Available online at www.sciencedirect.com
ScienceDirect
ELSEVIER
December 2016, 23(6): 8–15
www.sciencedirect.com/science/journal/10058885
The Journal of China Universities of Posts and Telecommunications
http://jcupt.bupt.edu.cn

Deadline based scheduling for data-intensive applications in clouds

Fu Xiong[1] (✉), Cang Yeliang[1], Zhu Lipeng[2], Hu Bin[2], Deng Song[3], Wang Dong[4]

1. School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
2. Global Energy Interconnection Research Institute, Beijing 102209, China
3. Institute of Advanced Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
4. State Grid Nantong Power Supply Company, Nantong 226001, China

Abstract

Cloud computing emerges as a new computing pattern that can provide elastic services for any users around the world. It provides good chances to solve large scale scientific problems with fewer efforts. Application deployment remains an important issue in clouds. Appropriate scheduling mechanisms can shorten the total completion time of an application and therefore improve the quality of service (QoS) for cloud users. Unlike current scheduling algorithms which mostly focus on single task allocation, we propose a deadline based scheduling approach for data-intensive applications in clouds. It does not simply consider the total completion time of an application as the sum of all its subtasks' completion time. Not only the computation capacity of virtual machine (VM) is considered, but also the communication delay and data access latencies are taken into account. Simulations show that our proposed approach has a decided advantage over the two other algorithms.

Keywords virtual machine placement, cloud computing, data intensive, deadline based

1 Introduction

Cloud computing represents a new computing model following the parallel computing, distributed computing and grid computing [1]. In a cloud computing system, computation resources are provided on demand and are distributed on servers of a large scale. In such a scenario, an application can not only access data that is located on local servers, but also communicate with another application that is assigned to a remote server. And there can be tremendous computation resources in a cloud system [2]. A cloud data center may contain tens of thousands of servers. For instance, the Microsoft Chicago-based Data Center supports the Microsoft Live online services. It spans over 65 000 m^2 and is one of the largest data centers in the world. Also, Google possesses more than one million servers all over the world (http://www.datacenterknowledge.com/inside-microsofts-chicago-data-center). Therefore, the bandwidth of the network connecting these servers has a great impact on the execution efficiency of cloud applications.

Currently, virtualization is the crucial technology in cloud computing [3]. This technology changes the way how the end user's requests are processed. Requests from end users can be divided into multiple subtasks. And these tasks are then assigned to different VMs. A VM can be treated as a collection of a task and the requested physical resources like random access memory (RAM), central processing unit (CPU) and bandwidth. All of the VMs will be placed in specific computation nodes to process tasks. Moreover, these VMs can be migrated among computation nodes and therefore improve the utilization of computation resources [4]. In this way, thousands of physical servers can be composited as a pool of computation resources and provide elastic services for cloud users [5]. These servers may be of poor or indifferent quality so that any average cloud providers can afford the expense, and computation

Received date: 12-08-2016
Corresponding author: Fu Xiong, E-mail: fux@njupt.edu.cn
DOI: 10.1016/S1005-8885(16)60064-X

resources like CPU and RAM also can be made the best of. Cloud users can take advantage of the flexible and scalable cloud services and do not need to install the computation resources directly on his/her system [6]. Therefore, thousands of common users or scientific workers resolve their complex applications using cloud computing.

A VM that executes a task is placed on a computation node so as to execute the users' requests. At the same time, data can be stored in storage nodes that have some logical or geographical distances away from the computation node [7]. For a given application in a cloud computing system, its requested data might be spread in many vastly distributed storage nodes. In such a situation, the execution performance of an application could be significantly affected by the network input/output (I/O) performance among storage nodes that store the requested data. Moreover, many cloud applications are data intensive, meaning that they often need to communicate with related data frequently. The total completion time of a data intensive cloud applications is an important performance metric. It can be affected by many factors like the task scheduling mechanism, the load on servers, communication and data access latencies. Network I/O performance can significantly affect the data access latency which is one of the dominating factor that affects the total completion time. And it is discussed in many Refs. [8–10]. It is important to shorten total completion time for data intensive applications. Currently, there are many VM placement algorithms concentrate on this aspect [11–13]. Yet, most of them only focus on the single task scheduling. In a more realistic cloud environment, there are workflow applications in which some subtasks are connected to each other. There might be communications between these tasks. For example, task A is executed only if the data outputted by task B is ready. In that case, task A can not be executed until task B is finished. Moreover, applications are often divided into multiple subtasks. Some of these tasks can be executed concurrently. Therefore, considering the total completion time of an application simply as the sum of execution time of subtasks is not applicable [8–9,14].

In this paper, we propose a novel application scheduling approach in a cloud environment. Not only the computation capacity of VMs is considered, but also the communication delay and data access latencies are taken into account. The total completion time of a subtask consists of three parts: the execution time specifies the time that being processing on a VM. The file access time represents the time that data being transferred between the computation nodes and the storage nodes. And the communication time that data being transferred from the preceding task to the current task. The structure of an application is modeled as the directed acyclic graph. And the total completion time of the application is the critical path in the graph. The objective of the approach is to shorten the critical path and therefore can cut down the total completion time of the application. We have compared the approach with other two algorithms. The simulations show that the proposed approach has a decided advantage over the two algorithms.

The remaining part of the paper is organized as follows. We will discuss related work in Sect. 2. Sect. 3 will present system model. The task scheduling algorithm will be introduced in Sects. 4. And we will present experiment results in Sect. 5. At last, conclusions and future work is shown in Sect. 6.

2 Related work

Computation resources management and VM placement are problems of importance in a cloud system. VM placement problem is an NP-hard variant of the N-dimensional bin-packing problem which could not be solved by any polynomial algorithms [15]. A lot of researchers have made great efforts to address the problems in this field. Generally speaking, current VM placement algorithms focus on promoting the efficiency of computation resources utilization [16–17], adopting data management mechanisms like caching or replication to ensure the fault tolerance or shorten the data access latencies [18–19], improving load balance of servers [20–21], and cutting down the energy consumption [22–23]. In this paper, a task scheduling algorithm is proposed for workflow management in cloud environments.

Wang et al. [20] introduces a load balancing task scheduling algorithm for large scale datacenters. It can keep load balancing of each server and obtain a task scheduling sequence with shorter job makespan. In Ref. [21], Shahapure et al. present a load balancing with optimal cost scheduling algorithm. It can process a task with minimum execution cost. However, the valuable bandwidth resource is not considered. A network aware VM placement and migration approach is proposed in Ref. [8]. It selects servers with faster data transmission speed so as to cut down data access latencies for data

Download English Version:

https://daneshyari.com/en/article/7116925

Download Persian Version:

https://daneshyari.com/article/7116925

Daneshyari.com