ISSN 2348 – 9928
Doi:01.0401/ijaict.2014.06.02 Published on 05 (5) 2016

DEADLINES BUDGET CONSTRAINT IN MULTIPLE CLOUD WORKFLOWS BY USING TASKS REPLICATION

Ms. B. Poornima
PG Scholar,
Computer Science and Engineering,
SVS College of Engineering,
Coimbatore, Tamilnadu, India

Prof. S. R. Mugunthan
Assistant Professor (SG) and Head,
Computer Science and Engineering,
SVS College of Engineering,
Coimbatore, Tamilnadu, India

Abstract— ***The cloud infrastructure supplies them to proper environment for developing and executing workflow application in deadline constrained. A workflow application deals with software base application which is to automate at least some degree of process. Existing research in scientific workflow deals with the idle time for provisioned resource and to replicating task among several clouds. It increases the four well known scientific workflow simulation experiment to dealing with deadline met and to reduce the total execution time corresponds to application as the budget available for replication increases. To increase the capacity of the EIRP algorithm to enabling replication among tasks into multiple clouds. We propose an algorithm for structure aware scheduling and ranking the tasks replication. Also deals with how scheduling processes performed into multiple clouds.***

Keywords — ***Cloud Computing, Scientific Workflow, Task Replication, Soft Deadline, Scheduling Jobs.***

I. INTRODUCTION

In general workflow application determines set of tasks that have to be processed in a proper order to generate a one specific goal. Among the several applications in scientific workflow used for various development paradigms. Direct acyclic graph described the workflow model of the scientific workflow. In a simple manner automated scientific workflow used to call in data programs and other required inputs and to produce new output that may include in the result. Because scientific workflow management consists of thousand or more number of task [1], this type of workflow used for very large scale infrastructure.

Among this infrastructure one main interest in this paper is public cloud [2].Cloud computing defines storing and accessing data programs over the internet of the computer. Hosting to the customer as pay as you use demand. It has capabilities of business applications are to be exposed in a sophisticated way that can be accessed via the network. The ability of cloud makes a suitable platform to host deadline constrained workflows. Here present a model for scheduling transaction with deadlines in a single processor memory in database system then evaluate through detailed simulation experiments. A soft deadline is set under the assumption that, when un met of computation, does not rent for the computational useless [3]. But the maximum value is achieved when deadline is met, but the investment is not lost if deadline is missed by some margins. The cloud cost for server can be charged in various ways. Even though the budget constraint of a workflow might be limited factor to its capability of being scaling across multiple resources on the cloud. The workflow scheduler needs an estimation of the run time to schedule the workflow to complete before its deadline met, for example, analyzing historical data. But the typical cloud does not exist in the regular performances of the execution and data transfer time in workflow environment.

In budget constraint maximization problem involved in steady state and makes suitable platform to the make span problem. This is caused by various technologies and factors involved in variation of up to 30% for execution time and 65% for data transfer time as reported by Jackson al.[4]. Deadline to be missed during the delay of completion time and were task in the part of their critical path of the workflow. If virtual machines fail the workflow is directed to delay. But cloud offers above 99.9%, hence performances degradation is a more serious concern than resource failure in such environment. Previous work in workflow management cloud tries to minimize the workflow execution time ignoring deadlines and budgets [5].

Existing research in scientific workflow deals with the idle time for provisioned resource and to replicate task among several clouds. It increases the four well known scientific workflow simulation experiment to dealing with deadline met and to reduce the total execution time corresponding to application as the budget available for replication increases [6]. To address the limitation of previous work, we propose an algorithm to using “idle time” effectively of provisioned resources. Another point is

ISSN 2348 – 9928
Doi:01.0401/ijaict.2014.06.02 Published on 05 (5) 2016

to explore a new criterion for ranking candidate tasks for replication and for workflow structure aware scheduling of replicas. Also determines how the replication approach can be used when the provisioning and scheduling processes to be performed in between multiple workflows.

