BASED ON MULTI-CONSTRAINT PARTITIONING MULTI-OBJECTIVE WORKFLOW SCHEDULING ALGORITHM IN HYBRID CLOUDS

Bin WANG[1*], Yong LIN[2]

In this paper, a multi-objective optimal Heterogeneous workflow Scheduling based on Multi-constraint ParTitioning algorithm (HSMPT) is proposed. The scheduling design considers a large inter-task communication of workflow, security, and cost of public cloud resources in hybrid clouds. The proposed algorithm adopts multi-constraint partitioning mechanism, divides workflow into several sub-workflows considering several characteristics, such as task demand and cloud resource supply, when private cloud cannot meet the deadline requirement for workflow. Then HSMPT allocates cloud resources reasonably for sub-workflows in single cloud according to task resource demand and cloud resource supply characteristics. The WorkflowSim is used to carry out the simulations and simulation results show that, compared with existing greedy and HCOC(Hybrid Cloud Optimized Cost scheduling algorithm) algorithms, HSMPT can effectively reduce inter-cloud traffic and public cloud cost.

Keywords: Multi-constraint partitioning; Inter-cloud traffic; Public cloud cost; Multi-objective workflow scheduling; Hybrid cloud

1. Introduction

With recent fast development of the new generation of information technology, cloud computing model has become the mainstream of computing. Cloud computing [1] which has the advantages of accurate on-demand provisioning and billing by usage, has been used in many institutions as the best environment for application deployment. Scientific workflow [2] is usually used by research institutions to explore the physical world. More and more institutions choose cloud computing to run scientific workflow which draw accurate scientific conclusions through analyze a large number of datasets. Institutions usually build private clouds to maintain their daily computing needs. If the requirements of workflow needed to run increase suddenly, private clouds cannot cope it to satisfy the urgent needs and profit maximization. Therefore, some sub-workflows can be migrated to public cloud resources with more selectivity, reliability, and security. Then, scientific workflow can be executed in hybrid cloud [3]. The deadline in hybrid cloud can be represented by a model including private clouds and public

[1] Chongqing College of Electronic Engineering, Chongqing, China; E-mail: wb548983@163.com
2 Chongqing College of Electronic Engineering, Chongqing, China

clouds. The key problem is to reduce the monetary cost and the traffic between clouds under deadline constraints.

Due to the urgent demand of scientific workflow and the maturity of cloud business model, a series of scientific workflow scheduling algorithms has been studied widely in multi-cloud environments including hybrid clouds [4,5]. Scientific workflow scheduling algorithm in cloud environment consists of two phases, cloud resource allocation and workflow task scheduling, to achieve the objectives such as cost [6-10], makespan [9-10], with the constraints, for example, deadline [6,11], budget [11-12], privacy [13]. Getting the optimal workflow schedule in clouds for multiple objectives and multiple constraints is an NP-hard problem [11]. There are mainly the following types of algorithms, heuristic [12-13] (such as list scheduling and clustering scheduling [13]), meta heuristic [14] (such as Particle Swarm Optimization (PSO), Ant Lion Optimizer (ALO) and Sine Cosine Algorithm (SCA), Grasshopper Optimization Algorithm (GOA), Seagull Optimization Algorithm (SOA)). For the scheduling of a single workflow, Bittencourt etc. [3] firstly proposed a HCOC workflow scheduling algorithm to reduce the invest of public cloud resources on the premise of meeting the deadline of workflow in hybrid cloud. However, it considers little the bandwidth between clouds, while scientific workflow usually is data-intensive application. Especially, when the communication delay between cloud tasks increases, the implementation of follow-up tasks postpones. Lin et al. [15] proposed a HIAP (hierarchical iterative program division) for isomorphic large-scale workflow scheduling is used to cut the workflow into multiple small-scale sub workflows. Considering bandwidth constraints, data transmission cost, and computing cost, priority queue sorting algorithm and indirect to public cloud strategy are applied in combination with minimum load and maximum length to reduce the cost, but the calculation amount and cost of each sub workflow are not balanced [16]. An evaluation mechanism based on hypervolume method is designed, and a multi-objective scheduling algorithm is suggested to reduce the execute time and monetary cost, but public cloud security is not considered [17].

