Network-Aware Task Scheduling of MapReduce Framework over Multi-Clouds and High Speed Networks

Praveenkumar Kondikoppa , Chui-Hui Chiu, Cheng Cui and Lin Xue and Seung-Jong Park
Department of Computer Science, Center for Computation & Technology,
Louisiana State University, LA, USA
{pkondi1,cchiu1,ccui,lxue2,sjpark}@lsu.edu

ABSTRACT
Hadoop [1] is an open source implementation of MapReduce [3] programming framework to process large data set. Originally it was developed to operate over single cloud. At multiple clouds over 10Gbps high speed networks, the computing resources are provisioned from any of physically distributed cluster.The placement of tasks is critical for performance of Hadoop over networked clouds. In this work we add network awareness in Hadoop while scheduling the tasks. We observe 12 % to 15 % reduction in execution time in FIFO and FAIR schedulers of Hadoop for varying workloads.

Keywords
Multi-Cloud, Hadoop Scheduling

1. INTRODUCTION
Cloud computing coupled with Hadoop/MapReduce programming model have made huge strides in big data analysis. Hadoop is extensively used on compute clouds at Amazon, Facebook and Yahoo. Recently Yahoo Grid team took a moderate sized Hadoop cluster of 910 nodes and won the TeraSort benchmark. Amazon's Elastic Compute cloud enables running Hadoop as a separate service. This has led many of the organizations to set up Hadoop based private cluster for internal data processing.

Geographically separated clouds expose new challenges to Hadoop based applications. Apart from increased computing power, networked clouds would be helpful for sharing the data between the clouds. At times data set resides on one cloud and map task is started at another cloud and data needs to be transferred, or when data is generated at one place and processed at another place, a typical case of astrophysics and large scale scientific discovery applications. Due to high degree of inter communication between the computing units running Hadoop tasks, available bandwidth plays critical role for efficient execution of a job. If networked clouds are connected with high speed network infrastructure providing assured bandwidth then deployment of Hadoop over distributed cloud is feasible.

2. NETWORK AWARE SCHEDULING
The goal of Task scheduling in hadoop is to take computation wherever the data is located. If it can't meet this objective then it schedules the task on whichever the compute node is free, this causes data to be transferred to compute node for processing. When resources are provisioned by distributed network, moving the data across the network causes the degradation in Hadoop's performance. In Network scheduling we provide administrator controlled script which has the physical locations of the computing resources. When head of the queue task doesn't find a compute node with data then scheduling of the task is delayed for specified delay. If any of the compute nodes becomes free having data to process head of the queue map task then corresponding map task is launched on that compute node. Duration for which a head of the queue map task is to be delayed is based on the average length of the map tasks for a Job hence requires careful tuning.

3. EXPERIMENTAL SETUP
CRON [2] is an Emulab based testbed; a cyberinfrastructure of reconfigurable optical networking environment that provides multiple emulation testbeds operating up to 10Gbps bandwidth. It consists of two main components: (i) hardware (H/W) components, including a switch, optical fibers, net- work emulators, and the workstations required to physically compose optical paths or function at the ends of these paths; and (ii) software (S/W) components, creating an automatic configuration server that will integrate all the H/W components to create virtual network environments based on the users requirements. All components are connected

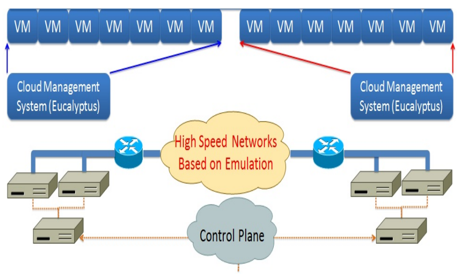

Figure 1: Experimental Setup

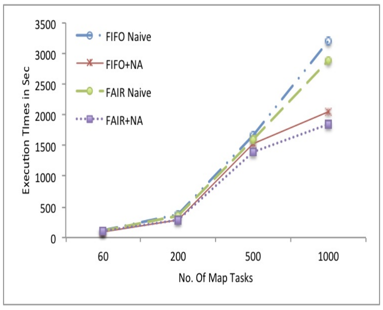

Figure 2: Execution times for Different schedulers

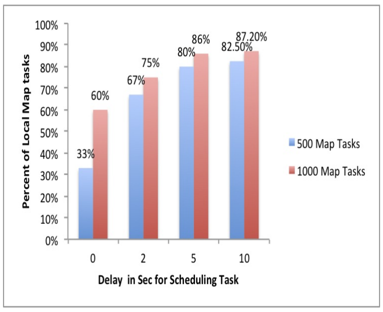

Figure 3: Increase in local map tasks with Network-aware

with 100/1000 Mbps Ethernet links for control. Each workstation is connected with 10 Gbps optical fibers for data movement.