II. SYSTEM MODELS AND ASSUMPTION

A scientific workflow application is modeled as a DAG (Direct Acyclic Graph) G=(T,ET), where T denoted the set of task that composed by the workflow and E_T denotes the set of dependencies between the task. Dependency are formed by edges $e_{i,j}$ where t_i,t_j€T that has to be established by t_j it depends on the data generated by t_i. Here the task t_i is a parent task and task t_j is a child task. Without parent tasks are called entry task and without child tasks are called exit task. We considered workflow contains only one entry task and one exit task. It is to be achieved by inserting a dummy task t_{entry} and t_{exit} that makes execution time equals to 0 [5].All the entry tasks are child of t_{entry} and all the exit task are the parent of t_{exit}.All parent and child task should be given in a function as parent (t_j) and child (t_j). And each workflow G has a softdealine associated with it dl(G).There are three major component of the workflow scheduler.

$$\text{est}(t_j) = \begin{cases} 0, & if\, t_j = t_{entry} \\ \max_{t_a \in parents(t_j)} \left(est(t_a) + R_{min}(t_a) + D(e_{a,j})\right), & otherwise \end{cases}$$

$$lft(t_j) = \begin{cases} dl(G), & if\, t_j = t_{exit} \\ \max_{t_a \in children(t_j)} \left(lft(t_s) - R_{min}(t_s) - D(e_{j,s})\right), & otherwise \end{cases}$$

A cloud provider has a responsibilities to provide set of n virtual machine (VM) denoted by VM=vm_1, vm2,…..vm_n. Each virtual machine provides a different resource and determines the cost per use. Let C=c_1,c_2,……c_n be the cost vector associated with the use of each VM. And it is charged time unit based and partial utilization of time period. Therefore if the time period is one hour, utilization of VM per 61 minutes occurs in the payment of two hours. Data transfer among different VM's indicate that, for each task t_j to be executed in a given VM and is deployed before the data transfer from parents of t_j start process and then decomposed after all the data transfer to its child are completed. In this paper problem addressed by execution of a workflow G in the cloud on or before dl (G).Also the workflows are subjected to a soft deadline. Here extra budget to be increased for proportional to the importance of the application to complete by its deadline [5]. For this problem to be solved in two ways they are provisioning and scheduling.

The provisioning problem deals with the optimal number problem and type of VMs that should be completed within its deadline. The scheduling problem consists of the determination of the placement and order of execution of the different task that have to be composed workflow in the VMs selection during provisioning stage. A scheduling process maps and manages the execution of the independent task on the distributed system.

The provisioning and scheduling problems are mostly interconnecting by different decision and number of machines might result in different scheduled tasks [7]. Now we assume that workflow application executing in a single cloud. Hence more predictable execution and data transfer times are supreme for meet application deadlines. It's kept the workflow in a single data center and eliminates one possible source of execution delay. It eliminates the cost occurred by the data transfer among data center and also ignore overheads arrived by the workflow management system.

III. THE ENHANCEDIC-PCP WITH REPLICATION ALGORITHM (EIRP)

The proposed algorithm for increasing the probability of completing the execution of the scientific application within the user defined deadline in the public cloud environment. And it offers high flexibility in a significant performances variation by using task replication. The proposed algorithm used three distinct ways to operate it:

- **Step 1.** Combined provisioned of Cloud resources and task scheduling (Section 3.1);
- **Step 2.** Data transfer-aware provisioning adjust (Section 3.2); and
- **Step 3.** Task replication (Section 3.3).