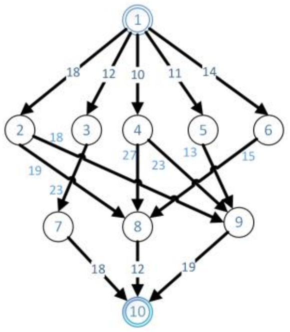
In this paper, we propose a novel multi-objective heterogeneous workflow scheduling based on multi-constraint partitioning algorithm (HSMPT) in hybrid clouds. The purpose of the proposed algorithm is to minimize cost under the constraints of deadline and security. To this end, HSMPT take accounts the following issues: (1) how to partition workflow with many constraints, for example, total computation of sub-workflow, bandwidth of inter-clouds, and the security of public cloud instance; (2) the number of cloud and Virtual Machine to be assigned to the subworkflow, and (3) the order of tasks of subworkflow executed. Firstly, a multi-constraint workflow partitioning which not only considers the communication overhead between clouds, the amount of sub workflow computing, and the security of public cloud resources, but also reduces the minimum data communication between clouds is adopted. Then, according to the deadline of subworkflow and the resource capacity of hybrid cloud, under the condition of meeting deadline constraints and security, virtual machine resources are selected for task scheduling to reduce the execution cost in hybrid cloud. Compared with Greedy and HCOC algorithm, the effectiveness of HSMPT algorithm is verified using WorkflowSim.

The rest of this paper is organized as follows. Section 2 defines hybrid cloud scheduling model and describes the problem formulation. The HSMPT algorithm is detailed in Section 3. Section 4 discusses the evaluation results and section 5 summary this paper and suggest improvements for future work.

2. Hybrid Cloud Workflow Scheduling Model

2.1 Workflow Model

As Fig. 1 shows, scientific workflow is usually represented by directed acyclic graph (DAG). Node vertices represent different tasks, which have different workload, security level and data requirement. Edge represents the data dependency between tasks, which denotes the size of the input or output data for such a dependency. Task execution requires one or more input data, and one or more data will be generated after task execution. It is specified that the task can be executed only after the execution of all precursor tasks is completed, the data required for execution are ready, and the virtual machine assigned to the task is available. Define the task without precursors as a start node and the task without followers as an end node. When there are multiple start nodes, a virtual start node T_{entry} should be added; when there are multiple end nodes, a virtual end node T_{exit} should be added.



Task	Time(s)	SL	Data(MB)
task1	300	1	200
task2	1000	0.8	200
task3	1300	0.6	200
task4	500	0.8	200
task5	1500	0.6	200
task6	700	0.8	200
task7	1400	0.6	200
task8	700	0.8	400
task9	800	0.4	800
task10	400	0.8	200

Fig. 1 Scientific Workflow Model

Workflow contains many types of tasks. Each type of task has different requirements for computing resources, storage resources, IO resources, and security level. The execution time of the task in different virtual machine resources (including calculation cost and data communication cost) are different. A scientific workflow has a submission time when workflow is submitted, and deadline which refers to the latest time when the user specifies that the workflow should be completed. When workflow is submitted, the workflow should be completed before deadline.

2.2 Hybrid Cloud Resource Model

Hybrid clouds (HC) include *m* public clouds and *k* private clouds. C_P represents any cloud service provider. There is a communication link between any two C_P and C_{P}'. Data transmission named $B_{PP'}$ can be carried out. Each C_P provides a cloud resource pool composed of *N* type of virtual Machine which provides different processing capabilities. Each VM has different floating-point computing capabilities, different vCPU cores, different memory size, storage capacity, communication bandwidth, and security level.
The private cloud is built by the enterprise itself. Although the construction and maintenance of the private cloud will incur certain costs, it does not affect the algorithm comparison. It is assumed that the virtual machine provided by private cloud is free. Public cloud will charge for renting virtual machines. In a billing cycle, C_P will charge users according to the performance of virtual machines, and charge cloud users according to the number of billing cycles. If it is less than one billing cycle, it will be charged as a whole. This paper refers to the public cloud in the current market and charges according to the data traffic between clouds, regardless of the data communication cost intra-cloud. It is assumed that the number of available virtual machines in public cloud is more than that in the private cloud, and other costs arising from data persistence, load balancing, and resource monitoring are not considered.

