Data-Locality Aware Scientific Workflow Scheduling Method in HPC Cloud Environments

Jieun Choi, Yoonhee Kim*
Dept. of Computer Science, Sookmyung Women's University
Seoul, Republic of Korea
{jechoi1205, yulan}@sookmyung.ac.kr

ABSTRACT

Efficient data-aware methods in job scheduling are necessary for successful execution of data-intensive applications. In this paper, we propose a Data-Locality Aware Workflow Scheduling(D-LAWS) technique for data-intensive scientific workflows on HPC cloud environments. We implement and validate the methods based on fairness in cloud environments. Experimental results show that, the proposed methods can improve performance and data-locality of data-intensive workflows in cloud environments.

Keywords

Data-aware Scheduling, Data-intensive application, Data-locality, Cloud, Scientific Workflow

1. INTRODUCTION

Cloud computing has rapidly become a widely adopted paradigm for scientific experiments. In highly distributed computing environments, traditional scientific applications, which are predominantly compute-intensive, require HPC resources to facilitate optimum execution of tasks. However, increasing data volumes generated during task executions of recent scientific applications reveal that these scheduling techniques do not adequately apply data-aware methods for optimal performance. Existing methods [3], [5], [6] do not consider data locality for optimal execution of data-intensive scientific workflows. However, data locality can drastically decrease network usage and application execution time and hence improve performance of data-intensive scientific workflows.

We propose a Data-Locality Aware Workflow Scheduling (D-LAWS) technique, which reduces data transfer time by applying data locality techniques based on network bandwidths, VM consolidation and user-specified SLA-sensitivity to data-intensive scientific workflow task scheduling. Our methods consolidate VMs and consider task parallelism by data flow during the planning of task executions of a data-intensive scientific workflow. We additionally consider more complex workflow models, GALFA-HI[7], and data locality during the placement and transfer of data prior to task executions. We showed that our method can improve performance and data-locality of data-intensive scientific workflows in OpenStack [1] cloud environments.

The rest of this paper is organized as follows: Section 2 describes our D-LAWS technique. In Section 3, we explain our experiment environments and conclude in Section 4.

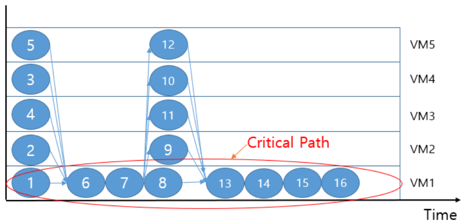
*Corresponding Author

DOI: 10.475/123_4

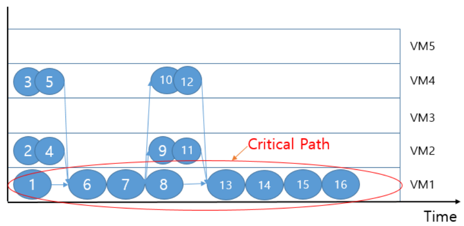
2. D-LAWS TECHNIQUE

D-LAWS technique includes a data-aware workflow scheduling and a resource consolidation method. The data-aware scheduling aims at minimizing the time taken by tasks to read inputs during executions and it considers data size, network bandwidth, and resource utilization. D-LAWS starts when a workflow comes into the system with Service Level Agreements (SLAs) which include a desired deadline for the workflow and minimum resource capacity of a VM such as core, memory, and disk capacity by a user. First, Estimated Finish Time(EFT), Execution Time(ET) and Estimated Start Time(EST) of all tasks in the workflow are initialized to zero and then the scheduler schedules each task into VMs. Next, all tasks on the critical path are scheduled on the same resource, capable of executing all tasks in the critical path. The scheduler then schedules tasks that are not in the critical path with consideration to EFT of previously related tasks. If the size of a precedence task of a task, equals zero then the task is an entry node and the scheduler finds a VM. However, in case that task is not in the critical path and has parent tasks, the task's EST is set to the sum of the longest finish time of the parent tasks and the data transfer time(DTT) which can be calculated using the data size of the task and network bandwidth for data transfer between VMs. After that, the scheduler finds VMs which already have input data of the task. Finally, the task is scheduled to the most suitable VM and EFT is calculated.

The VM consolidation method in [2]applies a parallelism reduction method of deadline assignment algorithm to a simple workflow. We consider task parallelism by data flow during the planning of task executions of a data-intensive scientific workflow. We additionally consider more complex workflow models and data locality regarding the placement and transfer of data prior to task executions. If all tasks can be executed on a single VM when they are combined, re-assign all tasks to that VM. This provides a parallelism reduction method with efficient VM utilization. Our method then proceeds to reschedule tasks into idle VMs to improve resource utilization.