The provisioning and scheduling problems are mostly interconnecting by different decision and number of machines might result in different scheduled tasks [8]. In the second step data transfer-aware provisioning adjust deals with non-entry task schedule as the last task of the VM. Then the algorithm met the requiring communication time by setting the start time of the machine. And the third step defines the task replication algorithm to replicate task into multiple cloud to make high availability of the resources. EIRP uses idle time effectively by provisioned VMs and allocated to extra replication. It increases the performances of task also deals with fault tolerance. Main target of EIRP is space replication. It denotes task only replicating in different virtual machines. If same task is scheduled multiple times it increases the fault tolerances. Therefore, the scheduling service can monitor executions for an application and notify the user or broker upon completion. IC-PCP algorithm used to assign partial critical path to VMs [11].

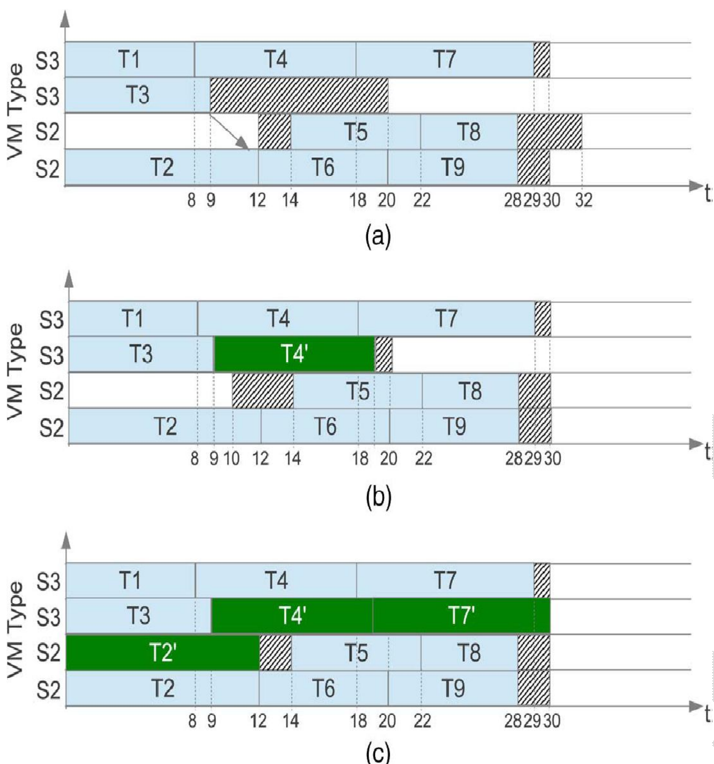

Fig 1: Schedule of workflow

(a) Original scheduling enabled by the IC-PCP algorithm.

(b) Utilization of an available idle slot for replication of T4 (no extra cost incurred).

(c) Allocation of VMs for two extra time units for replication of T2 and T7.

It determines the VMs starting from the cheapest to the more expensive path to find suitable platform to execute task. VM is selected when it meets the requirements. If existing machine is not meeting these requirements, a new provisioned. EIRP inserts new path beginning of the scheduling, when evaluating their suitable path to VMs. It schedules the path at the end of the schedule queue [10]. The following figure 1 shows the structure of the scheduling workflow of jobs in various tasks. With include cost effective.

In this diagram represents task with the data transfer time. And each task should be end in the 0.9 time. In order to provide efficient guidance in choosing the best strategy, we performed a joint analysis of several metrics according to methodology used in multiple workflows. They use an approach to multi-criteria analysis assuming equal importance of each metric. The goal is to find a robust and well performing strategy under all test cases, with the expectation that it will also perform well under other conditions, e.g., with different Grid configuration workload. They are well knows performance metrics commonly used to express the objectives of different stakeholders of Grid scheduling. The objective of the mean critical path waiting time and slowdown performances metrics is to evaluate the quality of the critical path.

3.1 Provisioning and Scheduling

The first step in EIRP algorithm consists of types of VMs to be used for workflow execution. In the execution phase user or broker submits application to execute in scheduling service at the master node [9]. Scheduling process first ensure that whether service satisfy the start time, end time and reservation phase that have been specified by the user or broker. If the reservation is still valid then scheduling service determines the reserved execution nodes for available before dispatching application to execution, otherwise to be waited in Once the execution service receives from the scheduling service each execution node starts executing an application and also updates the scheduling service.