2.3 Problem Description

Our object is to minimize public cloud cost and data transfer across hybrid clouds for scientific workflows. Before giving the definition of the hybrid cloud workflow scheduling problem, we show firstly how to calculate makespan, data transfer, and public cloud cost. Let $\mathrm{BT(Ti)}$ represent the begin time of task Ti. Since $\mathrm{BT(Ti)}$ depends on the end time $\mathrm{ET(Ti)}$ of task Ti of all its predecessors Tj, the communication time $\mathrm{CT(Tj,Ti)}$ between Tj and itself. The available time $avail(VM)$of the previous task of the previous task that has been executed on the same virtual machine, $\mathrm{BT(Ti)}$ is calculated as:

$$BT_{ti} = \begin{cases} 0, if\ ti = entry \\ max\{avail(VM), \max\limits_{tj \in ti's\ pre}(ET(Tj) + CT(\mathrm{Tj,Ti})\} \end{cases} \tag{1}$$

The end time $\mathrm{ET(Ti)}$ of task Ti is calculated as:

$$ET_{ti} = ST(Ti) + RT(Ti,VM) \tag{2}$$

Where $RT(Ti, VM)$ is the runtime of task Ti on scheduled cloud virtual machine instance. Note that the start time of Tentry is submit time, i.e., $\mathrm{BT(Tentry)} = 0$. The makespan defined as the end time of Texit is computed as:

$$Makespan = ET(Texit) \quad (3).$$

Let $\vec{Y}(VMpm, VM'pm)$ denote whether the virtual machine instance $VMpm \text{ and } VM'pm$ belong to hybrid clouds are scheduled. While $VMpm \text{ and } VM'pm$ is both scheduled, $\vec{Y}(VMpm, VM'pm) = 1$. There are two situations for communication between $VMpm \text{ and } VM'pm$. (1) $VMpm \text{ and } VM'pm$ are located in different cloud Cp and Cp, (2) $VMpm \text{ and } VM'pm$ are created in the same cloud Cp. The total data transfer TC across hybrid clouds is calculated as:

$$TC = \sum_{Cp \in HC} \vec{Y}(VMpm, VM'pm), t \in [0, Makespan] \quad (4)$$

When workflow is completed, we defined the cost of all public cloud virtual machine instance $VMpm$, is calculated as

$$CC = \sum_{p \in HC} \sum_{m=1}^{pM} \sum_{i=1}^{N_{pm}} CC(VMpm) * AP(VMpm), t \in [0, Makespan] = \sum_{p \in HC} \sum_{m=1}^{pM} \sum_{i=1}^{N_{pm}} \left\lceil \frac{VFT(Tj) - VST(T'j)}{Tc} \right\rceil * PCpm * AP(VMpm), t \in [0, Makespan] \quad (5)$$

Where $CCpm$ is the total cost of $VMpm$, the virtual machine will be charged according to one time period, and the time less than one billing cycle will be charged according to one time period. $\mathrm{VFT}(Tj)$ is the finish time of last task scheduled on the $VMpm$,while $\mathrm{VST}(Tj)$ is the finish time of first task scheduled on the $VMpm$. While public cloud virtual machine instance $VMpm$ is scheduled, $\mathrm{AP}(VMpm) = 1$. Otherwise, $\mathrm{AP}(VMpm) = 0$.

We define the problem as follows. Give scientific workflow and hybrid cloud, design an algorithm that determines the order of subworkflows and cloud resource provision, so as to minimize cost (given in Eq.(5)) and the total data transfer TC (given in Eq.(4)) while meeting the constraints of deadline and security level of task. Our workflow scheduling problem of minimizing public cloud monetary cost and data transfer is a typical multi-objective optimization problem (MOP), can be formulated as

$$\begin{gathered} minimize \;\; TC, CC \\ \text{subject to Makespan} < \text{Deadline}. \end{gathered} \quad (6)$$

3. The proposed Algorithm

3.1 Multi-constraint Workflow Partitioning

Aiming at the workflow scheduling problem in hybrid cloud, HSMPT algorithm is proposed. The specific steps are as follows, use private cloud workflow scheduling algorithm (PCWS) to schedule workflow with private cloud resources. When private cloud resources cannot meet the deadline, some subworkflows need to be migrated to the public cloud for execution. Workflow is firstly divided according to the current number of available clouds. In order to make full use of private cloud resources and purchase the least public cloud resources, considering the duration, security level consumption, and minimum cloud computing traffic, the workflow is partitioned into subworkflows by multi-constraint segmentation. Calculate the latest finish time LFT of each task under the current hybrid cloud resources and set the deadline of the sub workflow according to the maximum LFT of the sub workflow. LFT(Ti) of task Ti is computed as

$$LFTti = \begin{cases} WDT, & if\ ti = exit \\ \min\limits_{tj \in ti's\ \mathrm{sucs}} \{LFTtj - CTtj - SC_{tj} - BT_{titj}\}, & otherwise \end{cases} \quad (7)$$