(a) Before VM consolidation



(b) After VM consolidation

Figure 1: Scheduling scenario of D-LAWS

Figure 1 describes the scheduling result of our data-aware workflow scheduling method. There are five VMs in three different nodes. We used two instance types, medium and large, of VM with 4GB RAM each and having 2cores and 4cores correspondingly. Among the VMs, three VMs(VM1, VM3, VM5) are of medium type and two VMs(VM2, VM4) are of large instance type. VM1 is in node 1, VM2 and VM3 are in node 2 and, VM4 and VM5 are in node 3. The arrows represent data flows. For the case of Figure 1a, there is one critical path with tasks (1, 6, 7, 8, 13, 14, 15, 16) scheduled to VM1. In Figure 1a, the tasks which have same parent and children tasks are set A(tasks 1∼5) and set B(tasks 8∼12). According to the VM consolidation method, the scheduler can perform consolidation on a set of tasks without affecting other sets. As a result, we see the scheduling of tasks from consolidating VMs as shown in Figure 1b.

3. EXPERIMENT RESULT

The GALFA workflow in our experiments reads 27.5GB of input data and 8.4GB of intermediary data and writes 11GB of output data for data cubes shrunk by averaging every 5 planes. We compare the proposed technique which is represented as D-LAWS in the graph with CP, which schedules a workflow considering only the critical path [3], and FCFS which schedules a workflow regarding task priority and queue order.

Figure 2 shows the execution time and efficiency of CP, FCFS, and D-LAWS. In the paper, execution time is the normalized ratio of time relative to FCFS with 100% as the worst value. Results show execution time for CP, FCFS, and D-LAWS as 74%, 100%, 62% respectively. The D-LAWS technique speeds-up execution times of a workflow. The improvement in speed-up times of D-LAWS is a result of considerations for data locality and data transfer times. Efficiency refers to the ratio of time that the system is executing tasks [4] with 100% as an ideal value from the system view. We also see efficiency values as 68%, 76%, 81% corresponding to CP, FCFS, and D-LAWS respectively. When using the D-LAWS technique the scheduler could reduce the data transfer time considering data-locality based on network bandwidth and data size.

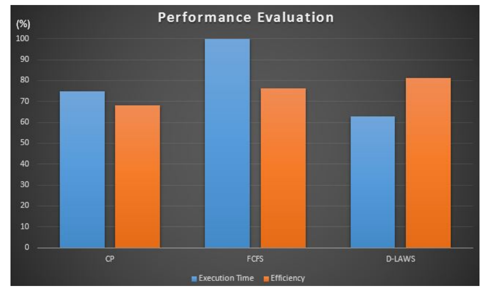


Figure 2: Normalized execution time and efficiency of CP, FCFS, and D-LAWS

4. CONCLUSIONS

In this paper, we propose D-LAWS technique for data-intensive scientific workflows on HPC cloud environments. D-LAWS applies data-locality and data transfer time based on network bandwidth to scientific workflow task scheduling and balances resource utilization and parallelism of tasks. It consolidates VMs and considers task parallelism by data flow during the planning of task executions of a data-intensive scientific workflow. In the future, we will consider hybrid cloud environments and experiment for data-intensive scientific applications with variant characteristics.

5. ACKNOWLEGMENTS

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (2015M3C4A70 65646)

6. REFERENCES

[1] OpenStack, http://www.OpenStack.org
[2] Mao et al., Auto-scaling to minimize cost and meet application deadlines in cloud workflows, Proceedings of 2011 Int. Conf. High Perform. Comput., Netw., Storage Anal., 2011. pp. 1-12.
[3] Bittencourt et al., HCOC: A Cost Optimization Algorithm for Workflow Scheduling in Hybrid Clouds, Journal of Internet Services and Applications. 2011.
[4] Wang et al., Load-balanced and locality-aware scheduling for data-intensive workloads at extreme scales, Journal of CCPE 2015.
[5] Ahn et al., Auto-scaling of virtual resources for scientific workflows on hybrid clouds, ScienceCloud '14. pp.47-52, Vancouver, Canada, 2014.
[6] Choi et al., VM Auto-Scaling Methods for High Throughput Computing on Hybrid Infrastructure, Journal of Cluster Computing, 2015.
[7] Peek et al., The GALFA-HI Survey: Data Release 1, The Astrophysical Journal Supplement, Vol. 194, Issue 2, article id. 20, 2011.