3.2 Data-Transfer Aware Provisioning Adjust

We are not only determined start and end time of the scheduled task. Entry task is not considered as the first scheduled task in the VM it has to be moved to the execution before the task would run. Before start time of the first task VM to be provisioning. These may affects the start time of the task and provisioning of VM. In the DAG model all edges are connected to the corresponding task only task is enables between the two edges. And run time matrix could be calculated under the estimation of the edges and data transfer time defined by the graph. And task t_j input should depends on the t_i output that produced by the virtual machine. And all VMs should process at least one partial order resources that also taken into account to cost the data transfer between two tasks. Each VM should have the two problems they are caching problem and replication.

It also causes the workflow execution delay and execution of VMs for extra billing amount period. In the second step of the EIRP algorithm provisioning decision to be taken in the step 1.Each and every non-entry task scheduling as first task of a VM and non-empty task scheduled as the exit task of a VM. By setting the start time of the machine its meets the required time communication time. At last, the beginning of the first allocation slot of each virtual machine is anticipated by the estimated deployment a boot time for virtual environment, which is to be accounted for during the scheduling process of the prescribed in the previous section.

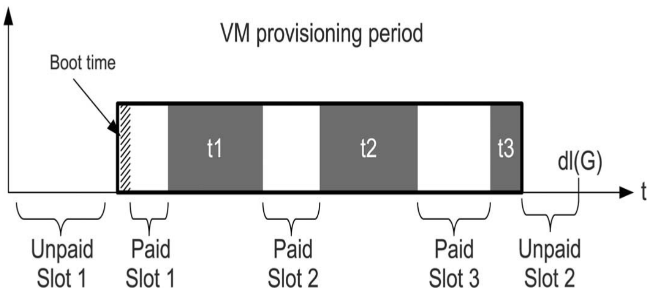

Fig 2 : Paid and unpaid idle time slots of provisioned VMs

3.3 Task Replication

Replicate task allows you to automate the copy of resources to another system over an encrypted connection. It allows us to create an off-site backup of the dataset of resources. The foreside corrections of the virtual machines to be ready to receive data and tasks in the moment that they are required to meet times estimated during the scheduling process [12]. But it does not account for delays in the tasks execution caused by poor performances of multiple cloud resources. The algorithm allows us to mitigate such effects with the utilization of task replication in the idle slots of provisioned resources. Extra replication will reduces the increasing performances so often listen the extra replication of the task in the virtual machine [10]. Here same task should be replicated to increase the fault tolerances of the VM.

By the two main to idle slot execute in the scheduling process. First defines the dependencies between the tasks, it may lead to period where that next task scheduled to a virtual machine has to wait for data being generated during execution of another task in another machine. EIRP algorithm is used to cause of idleness of applied provisioning time of VMs. Task replication process uses as a semi-active replication technique for fault tolerances [8]. Our approach is to maintain by a single entity in the case of replication. The first algorithm generating new VMs based on the available replication budget constraint. Then it creates the available list of possible slots.

It includes both paid slots, which are the slots is actually provisioned in the virtual machine and no task should be scheduled on it. Unpaid slot is between the where start and the deadline are the VM is not provisioned. FIG2 depicts the differences between these two types of slots. The slots are sorted based on its size and identified. The first slot is inserted first then insert unpaid slot. Next step consider defining the order for tentative replication of task available in the time slot. Here using three criteria to sort their “Replication Precedence Order” following below.

- **Ratio between execution time and lag time**
  The lag time lt (t,v) of a task t schedule to a VM v depends on the task latest finish time, start time and run time.

- **Execution time**
  task t_i and t_j respectively scheduled in VMs u and v and with the same ratio, t_i has precedence over t_j if $r_{ui}>r_{jv}$.