HSMPT algorithm is shown in table 1. Firstly, place the subworkflow in private cloud resource. If the finish time of the subworkflow cannot meet the deadline, then select the best public cloud resource to place the subworkflow by public cloud workflow scheduling algorithm (PCWS). According to the public cloud scheduling algorithm, find the cloud resource that can be completed within the deadline constraint, ensure security, and minimize the cost to place the subworkflow. If the public cloud resources cannot meet at this time, more public cloud resources are required to participate in order to complete the subworkflow within the time limit. Therefore, the workflow is divided into three and each subworkflow is allocated with corresponding public cloud resources, until the conditions are met. We note that the number of public clouds is M, the maximum number K of subworkflows is M + 1.

Table 1

Hybrid Cloud Workflow Scheduling Based Multi-Constraint Workflow Partitioning Algorithm

Algorithm 1. Hybrid Cloud Workflow Scheduling Based Multi-Constraint Workflow Partitioning Algorithm	
	//firstly, use private cloud workflow scheduling algorithm to schedule the whole scientific workflow in private cloud
1:	Schedule workflow in the private cloud using PCWS;
2:	K = 1;

```
3:   // If the makespan of workflow is greater than the deadline, Multi-Constraint Workflow
     Division shall be adopted to migrate part of workflow to the public cloud for execution.
4:   while makespan(subWorkflow) > Deadline AND K < M+1
5:     K = K + 1;
6:     using Multi-K-Way to partitioning workflow to subWorkflow;
7:     Compute all tasks deadline assign according to optimal cloud resource, Assign every
     subWorkflow subdeadline;
8:     sort subWorkflow By Ascending SubDeadline;
9:     for each subWorkflow in subWorkflow
     using PCWS to schedule all tasks in private cloud firstly
        if (makespan(subWorkflow) > subDeadline(subWorkflow))
10:          for each cloud in public clouds
              find a minimum cost cloud can finish subWorkflow within SubDeadline
11:             if (found)
12:               using UCWS to schedule all that subWorkflow tasks in found cloud
13:             else
14:               break;
15:             end if
16:          end for
17:       end if
18:    end for
19:  end while
```

Firstly, only private cloud resources are considered, Private Cloud Workflow Scheduling and is used to schedule the workflow, as shown in line 1. If private cloud cannot meet the preset deadline, the workflow is divided into multiple subworkflows, with K starting at 2, as shown in line 5. Select the sub workflow with the smallest deadline for execution. In order to make best use of private cloud resources, the scheduling in the private cloud is firstly considered. If it can be completed in self private cloud, we schedule subworkflow according to the private cloud scheduling process; otherwise, the public cloud scheduling process is used to select the cloud with the lowest cost within the deadline from the public cloud to deploy the current sub workflow. If the current subWorkflow allocates corresponding cloud resources, update other workflow execution. Otherwise, workflow needs more public cloud resources to execute. Jump out of the current cycle, as shown in line 16. Therefore, increase the number of subworkflows sub num by 1 and repeat the above process until the workflow scheduling is completed.

3.2 Private Cloud Workflow Scheduling

The priority Pti of each scheduled task is decided by four factors: execution time of the task; Data communication time; The earliest completion time of all post drive tasks. The priority Pti can be calculated according to formula (8).

$$Pti = \begin{cases} CTti,\ ti\ is\ exit\ node \\ \max_{tj \in ti's\ sucs} CTtj + BTtitj + Ptj,\ otherwise \end{cases} \quad (8)$$

At this time, the security overhead is not considered because the private cloud has the highest security level.

