Vrije Universiteit Amsterdam

Universiteit van Amsterdam

Master Thesis

Self adjusted auto provision system at resource level

Author: You Hu 2631052

1st daily supervisor:	Prof. Adam Belloum	UvA, Netherlands eScience Center
2nd supervisor:	Dr. Jason Maassen	Netherlands eScience Center
2nd reader:	supervisor name	

A thesis submitted in fulfillment of the requirements for the joint UvA-VU Master of Science degree in Computer Science

July 14, 2020

"History, as Hegel said, moves upward in a spiral of negations"

Abstract

The resource management is no doubt one of the key problems that all cloud or cluster need to face. Here, the special scenario the LOFAR team is taking: they build computation infrastructure of their own to process the data collected by the tremendous telescopes. They are both cloud provider and cloud user. The jobs the cloud is undertaking are long and highly computation consuming. The current horizontal distributed solutions, MPI and Spark, have intrinsic drawback on resource utilizing. To increase the resource utilization level, an auto-provisioning distributed computing system is designed. The auto-scaling mechanism enables the applications to dynamically fetch and release resources, and in the same time the resources of the cluster are maximally used. The results shows:

Contents

List of Figures

List of Tables

LIST OF TABLES

1

Introduction

1.1 Context

The LOw Frequency Array(LOFAR) telescope[1] demands for huge data processing ability. It consists of 51 stations cross Europe and a typical LOFAR observation has the size of 100TB, after frequency averaging, the size can be reduced to 16TB. (1) Collectively, there are over 5 PB of data will be stored each year. (2) In the data processing pipeline, the image calibration is a vital step. As the data collecting speed exceeds the capability of processing, the data will be stored and archived first. It will be feed to processing pipeline when it is needed. The Netherlands eScience Center has developed solutions for calibrating imaged observation collected by LOFAR. The one of the way is sky map based direction independent calibration. To calibrate the observation by given sky map, SAGECaL is invented and implemented for this purpose.(3). By given pre-processed observation data, sky model and parameters, the calibration can be done independently. However, it is a computation consuming application. Currently, eScience Center has developed GPU, MPI and Spark versions for acceleration. All of them have achieved great acceleration compared to the naive uni-node version.

However, under the huge requirement of computation, all three solutions have limitation. The GPU version provide no doubly great acceleration, which is in a sense vertical scaling. MPI and Spark can be considered as horizontal scaling, GPU can provide support as well. But as it is said, LOAFR is cloud user and cloud provider in the same time, the resource utilization of their infrastructure is also important. The MIP and Spark solutions may lead to waste of resources. The Fig. 1.1 gives an example that when the required computation resources decrease, idle resources(compute nodes) wouldn't be released by Spark. On the

[1] http://www.lofar.org/

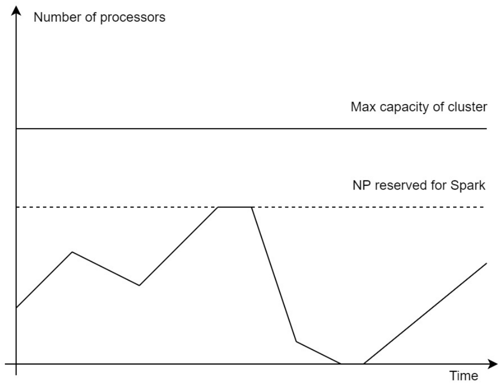

Figure 1.1: Resource utilization by Spark - Spark occupys fixed resources

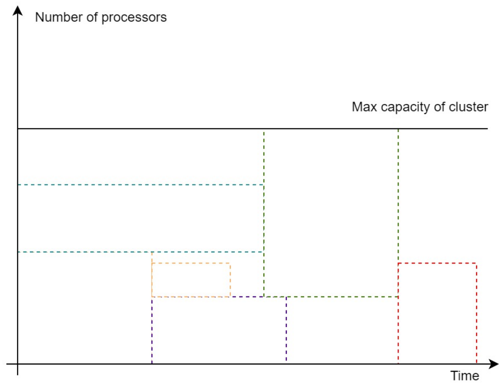

Figure 1.2: Resource utilization by MPI - Too large jobs make resource waste

other hand, a pure batch job system may enter a very special situation that a big job is waiting for resource while idle resources can not fulfill the requirement, for instance as shown in Fig. 1.2.

1.2 Objective

In the previous section, it is mentioned that LOAFR team has the role of both sides. Therefore, in this research, higher resource utilization of cloud system is the first objective. In the same time, the acceleration of calibration processing is the second goal to achieve. Hopefully, the higher resource utilization, or in other words more active computation resources, is able to shrink the time for computing.

1.3 Research Question

The research question overall is how to implement an elastic distributed algorithm that can be auto-scaled dynamically that reduces the waste of resources. More particularly, it can be extended into more specific questions:

- How to set up an auto provisioning system adopting to the workload of a cloud that requests and releases resources dynamically?
- How to transform the calibration application into distributed and parallel version?
- How to make the distributed application capable to the dynamic scaling environment?