- **Number of children**
  For task t_i and t_j with the same ratio and execution time, t_i has precedence over t_j.

The algorithm prioritizes tasks whose proportion between time and available time is bigger than the larger task. Then finally task that have many children. If the priority of the time slot and priority of tasks is defined, then their algorithm for each slot iterates the task list. The algorithm checks the available budget for unpaid slots process, but before considering the utilization of the slot.

Provisioning information contains in the virtual machine was updated to reflect the choice for allocation of a new time slot. It’s enabling the extra replication where the VMs provisioned end of the process. Then remove the unused machines from the provisioned resources. It respects to the deadline constraint and replication budget set for execution of the workflow management.

IV. WORKFLOW EXECUTION PROFILE

In general workflow system consists of basic structure of process, pipeline, data distribution, data arrogation and data redistribution. All components are combined together finally to produce control flow structure [13]. Normally scientific workflow executes on the grid and some large applications. By using such components analyze the individual’s jobs and their relationship.

Workflow is generating by using workflow generator. Actual execution time and data size of the section to be obtained by using the workflow system.

Corresponding Author: Ms. B. Poornima, SVS College of Engineering, Coimbatore, Tamilnadu, India.

4.1 Montage

To generate custom mosaics of the sky using input images in the Flexible Image Transport System (FITS) format. The output of the geometry is calculated by using the geometry of the input images during the production of the final mosaic. Again the inputs are then re-projected to be of the same spatial scale and rotation. Montage [14] was created by the NASA/IPC Infrared Science archive as an open source toolkit .By using EIRP algorithm Montage is very large. In the all images background emissions are corrected in the same level. Finally the re-projected and corrected images are form in the final mosaic. It is also represented as the workflow and it can be executed in the Grid environment such as the TeraGrid [15]. The structure supports to adapt the large number of inputs processed by the workflow by over time, which to increase in the number of computational jobs.

4.2 Cyber shake

Cyber shake workflow is mainly used in the Southern California Earthquake Center (SCEC [16]) to characterize the earthquake hazards in a region using the Probabilistic Seismic Hazard Analysis (PSHA) technique. Strain Green Tensors is generated by using MPI finite simulation given a region of interest. The result of the Cyber Shake workflow is total of more than 800,000 jobs. It produces two different environments to classify the workflow management: Data movement and fault management.

4.3 LIGO

LIGO is used to detect the gravitational waves to produce by the various events in the world. In general the workflows have thousands of jobs to be running in the application. The all waves of a same family run by using the LIGO detectors. It has been split into multiple blocks. And it's matched by the different filter to find a matching inspiral [16]. By using the data aggregation jobs, the data can be obtained from the multiple jobs. Mainly used to comparison purpose to lists the runtime for jobs from an actual execution time and provide then total size of input generated and output detected by the each job running on the resource.

4.4 SIPHT

SIPHT workflow having a identical structure and have a large workflow that can be generated by composed independent smaller workflow. The result of the concatenated jobs only depends on the defined job added in the transition. There are various workflow jobs are available to comparing different combination of sequences. The initial job not depends on the any other job in the workflow. The job output depends on the previous job output defined by the previous job itself. Therefore the result of the jobs depends on the data regarded in the redistribution of job. The multiple features such are there to define the candidate jobs. All the jobs in the workflow, the candidate jobs that compare sequence between several replicas pairs are most computational intensive to the workflow. And this application is used to conducting a worldwide search for untrusted RNAs that regulate several processes such as secretion or virulence in bacteria. It's involved in the variety of individual programs that should be executed in the proper workflow application. Batch mode scheduling algorithms are initially designed for scheduling parallel independent tasks, such as bag of tasks and parameter tasks, on a pool of resources. Since the number of resources is much less than number of the task need to schedule on the resource.