The earliest start time EST(Ti) is computed as:

$$EST_{ti} = \begin{cases} 0, if\ ti = entry \\ max\{avail(Vpmi), \max_{tj \in ti's\ pre} (EFT_{tj} + BT_{titj}^{VpmiVp'm'j})\} \end{cases} \quad (9)$$

The earliest finish time EFT(Ti) is counted as

$$EFT_{ti}^{Vpmi} = EST_{ti}^{Vpmi} + CT_{ti}^{Vpmi} \quad (10)$$

Because the private cloud does not charge fees and all virtual machines in private cloud are secure, PCWS select VM (virtual machine) instance with the best current performance for allocation to obtain the earliest completion time of the sub workflow. The higher the task priority, the earlier the task is scheduled.

3.3 Public Cloud Workflow Scheduling

Due to the public cloud security problem, the security encryption cost SC_{ti}^{Vpm} of task ti running on the instance of public cloud is calculated by

$$SC_{ti}^{Vpm} = \sum_{k=1}^{K} SC_{ti,vpnj}^{ssk} \quad (11)$$

Represents the cost of security service level. So EFT (Ti) is calculated as

$$EFT_{ti}^{Vpmi} = EST_{ti}^{Vpmi} + CT_{ti}^{Vpmi} + SC_{ti}^{Vpmi} \quad (12)$$

UCWS first looks for VM instances that meet the deadline of tasks in the set of allocated virtual machines, so as to reduce the number of VM instances used. If there is a VM, we find the suitable with the lowest running cost and place the task in the VM instance; Otherwise, find a new virtual machine instance and place the task in the VM instance according to the rule of completing within the specified elastic period constraint and minimum cost.

4. Performance Evaluation

In this section, we estimate the performance of proposed HSMPT algorithm using WorkflowSim which is an extended workflow cloud scheduling simulator based on CloudSim. Cloudsim is a simulation tool for the construction of cloud computing system and the resource allocation and scheduling of applications. Workflowsim makes up for the fact that CloudSim does not support workflow scheduling and fault-tolerant scheduling, and provides workflow level functions such as task clustering, resource configuration, and task scheduling. We adopt the real scientific workflow Montage [18], which takes the pictures in the format of elastic image transmission system as the input source to construct the map of the sky, as shown in Fig. 2. Constructing high-quality images consists of four main tasks: reprojection, background radiation modeling, rectification, and synthesis.

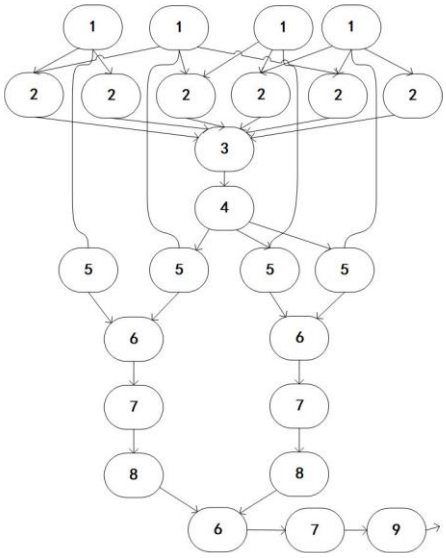


Fig. 2. Montage Workflow

Hybrid cloud environment includes 1 private cloud and 4 public amazon clouds. Private cloud provides two types of VM instance types. The number of vCPU, memory size, and storage capacity are shown in Table 2. Public cloud refers to the virtual machine type provided by Amazon EC2. Amazon EC2 selects calculation optimization and storage optimization instances. The vCPU, memory size, storage capacity, and unit price of each type of virtual machine instance are shown in Table 3. The number of virtual machines is limited to 20. All cloud service providers charge according to the usage of virtual machines. The incoming bandwidth and internal network of Amazon EC2 are free, and only charge for the data flowing out of the current cloud.

