHO, NSGA2 and SPEA2 are also given in Table 3.

We repeated our experiments ten times and then reported the average data for three factors Makespan, Cost, and resource utilization for two workloads of montage and Ligo in sizes small, medium and large. We have used the decision-making method AHP to weigh the objectives that are used in the proposed multi-objective algorithm. If the importance of each objective to be considered as following, then matrix A will be as follows:

The importance of Makespan in comparison with resource utilization: 5,

The importance of Cost in comparison with Makespan: 3,
The importance of Cost in comparison with resource utilization: 8.

$$A = \begin{array}{r} Makespan \\ Cost \\ THroughput \end{array} \begin{bmatrix} 1 & \frac{1}{3} & 5 \\ 3 & 1 & 8 \\ \frac{1}{5} & \frac{1}{8} & 1 \end{bmatrix}$$

TABLE 1 HOSTS' TECHNICAL SPECIFICATIONS

Host ID	Number of processing Cores	Processing speed (MIPS)	Ram (MB)	Hard (MB)	Bandwidth (Mbps)
1-20	8	2000000	2048000000	1000000	10000

TABLE 2 VIRTUAL MACHINE' TECHNICAL SPECIFICATIONS

Host ID	Number of processing Cores	Processing speed (MIPS)	Ram (MB)	Hard (MB)	Bandwidth (Mbps)
1-15	1	1000	512	10000	1000
16-30	1	1000	512	10000	1000
31-45	2	1000	1024	20000	2000
46-60	4	2000	1024	20000	2000

TABLE 3- THE ALGORITHM PARAMETERS

Parameter	Value
Population Size (PBHO, NSGA2,SPEA2)	50
Archive Size (PBHO ,NSGA2,SPEA2)	10
Maximum Iteration (PBHO ,NSGA2,SPEA2)	20
Maximum Generation (SPEA2, NSGA2)	100
Mutation Probability (SPEA2, NSGA2)	0.5
Crossover Probability (SPEA2, NSGA2)	0.9

Weight of Objectives is obtained as w = (0.2718, 0.6612, 0.0670) and the value of CR = 0.04 <0.1, and therefore compatibility is acceptable. Table 4 shows the values of the Makespan, resource utilization and cost for the Spea2, NSGA2, and Black hole algorithms. Compared with the most existing methods, our approaches which apply AHP on calculating the dominance relation in Black hole are able to meet user's preferences better. Figure 5 shows optimal Pareto front obtained from AHP base multi objective black hole.

TABLE 4.A: RESULT OF PBHO, NSGA2 AND SPEA2 FOR MONTAGE WORKFLOW

Type	Algorithm	Makespan (ms)	Cost $	Resource Utilization%
Small	Spea2	125.21	4082	0.13
	Nsga2	125.49	4083	0.13
	PBHO	123.22	4053	0.13
medium	Spea2	270.54	22033	0.32
	Nsga2	271.84	22299	0.30
	PBHO	257.11	21720	0.34
Large	Spea2	446.1598	45557.59	0.371931
	Nsga2	485.6466	44624.8	0.347917
	PBHO	386.136573	42826.28	0.4073

TABLE4.B: RESULT OF PBHO, NSGA2 AND SPEA2 FOR LIGO WORKFLOW

Type	Algorithm	Makespan (ms)	Cost $	Resource Utilization%
Small	Spea2	3130.951	74027.23	0.103792
	Nsga2	3099.543	74984.73	0.105654
	PBHO	3094.21	72082.71	0.103498
medium	Spea2	3480.52	431702.2	0.464593
	Nsga2	3471.826	428138.6	0.479319
	PBHO	3322.062	418645.5	0.473009
Large	Spea2	3468.511	919475.9	0.960096
	Nsga2	3478.975	925266.1	0.993529
	PBHO	3298.811	903725.8	0.968106

As it is clear from the results in Table 4-a and 4-b, the amount of cost in all the workloads in all sizes in AHP base multi objective black hole algorithm is more than other algorithms. Considering that the importance of cost for the user is more than other objectives, the proposed method chooses solutions with the lowest cost.

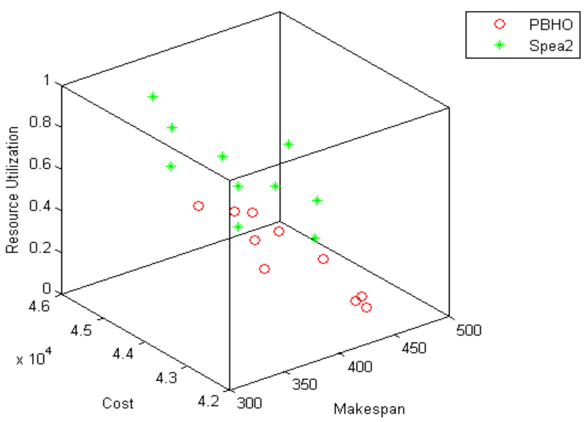


Figure. 5. Pareto front of AHP base multi objective black hole

VI. CONCLUSION AND FUTURE WORKS

Task scheduling is one of the most important challenges for cloud providers and users. In this paper, we present a multi-objective workflow scheduling method by using the Black hole algorithm. Our proposed method, on the one hand, by reducing Makespan and cost supplies the benefit of users and on the other hand, by increasing resource utilization, considers the benefits of service providers simultaneously. In the proposed method, the Black Hole algorithm has been combined with the concept of AHP and the AHP Base Multi objective Black hole has been provided using Pareto optimizer. Since in cloud environments different users require different QoS requirement, in our proposed method, the AHP decision-making technique has been used in Pareto domination relationship, so that user preferences are taken into consideration in choosing the optimal solution at any stage and as a result, by increasing the search space of the problem, we can find a better Pareto optimal set.