1.4 Research Method

The research method is practice based implementation plus comparison. In this research, firstly the previous work and current solutions will be explored. The limit, pros and cons of previous work will be collected and analyzed. Secondly, a system that fulfills the demand of auto provisioning and overall higher resource utilization will be implemented. Last, the system will be tested by given benchmark, and compared to other systems on the objective goals.

2

Literature Review

In this chapter, the previous related works will be explored. It is divided into two main topics: how resource utilization in cloud/cluster is improved, and how to build an auto-scaling distributed application. The outline of the first topic follows the development along with the new technologies introduced. For the second topic, many kinds of existing solutions will be listed and the benefit and drawback of them will be discussed.

2.1 Resource utilization optimization on Cloud/Cluster environment

In this section, the definition of resource in the cloud field will be elaborated first, and together with the other important metrics that stakeholders concern. Then following the path of development in this topic, a few kinds of approaches, and their typical examples will be explored. Note that, these approaches became popular one by one, but it doesn't mean the later ones are the replacement of previous ones. The introduction of new technologies has led to the emergence of new methods and expanded the boundaries of the field.

2.1.1 Definition of resource and QoS

In the cloud environment, there are many kinds of resources and a set of aspects around the cloud economy. Both Jennings (4) and Manvi (5) starts with the definition of resources.

Jennings et al. categorize the resources into compute, networking, storage, and power. Manvi et al. summarizes that there are physical resources(CPU, memory, storage, network elements and sensors) and logical resources(OS, energy, Network throughput/bandwidth, Load balancing mechanisms and so on). There are overlap between two of them especially in the physical part, while Manvi adds API, OS, load balancing to logical resource concept.

However, it is understandable that API , OS and some protocols and mechanisms can be viewed as some sort of asset of cloud owner, but they are more acceptable to be considered as part of Quality of Service(QoS) which requires resources to fulfill. Therefore, in this paper, we mainly focus on the utilization of physical resources like CPU, memory, storage; and consider the trade-off between utilization rate and QoS.

QoS (Quality of Service) metrics are import to both cloud provider and consumer, they are good for optimizing resource utilization efficiency Bardsiri and Hashemi listed detailed metrics from four kinds of features:performance, economic, security and general. (6) Their coverage is comprehensive. There are plenty of features and corresponding metrics that cloud users would put concern.

Given the background of researching utilization under limited resources, there are few metrics we consider important. For performance features, the CPU load rate and packet loss frequency are what users may concern while the cloud provides needs to make a compromise for utilization of resources. In the economic aspect, the price per resource unit is the key point that cloud providers and users wrestle on. However, from the technical view, the time for VM booting/deleting/suspending/provison attract more attantion. Besides, the availability and reliability are very important as well. The response time is the key metric for auto-scaling mechanism. Cloud providers need to pay effort on fault tolerance to make sure the safety of the cloud.

In the following sections, we will explore how cloud providers face resource management issues and the metrics shown above play important roles in those researches.

2.1.2 Scheduling strategies on batch queuing systems

The batch scheduling has long history in the entire computer science field, from the very beginning of mainframe age and still part of the fundamental configuration of current researches and systems. The queuing system schedules jobs according to their priorities as the resource waste of FIFO is obvious. Therefore, A good deal of optimization are applied on the priorities related concepts. Preemption, backfill and heuristics are traditional routes for scheduling. Besides, with the growth of computation ability, the machine learning/deep learning approaches which are based on historical data become the front of researches in this field.

2.1.3 Virtual machine founded cloud era

2.1.4 Container and orchestra

2.2 Distributed systems and auto-scaled algorithms

3

Architecture and Methods

3.1 Overview design

3.2 Auto provision management

3.3 Elastic distributed application

4

Analysis and results

4.1 Resource utilization optimization on Cloud

4.2 Auto-scaled distributed algorithms

5

Discussions

6

Conclusion

6. CONCLUSION

References

[1] Hanno Spreeuw, Souley Madougou, Ronald Van Haren, Berend Weel, Adam Belloum, and Jason Maassen. **Unlocking the LOFAR LTA**. pages 467–470, 2019. 1

[2] **PROviding Computing solutions for ExaScale ChallengeS**. (777533):1–163, 2020. 1

[3] S. Kazemi, S. Yatawatta, S. Zaroubi, P. Lampropoulos, A. G. de Bruyn, L. V.E. Koopmans, and J. Noordam. **Radio interferometric calibration using the SAGE algorithm**. *Monthly Notices of the Royal Astronomical Society*, **414**(2):1656–1666, 2011. 1

[4] Brendan Jennings and Rolf Stadler. **Resource Management in Clouds: Survey and Research Challenges**. *Journal of Network and Systems Management*, **23**(3):567–619, 2015. 5

[5] Sunilkumar S. Manvi and Gopal Krishna Shyam. **Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey**. *Journal of Network and Computer Applications*, **41**(1):424–440, 2014. 5

[6] Amid Khatibi Bardsiri and Seyyed Mohsen Hashemi. **QoS Metrics for Cloud Computing Services Evaluation**. *International Journal of Intelligent Systems and Applications*, **6**(12):27–33, 2014. 6