Table 2

Instance type of private cloud

Instance type	vCPU	Memory (GiB)	Storage (GB)
small	1	3.5	100
Medium	2	7.0	200

Table 3

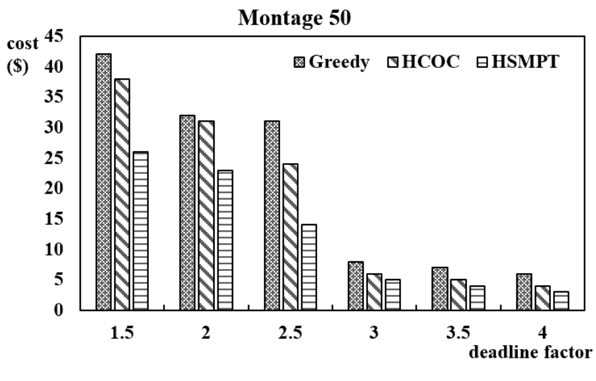
Instance type of public cloud

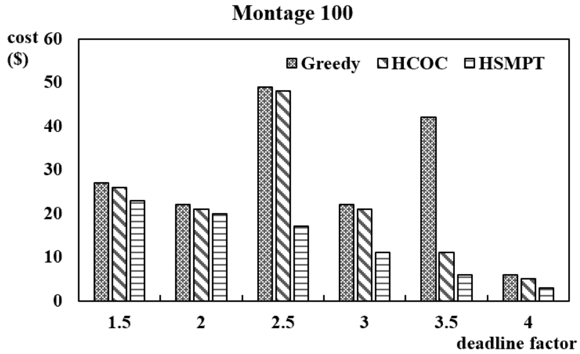
Instance type	vCPU	Memory(GiB)	storage(GB)	cost($/h)
C3.xlarge	4	7.5	80	0.265
C3.2xlarge	8	15	160	0.529
C3.4xlarge	16	30	320	1.508
i2.xlarge	4	30.5	800	1.018
i2.2xlarge	8	61	1600	2.035
i2.4xlarge	16	122	3200	4.07

For the deadline, we select the time used to place the workflow on the fastest virtual machine as the benchmark time. Due to the communication overhead and security overhead, scientific workflow cannot be completed within the benchmark time. We set the deadline span, that is, the deadline is a multiple of the benchmark time, including 1.5, 2, 2.5, 3, 3.5 and 4. The experimental results are the average values obtained after running 1000 times under the specified parameter settings.

4.1 Cost

This paper discusses the execution cost of scientific workflow with different number under deadline span. Fig. 3 shows the execution cost of montage of 25, 50 and 100 tasks under the corresponding deadline span respectively. The execution cost of Greedy and HCOC algorithm [3] is more than that of HSMPT algorithm. The reason is that when the execution time exceeds the deadline, Greedy algorithm selects the virtual machine with the best performance, and its unit price is relatively high, so the execution cost is the highest.





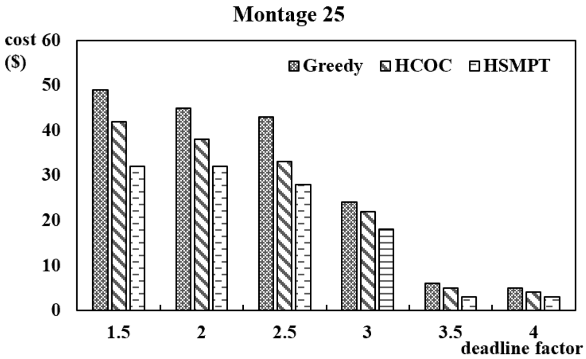


Fig. 3 cost of 25, 50 and 100 tasks montage under different deadline

Both HCOC and HSMPT algorithm select the virtual machine with less cost that can be completed within the deadline to execute the task, so that the task can be completed and reduce the execution cost with deadline. HCOC algorithm does not consider the communication time delay and communication cost between clouds, so the cost of HCOC algorithm is higher than that of HSMPT algorithm. However, because HSMPT algorithm adopts multi-constraint workflow division, the traffic between clouds is reduced. The communication cost of public cloud is reduced, and when selecting the public cost, the virtual machine resources that can be completed within the deadline are selected to deploy tasks. Therefore, HSMPT algorithm can reduce the execution cost.

4.2 Traffic

This paper discusses the cloud traffic of scientific workflow with different number of tasks under different deadline span. Fig. 4 shows the inter cloud traffic of montage with 25, 50 and 100 tasks under the corresponding deadline. Because deadline is small, that is, the urgency of workflow execution is high. More tasks need to be selected to be executed in the public cloud, resulting in very high traffic between the clouds. With the relaxation of the deadline, fewer tasks can be migrated to the public cloud to meet the time constraint, so the traffic becomes less. The inter cloud traffic of HSMPT algorithm is significantly less than that of Greedy and HCOC algorithm.