V. RELATED WORKS

In extensive research of parallel and distributed workflow is subjected to workflow also referred as the DAG[17], it's presenting by the Shi and Dongarra defines the algorithm for describing and compare the static workflow scheduling algorithm[18]. Yu et al [19] and Hirales et al [20] presented the algorithm for scheduling workflow in Grid Computing environment. This model only adopted by the Cloud provider because this model consumes the financial cost of free model on an economic model. It cannot used by the cloud computing and amount is charged by the Cloud provider.

There is an algorithm for Grid computing to utilize dynamic provisioning of resource. So the algorithm is developed by Lin and Lu [21] for scheduling of workflow in service oriented application. Whatever it missed to include cost of utilization of resources required for its Cloud environment. The existing algorithm for Clouds to complement the desktop environment is developed by Reynolds et al [22] for Grid resources. However, Cloud resources are used to deploy the main goal in workflow. It increases the chance of task available to the all application in the workflow scheduling. The proposed algorithm not optimized the soft deadline constrained so the Xu et al[23] and Mao Humphery [23] produced algorithm to increase the best-effort of the algorithm to where the task to be detected. The algorithm used for cost optimization and deadline constrained execution of workflow. It consists of only one type of cloud model, to choose priority and take provisioning and scheduling decision. The meaning of replication of task has been intensively explained by the Grid systems without addressing the issue of cost for resources utilization. Prodan [23] proposed algorithm to determine the scheduling workflow application on the Grid computing with task replication to meet the deadline constraints.

VI. CONCLUSIONS AND FUTURE WORK

Scientific workflow management requires the effective allocation of tasks to limited resources over time and reduces the cost effect. But the multiple workflows add the new complexity to the problem. We have proposed EIRP algorithm for provisioning and scheduling task.

It takes into behavior of the cloud resources during the scheduling process to increase the capability of the replication of task. The most of the workflow application used to schedule the jobs depends on the scheduler in the Virtual Machine. But consider the budget constraints it decrease the soft deadline effects which imposed by the workflow in the multiple cloud.

Future work consists of ranking candidate tasks in the multiple clouds to increase the performance of the individual task. And decrease the replication of task in multiple workflow application. Finding the different rates arrived at workflows during the scheduling process.

References

[1] G. Juve, A. Chervenak, E. Deelman, S. Bharathi, G. Mehta, and K. Vahi, ‘‘Characterizing and Profiling Scientific Workflows,’’ Future Gener. Comput. Syst., vol. 29, no. 3, pp. 682-692, Mar. 2013.

[2] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, and I. Brandic,‘‘Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility,’’ Future Gener. Comput. Syst., vol. 25, no. 6, pp. 599-616, June 2009.

[3] R. Abbott and H. Garcia-Molina, ‘‘Scheduling Real-Time Transactions,’’ ACM SIGMOD Rec., vol. 17, no. 1, pp. 71-81, Mar. 1988.

[4] K.R. Jackson, L. Ramakrishnan,K.Muriki, S. Canon, S. Cholia, J. Shalf, H.J.Wasserman, and N.J.Wright, ‘‘Performance Analysis of High Performance Computing Applications on the Amazon Web Services Cloud,’’ in Proc. 2nd Int’l Conf. CloudCom, 2010, pp. 159-168.

[5] Y.-K. Kwok and I. Ahmad, ‘‘Static Scheduling Algorithms for Allocating Directed Task Graphs to Multiprocessors,’’ ACM Comput. Surveys, vol. 31, no. 4, pp. 406-471, Dec. 1999.

[6] Z. Shi and J.J. Dongarra, ‘‘Scheduling Workflow Applications on Processors with Different Capabilities,’’ Future Gener. Comput. Syst., vol. 22, no. 6, pp. 665-675, May 2006.

[7] E.-K. Byun, Y.-S. Kee, J.-S. Kim, S. Maeng, Cost optimized provisioning of elastic resources for application workflows, Future Gener. Comput. Syst. 27 (8) (2011) 1011–1026. [Online]. Available: http://www.sciencedirect.com/ science/article/pii/S0167739X11000744.

