IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

A Survey on Dynamic Slot Configuration Techniques for Hadoop Clusters

Rashmi S[1] , D Khasim Vali[2]

PG Scholar, Department of CSE, VidyaVardhaka College of Engineering, Mysuru, India[1]

Associate Professor, Department of CSE, VidyaVardhaka College of Engineering, Mysuru, India[2]

Abstract: Hadoop is an open-source framework used for processing and handling big data in a distributed environment. MapReduce is a programming model using which we can create applications to process massive amount of data laterally, on large clusters of commodity hardware in a reliable manner. One of the major concerns in Hadoop is to minimize the makespan of a collection of MapReduce jobs. The static slot configuration technique in Hadoop may lead to long makespan and low resource utilization. To improve the performance of system dynamic slot configuration and appropriate scheduling policy must be adapted. This paper is a survey on different dynamic slot configuration techniques.

Keywords: Hadoop, MapReduce, minimum makespan, dynamic slot.

I. INTRODUCTION

In recent years Hadoop-MapReduce has become a powerful Computation Model for processing large data on distributed commodity hardware clusters. The Hadoop framework is deployed by many major companies like Google, Facebook, Yahoo, IBM. With the emerging cloud computing technology, MapReduce is not only for processing data in major companies. It can be easily adopted by financial institute, government sector, academic and industry by launching a MapReduce cluster on cloud.

The classic Hadoop is based on master/slave architecture. Each Hadoop cluster has one master node and one or more slave nodes. The master node consists of jobtracker, task tracker routines, namenode and datanode. The master node is responsible for allocating jobs and co ordination of tasks. Slave node works as both data node and namenode and is responsible for execution of MapReduce jobs.

The slot is used to indicate number of task each node can accommodate. Each node has preset number of slots assigned to it. And slot can be either map slot or reduce slot for map and reduce task respectively. By default the Hadoop uses static slot configuration which means fixed number of slots MapReduce jobs are used. This leads to long makespan and low resource utilization which may hinder the performance of the Hadoop cluster.

Researchers have put tremendous effort on job scheduling, resource management and slot configuration. The Hadoop consists of many system parameters. By modifying these system parameters the performance of the Hadoop can be improved.

One way to improve the performance of Hadoop is modifying the slot configuration which has crucial effect on system performance. The slot values in static configuration are just random numbers without considering the job characteristics. So a new mechanism is to assign the slots dynamically.

In this survey paper, we focus on different dynamic slot configuration techniques to leverage the performance of the system by reducing makespan of jobs.

II. RELATED WORK

One of the efficient ways to leverage the performance of MapReduce job is through dynamic slot configuration.

J. Polo et al. [3] introduced a novel job scheduling and resource management technique called Resource-aware Adaptive Scheduler (RAS). RAS can greatly improve resource utilization in turn the performance of a job is also increased. The RAS technique focuses on job slot and sloves the problem by utility-driven algorithm. In this algorithm the scheduler has the ability to adapt and respond to changing demand in resource and availability.
The RAS uses ‘job slot’ by extending abstraction of ‘task slot’. A ‘job slot’ is specific to each job and each as associated requirement for map and reduce task. This technique uses existing profiling information based on previous executions of tasks to make resource allocation and scheduling. And leverages resource profile to achieve better resource utilization and thereby increases the performance of system.

J. Wolf et al. [4] focused on flexible scheduling allocation technique called as FLEX with Hadoop fair scheduler. Its main goal is to optimize response time, SLA, makespan of MapReduce jobs. They proposed penalty function for estimation of job makespan, epoch scheduling to measure partitioning time, moldable scheduling for job parallelization and malleable scheduling for different interval parallelization.

J. Wang et al. [5] proposed a fair and efficient slot configuration technique called FRESH. It addresses the issues of static slot configuration in two steps. First decide slot allocation, i.e. how many slots to be considered as map and reduce slots from the given total number of slots.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 3, March 2016

Second, when a free slot is available assigning map/reduce task appropriately. FRESH are used for both static and dynamic slot configuration. In first static slot technique the number of map and reduce task fixed and defined before creating cluster.

In dynamic model the job profiles are considered as prior knowledge. To obtain best slot setting first fair scheduler is used to for assigning task for free slots. Next FRESH dynamically allocates map and reduce slots by allowing a slot to change its type online.

S. J. Tang et al. [6] proposed step-by-step allocation technique called DynamicMR to optimize the resource allocation in MapReduce. In this technique the slots are classified into two types, busy slots and idle slots. The slot allocation optimization is done at two levels namely macro-level and micro level. DynamicMR consists of three techniques Dynamic Host Slot Allocation (DHSA), Speculative Execution Performance Balancing (SEPB), Slot Pre Scheduling.