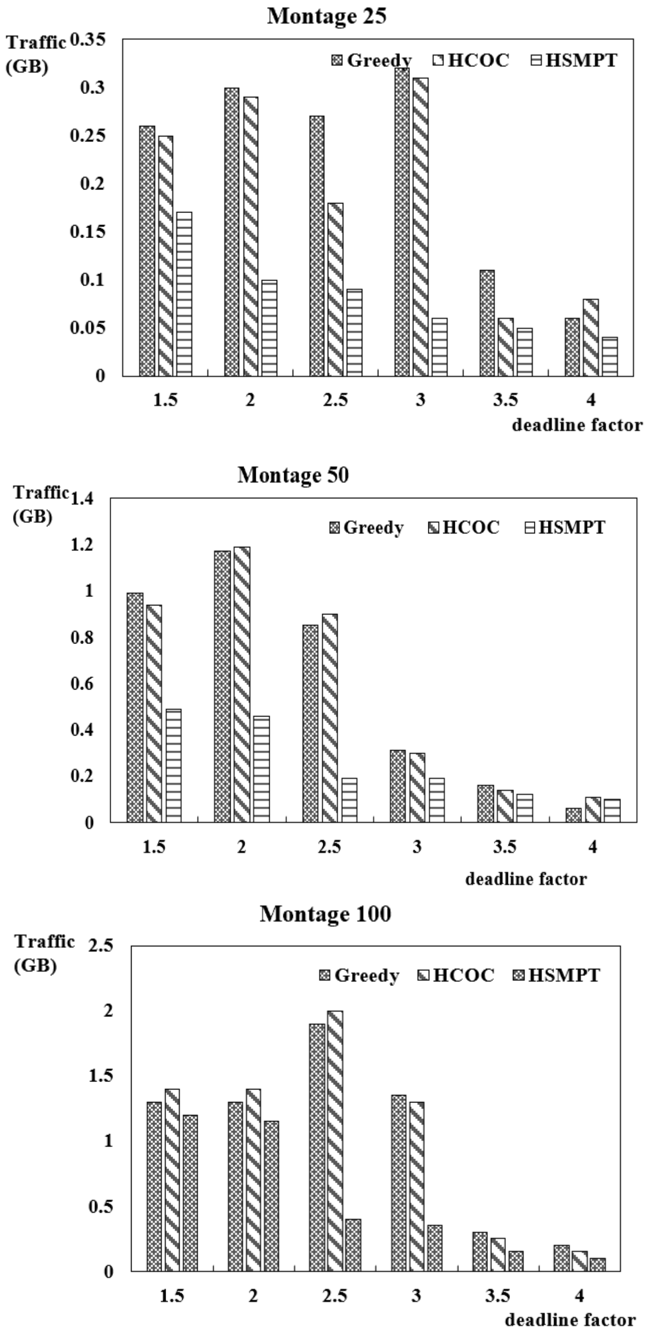


Fig. 4 traffic of 25, 50 and 100 tasks montage under different deadline

HSMPT algorithm divides the workflow according to the resource requirements and available resources of the workflow to ensure the minimum traffic between tasks. First, try to execute on the virtual machine resources of the private cloud. When the time constraint cannot be met, move some sub workflows to the public cloud for execution. Greedy algorithm selects the virtual machine with the best performance, and HCOC algorithm choose the virtual machine with the best cost performance. It does not consider the traffic between clouds, which increases the traffic between clouds, and restricts the tasks with cloud communication, increasing the execution time of workflow.

5. Conclusion

In this paper, we presented a novel a multi-objective heterogeneous workflow scheduling based on multi-constraint partitioning algorithm in hybrid cloud. On the premise of meeting the workflow deadline and task safe execution, the workflow is segmented, and the resource allocation and task scheduling are carried out on the private cloud and public cloud resources respectively. Compared with the Greedy algorithm and HCOC algorithm, the traffic between clouds and the execution cost of public cloud are reduced.

For future work, we will consider the following aspects: 1) due to the uncertainty of workflow resource demand, the estimated running time of tasks becomes inaccurate, resulting in the lack of optimal resource allocation for some tasks. Constructing an efficient and dynamic workflow scheduling strategy in hybrid cloud environment is an important problem to be solved; 2) due to the underlying failure of infrastructure and the mutual interference between virtual machines, the task execution may fail in a virtual machine, which seriously affects the subsequent task execution of workflow. How to reselect resources after task execution fails has also become the main research content in the next step.