[8] J. Yu, R. Buyya, C.K. Tham, Cost-based scheduling of scientific workflow applications on utility Grids, in: First Int’l Conference on e-Science and Grid Computing, July 2005, pp. 140–147

[9] I. Brandic, S. Benkner, G. Engelbrecht, R. Schmidt, QoS support for timecritical Grid workflow applications, in: Int’l Conference on e-Science and Grid Computing, IEEE Computer Society, Los Alamitos, USA, 2005, pp. 108–115.

[10] C. Lin and S. Lu, ‘‘SCPOR: An Elastic Workflow Scheduling Algorithm for Services Computing,’’ in Proc. Int’l Conf. SOCA, 2011, pp. 1-8.

[11] W.-N. Chen, J. Zhang, An ant colony optimization approach to Grid workflow scheduling problem with various QoS requirements, IEEE Trans. Syst. Man Cybern. 39 (1) (2009) 29–43.

[12] D.M. Quan, D.F. Hsu, Mapping heavy communication Grid-based workflows onto Grid resources within an SLA context using metaheuristics, Int. J. High Perform. Comput. Appl. 22 (3) (2008) 330–346.

[13] “The workflow patterns initiative.” [Online]. Available: http://www.workflowpatterns.com

[14] “Montage: An astronomical image engine.” [Online].Available: http://montage.ipac.caltech.edu

[15] G. B. Berriman, E. Deelman, J. Good, J. Jacob, D. S. Katz, C. Kesselman, A. Laity, T. A. Prince, G. Singh, and M. Su, “Montage: A grid enabled engine for delivering custom science-grade mosaics on demand,” in SPIE, 2004.

[16] D. A. Brown, P. R. Brady, A. Dietz, J. Cao, B. Johnson, and J. McNabb, “A case study on the use of workflow technologies for scientific analysis: Gravitational wave data analysis,” in Workflows for eScience. Springer-Verlag, 2006.

[17] Y.-K. Kwok and I. Ahmad, ‘‘Static Scheduling Algorithms for Allocating Directed Task Graphs to Multiprocessors,’’ ACM Comput. Surveys, vol. 31, no. 4, pp. 406-471, Dec. 1999.

[18] Z. Shi and J.J. Dongarra, ‘‘Scheduling Workflow Applications on Processors with Different Capabilities,’’ Future Gener. Comput. Syst., vol. 22, no. 6, pp. 665-675, May 2006.

[19]J. Yu, R. Buyya, and K. Ramamohanarao, ‘‘Workflow Scheduling Algorithms for Grid Computing,’’ in Metaheuristics for Scheduling in Distributed Computing Environments, F. Xhafa and A.Abraham, Eds. New York, NY, USA: Springer-Verlag, 2008.

[20] A.Hirales-Carbajal, A. Tchernykh, R. Yahyapour, J.L. Gonza´lez-Garcı´a, T. Ro¨ blitz, and J.M. Ramı´rez-Alcaraz, ‘‘Multiple Workflow Scheduling Strategies with User Run Time Estimates on a Grid,’’ J. Grid Comput., vol. 10, no. 2, pp. 325-346, June 2012

[21] C. Lin and S. Lu, ‘‘SCPOR: An Elastic Workflow Scheduling Algorithm for Services Computing,’’ in Proc. Intl Conf. SOCA, 2011, pp. 1-8.

[22] C.J.Reynolds, S.Winter,G.Z.Terstyanszky, T. Kiss,P.Greenwell, S. Acs, and P. Kacsuk, ‘‘Scientific Workflow Makespan Reduction through Cloud Augmented Desktop Grids,’’ in Proc. 3rd Int’l Conf. CloudCom, 2011, pp. 18-23.

[23] M. Xu, L. Cui, H. Wang, and Y. Bi, ‘‘AMultiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing,’’ in Proc. Int’l Symp. ISPA, 2009, pp. 629-634.