DHSA is used to maximize slot utilization by dynamically allocating map or reduce slots to map or reduce tasks while maintaining fairness. It can be used for any MapReduce jobs without any initial knowledge. User can choose either of the two DHSA namely PI-DHSA and PD-DHSA. SEPB technique identifies inefficiency of slot allocation caused by speculative job execution. It balances half measure between a single job and a batch of jobs. Slot Pre-Scheduling leverages performance and slot utilization proficiency by increasing data locality. It pre-schedules the task when there are map tasks which are pending but no idle map slots.

Z. Liu et al. [7] proposed a framework called DREAMS to reduce the negative effect of partitioning skew where the output of map task is asymmetrically distributed to reduce task. This data skew causes huge difference between the fastest and slowest task runtime. DREAMS is a framework for reducing runtime partitioning skew. This is done by adjusting runtime resource allocation for reach reduce slot to match corresponding size of reduce task.

In order to do this, first an online partition size prediction model is developed which is used to measure the partition size of reduce tasks at runtime. Second a reduce task performance model is established which associates runtime resource allocation with task completion time.

Y. Yao et al. [8] introduced TuMM a unique slot management technique which overcomes the issue of fixed slot by enabling to configure slots dynamically. The goal of TuMM is to enhance resource utilization and lower the makespan of multiple jobs by automating slot assignment ratio.

The two components are introduced to achieve this goal: Workload monitor (Wm) and Slot assigner (Sa). The workload monitor task is to first collect the execution time information of recently executed jobs periodically. Then it calculates the current map and reduce workloads in the cluster. The slot assigner decides and tunes the slot ratio between map and reduce task based on estimate of Wm.

For heterogeneous environment a different version of TuMM called H_TuMM is introduced by extending the TuMM technique. The H_TuMM sets the slot configuration for each individual node separately and in turn reducing the makespan of a batch of jobs. In H_TuMM each node will have different slot assignment ratio between map and reduce tasks.

III. CONCLUSION

The performance of Hadoop MapReduce can be optimized by fine- tuning system parameters. Slot configuration system parameter has great impact on performance. By dynamically configuring MapReduce slots improved resource utilization and short makespan can be achieved. From the survey, it can be concluded to prefer a dynamic slot allocation technique with optimal scheduling policy for better performance.

REFERENCES

[1] J. Dean and S. Ghemawat, “MapReduce: simplified data processing on large clusters”, in Communications of the ACM, vol. 51, 2008.

[2] Apache Hadoop. Reference link: http://Hadoop.apache.org.

[3] J. Polo, C. Castillo, D. Carrera et al., “Resource-aware adaptive scheduling for MapReduce clusters,” in Proceedings of the 12th ACM/IFIP/USENIX international conference on Middleware, 2011.

[4] J. Wolf, D. Rajan, K. Hildrum, R. Khandekar, V. Kumar, S. Parekh, K.-L. Wu, and A. Balmin, “Flex: A slot allocation scheduling optimizer for MapReduce workloads”, in Middleware 2010, ser. Lecture Notes in Computer Science, I. Gupta and C. Mascolo, Eds. Springer Berlin / Heidelberg, vol. 6452. pp. 1, 2010.

[5] Jiayin Wang, Yi Yao, Ying Mao, Bo Sheng, N. Mi, “FRESH: Fair and Efficient Slot Configuration and Scheduling for Hadoop Clusters”, IEEE 7th International Conference on Cloud Computing, DOI: 10.1109/CLOUD.2014.106. Anchorage, AK, pp 761, June 2014.

[6] S. Tang, B. Lee, and B. He,” Dynamic MR: A Dynamic Slot Allocation Optimization Framework for MapReduce Clusters”, IEEE Transactions on Cloud Computing, vol. 2, issue. 3, September 2014.

[7] Z. Li, Q. Zhang, M. F. Zhani, R. Boutaba, Y. Liu, Z. Gong et al., “DREAMS: Dynamic Resource Allocation for MapReduce with Data Skew”, Integrated Network Management (IM), DOI 10.1109/INM.2015.7140272, 2015 IFIP/IEEE International Symposium on May 2015.

[8] Y. Yao, J. Wang, B. Sheng, C. Tan and N. Mi, “Self-Adjusting Slot Configurations for Homogeneous and Heterogeneous Hadoop Clusters”, IEEE Transactions on Cloud Computing, Vol. PP, September 2015.

DOI 10.17148/IJARCCE.2016.53112