Acknowledgements

This research was supported by Research on Multi-Objective Scientific Workflow Scheduling Algorithm in Hybrid Cloud Environment project of science and technology research program of Chongqing Education Commission of China. (No KJQN202003104)

R E F E R E N C E S

[1]. *A Quarati, A Clematis, D D Agostino.* Delivering cloud services with QoS requirements: Business opportunities, architectural solutions and energy-saving aspects. Future Generation Computer Systems, 2015.

[2]. *Mainak Adhikari, Tarachand Amgoth, and Satish Narayana Srirama*. 2019. A Survey on Scheduling Strategies for Workflows in Cloud Environment and Emerging Trends. ACM Comput. Surv. 52, 4, Article 68, August 2019, 36 pages.

[3]. *L F Bittencourt, E R M Madeira.* HCOC: a cost optimization algorithm for workflow scheduling in hybrid clouds. Journal of Internet Services and Applications, 2011, 2(3): 207-227..

[4]. *M Masdari, M Zangakani* (2020) Efficient task and workflow scheduling in inter-cloud environments: challenges and opportunities. J Supercomput 76:499–535.

[5]. *H Muhammad, Hilman, A Maria.* Rodriguez, and Rajkumar Buyya. 2016. Multiple Workflows Scheduling in Multi-tenant Distributed Systems: A Taxonomy and Future Directions. 1, 1, Article 1 January 2016, 37 pages.

[6]. Chopra, Nitish, and Sarbjeet Singh. "Deadline and cost based workflow scheduling in hybrid cloud." 2013 international conference on advances in computing, communications and informatics (ICACCI). IEEE, 2013.

[7]. Malawski, Maciej, Kamil Figiela, and Jarek Nabrzyski. "Cost minimization for computational applications on hybrid cloud infrastructures." Future Generation Computer Systems 29.7 (2013): 1786-1794.

[8]. Al-Khanak, Ehab Nabiel, et al. "A heuristics-based cost model for scientific workflow scheduling in cloud." CMC Comput. Mater. Contin 67 (2021): 3265-3282.

[9]. *J. Zhou, T. Wang, P. Cong, P. Lu, T. Wei, M. Chen*: Cost and makespan-aware workflow scheduling in hybrid clouds. J. Syst. Archit. 100, 101631, 2019.

[10]. *P Han, C Du, J Chen, F Ling, X Du*, Cost and makespan scheduling of workflows in clouds using list multiobjective optimization technique. J Syst Archit 112:101837, 2021.

[11]. *V. Arabnejad, K. Bubendorfer and B.* Ng, Budget and Deadline Aware e-Science Workflow Scheduling in Clouds, in IEEE Transactions on Parallel and Distributed Systems, vol. 30, no. 1, pp. 29-44, 1 Jan. 2019.

[12]. *K. Kalyan Chakravarthi, L. Shyamala & V. Vaidehi,* Budget aware scheduling algorithm for workflow applications in IaaS clouds. Cluster Comput 23, 3405–3419, 2020.

[13]. *B Wang, C H Wang, W W Huang, Y Song, X Y Qin.* Security-aware task scheduling with deadline constraints on heterogeneous hybrid clouds. Journal of Parallel and Distributed Computing, Volume 153, July 2021, Pages 15-28.

[14]. *Oliver Sinnen*. Task scheduling for parsallel systems, volume 60. John Wiley & Sons, 2007.

[15]. *B Lin, W Guo, X Lin.* Online optimization scheduling for scientific workflows with deadline constraint on hybrid clouds. Concurrency and Computation: Practice and Experience, 2015.

[16]. *H J Wang & Sinnen, Oliver*. List-Scheduling vs. Cluster-Scheduling. IEEE Transactions on Parallel and Distributed Systems. PP. 1-1. 10.1109/TPDS.2018.2808959, 2018.

[17]. *M Adhikari, T Amgoth, S N Srirama*, Multi-objective scheduling strategy for scientific workflows in cloud environment: a Firefly-based approach. Appl Soft Comput 93:106411, 2020.

[18]. *G. Juve, A. Chervenak, E. Deelman, S. Bharathi, G. Mehta, K. Vahi,* Characterizing and profiling scientific workflows, Future Gener. Comput. Syst. 29 (3) (2013) 682–692, Special Section: Recent Developments in High Performance Computing and Security.