In the future, we plan to consider more objectives and examine the effectiveness of this technique for the many objective problems. We can also use the proposed method for dynamic workflow scheduling.

REFERENCE

[1] F. Ebadifard and S. M. Babamir, "A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment," Concurrency and Computation: Practice and Experience, pp. e4368-n/a, Art. no. e4368, 2017.

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

[3] F. Ebadifard and S. M. Babamir, "Optimizing multi objective based workflow scheduling in cloud computing using black hole algorithm," in 2017 3th International Conference on Web Research (ICWR), 2017.

[4] Abdolreza Hatamlou, "Black hole: A new heuristic optimization approach for data clustering", Information Sciences, journal homepage: www.elsevier.com/locate/ins, 2013. Pp.175–184.

[5] T. L. Saaty, "Decision making with the analytic hierarchy process," International Journal of Services Sciences, InderScience, vol. 1, no.1, pp. 83-98, 2008.

[6] W. Chen and E. Deelman, "Workflowsim: A toolkit for simulating scientific workflows in distributed environments," IEEE International Conference on EScience (e-Science), pp. 1-8, 2012.

[7] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya,"CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software: Practice and Experience, vol. 41, no. 1, pp. 23-50, 2011.

[8] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm,"Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.

[9] K. Deb ; A. Pratap ; S. Agarwal ; T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II”, IEEE Transactions on Evolutionary Computation (Volume: 6, Issue: 2, Apr 2002).

[10] J. Li, S. Su, X. Cheng, Q. Huang, and Z. Zhang, "Cost-Conscious Scheduling for Large Graph Processing in the Cloud," in 2011 IEEE International Conference on High Performance Computing and Communications, 2011, pp. 808-813.

[11] J. J. Dongarra, E. Jeannot, E. Saule, and Z. Shi, "Bi-objective scheduling algorithms for optimizing makespan and reliability on heterogeneous systems," presented at the Proceedings of the nineteenth annual ACM symposium on Parallel algorithms and architectures, San Diego, California, USA, 2007.

[12] G. C. Sih and E. A. Lee, "A compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures," IEEE Transactions on Parallel and Distributed Systems, vol. 4, no. 2, pp. 175-187, 1993.

[13] A. Doğan and F. Özgüner, "Biobjective Scheduling Algorithms for Execution Time–Reliability Trade-off in Heterogeneous Computing Systems*," The Computer Journal, vol. 48, no. 3, pp. 300-314, 2005.

[14] J. Yu, M. Kirley, and R. Buyya, "Multi-objective planning for workflow execution on Grids," presented at the Proceedings of the 8th IEEE/ACM International Conference on Grid Computing, 2007.

[15] J. Knowles and D. Corne, "The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation," in Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), 1999, vol. 1, pp. 1-105 Vol. 1.

[16] O. Udomkasemsub, X. Li, and T. Achalakul, "A multiple-objective workflow scheduling framework for cloud data analytics," in 2012 Ninth International Conference on Computer Science and Software Engineering (JCSSE), 2012, pp. 391-398.

[17] Z. Wu, Z. Ni, L. Gu, and X. Liu, "A Revised Discrete Particle Swarm Optimization for Cloud Workflow Scheduling," in 2010 International Conference on Computational Intelligence and Security, 2010, pp. 184-188.

[18] A. Khalili and S. M. Babamir, " Optimal Scheduling workflows in Cloud Computing Environment Using Pareto based Grey Wolf Optimizer" In Concurrency and Computation: Practice and Experience, http://mc.manuscriptcentral.com/cpe, 2016.

[19] A. V. Dastjerdi, and R. Buyya, “Compatibility-aware Cloud Service Composition Under Fuzzy Preferences of Users,” IEEE Transactions on Cloud Computing, vol.2, no.1, pp.1-13, 2014.

[20] L. Liu and M. Zhang, "Multi-objective Optimization Model with AHP Decision making for Cloud Service Composition," KSII Transactions on Internet &Information Systems, vol. 9, no. 9, 2015.

[21] Bharathi, S., Chervenak, A., Deelman, E., Mehta, G., Su, M.H. and Vahi, K.‘‘Characterization of scientific workflows’’, The 3rd Workshop on Workflowsim Support of Large Scale Science, Austin, TX, USA, pp. 1–10 (2008).

[22] Pegasus Workflow Management System, available at:http://pegasus.isi.edu/workflow_gallery/index.php (Last visited: 2017).