We use Eucalyptus as cloud computing software,We setup two NEuca-patched [4] Eucalyptus Clouds to construct the multi-Cloud scenario.The NEuca patch attaches additional exclusive virtual NICs to a VM for application data transmission. We have two computing nodes in each Cloud and each computing node accommodates three VMs. Then, all VMs connect to each other through the additional NICs to form a virtual cluster. We deploy our enhanced Hadoop on the virtual cluster to examine its performance in the complicated network environment Figure 1 shows the scenario used for conducting the experiments.

We avoid to multiplex VMs on a physical machine for the better and stabler execution. Each VM is allocated with 2 physical CPU cores, 2 GB of physical memory, and 10 GB of local hard disk space. Regarding the NIC, the VirtIO interface is adopted to serve the virtual NIC so that the transmission rate is as high as the Hypervisor can provide.

4. RESULTS

We compare the execution times of Hadoop native scheduling with Network Aware Hadoop for varying delay values. Figure 2 shows the execution times for varying number map tasks, FIFO Network aware scheduler identified as FIFO+NA shows the reduction in execution time of about 12 % on average and maximum reduction of 15 % when number of map tasks is higher compared to Hadoop Naive scheduler identified as FIFO Naive. FAIR scheduler with Network Aware identified as FAIR+NA also shows the similar re-

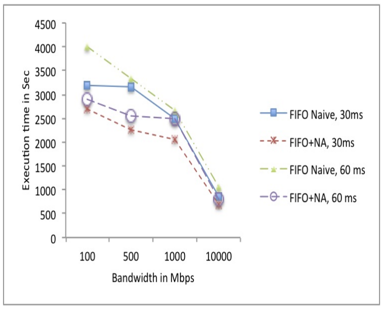

Figure 4: Execution times for varying inter-cloud bandwidth

sults, FAIR scheduler in general takes less execution time compared to FIFO, The reason being effective utilization of the cluster. We performed experiments under varying bandwidth and delay between the clouds. In all the cases the significant reduction in execution time is observed between Native Hadoop scheduler and Network-aware scheduler as number of Map task becomes more. Figure (Need to add the figure) shows the execution times for varying inter-cloud bandwidth. It is evident that for higher inter cloud bandwidth the performance of Hadoop with Network is greatly increased compared to native Hadoop scheduling. Figure 3 shows percentage increase in the local map tasks. This results shows that, Network awareness coupled with delay scheduling could be used to minimize the transfer of the data between the clouds, However as the delay is increased which might add to the overall execution time. Hence selecting the optimum delay value is critical.

Figure 4 shows the results of varying inter-cloud bandwidth and delay for processing of 10 GB of data size. For higher bandwidth provisioned between the clouds the performance of Hadoop FIFO scheduler with Network awareness is greatly improved. Similar results are observed for FAIR scheduler.

5. CONCLUSIONS AND FUTURE WORK

Placement of the task on a node which has the data is critical for the performance of Hadoop over distributed cloud. single-cluster MapReduce-duce architecture may not suitable for situations when data and compute resources are widely distributed In this work we provide network awareness to the FIFO and Fair schedulers in Hadoop. We evaluate our implementation on resources provided by CRON testbed. performance improvement of 12 % to 15 % is observed in both FIFO and Fair schedulers. We plan to extend the Network awareness while placing the reduce task since Reduce phase in MapReduce adds significantly to the overall execution time.

6. REFERENCES

[1] Apache Hadoop. http://hadoop.apache.org/.

[2] CRON Project: Cyberinfrastructure for Reconfigurable Optical Networking Environment, 2011. http://www.cron.loni.org/.

[3] ACM. *MapReduce: Simplified Data Processing on Large Clusters*, volume 6, 2004.

[4] E. S. Inc. Installing eucalyptus 2.0. http://open.eucalyptus.com/wiki/EucalyptusInstallation_v2.0.