The Journal of Systems and Software 115 (2016) 144–156

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Energy-aware dynamical hosts and tasks assignment for cloud computing

Yean-Fu Wen*

Graduate Institute of Information Management, National Taipei University, New Taipei City, Taiwan, ROC

ARTICLE INFO

Article history:
Received 29 June 2015
Revised 2 December 2015
Accepted 22 January 2016
Available online 12 February 2016

Keywords:
Energy efficiency
Load balance
Performance
Scheduling
Threshold

ABSTRACT

One feature of MapReduce is to split user request into multiple tasks and then process around multiple datacenters for cloud computing. This study addresses an energy efficiency problem of dynamic cloud hosts (CHs) and task assignments as well as a subset of CH power-on or suspended schedules by controlling the range between the power-on and suspended thresholds for high-energy efficiency. A dynamical CHs and tasks assignment scheme is proposed to reduce the overall system energy consumption. The main concept of the proposed scheme entails setting the thresholds to satisfy the constant and variable traffic loads, nodal load balance, migration overhead, basic required power, and processing power. The reason is the established energy consumption required for initialing power-on and variable rates to keep working. This work evaluates the proposed scheme and compares it with the CHs and tasks assignment schemes to show how the proposed scheme achieves energy efficiency. The simulation results show that the proposed scheme obtains the lowest energy consumption under the tolerable responding time constraints even though the request traffic load is varying. The average improvement rate is 16.3% to balance the number of active hosts and migration overhead as well as 4.8% for task schedule.

1. Introduction

Cloud computing provides services for multiple user programs and virtual machines (VMs) running on a cloud host (CH) simultaneously (Chen et al., 2015; Shuja et al., 2014). One of the main ideas for VMs is to enhance the system utilization (Tsai and Rodrigues, 2014). High resource utilization requires few numbers of CHs so that the total own cost is reduced. However, it results in high energy consumption and responding time when the traffic load is high. The power consumption rate is large if the system load is high with extra buffering, memory swapping in/out, and content switching etc. Although a low-load host consumes less energy, the required energy to keep the host working is large (Bertinia et al., 2010). The total energy consumption is large if there are many low-load active hosts. Hence, how to (i) balance the traffic load on networks and hosts and (ii) dynamically assign serving hosts within a reasonable range becomes an important energy efficiency problem.

Generally, the overall energy consumption rate decreases when the given aggregated loads are distributed among CHs. Because the power consumption does not follow the load linear distribution, it follows an exponential-like (namely, the high slide-rate increases as the traffic load increases) rise that the heavier loads lead to the increasing power consumption exponentially (Chen et al., 2005; Kliazovich et al., 2010). However, a host requires high power consumption to power-on the system and keep operating (Bertinia et al., 2010) that means larger number of low-load hosts leads to higher energy consumption. The traffic load is distributed on a necessary number of hosts such that each host works in a reasonable ratio of initial and process energy consumption. Namely, we assign an exact number of serving hosts according to the power-on and suspending thresholds to serve the given amount of traffic loads.

MapReduce framework is a distributed mechanism, which includes Map and Reduce functions, to enhance cloud computing performance (Loughran et al., 2012; Warneke and Kao, 2011). The main idea of the Map function is to split a job into multiple tasks and distribute them into several CHs for processing. Subsequently, the computing results are summarized and collected to master server (MS) through Reduce function. This work considers how to distribute the split tasks among the CHs and aggregate the results to the MS based on MapReduce mechanism to minimize the energy consumption. Among the energy consumption of the cloud computing operations, processors, communications, and cooling systems are the major sources of energy consumption. The energy consumption on the storage is one of the major sources (Long et al., 2014).

* Tel.: +886 226748189x67719; fax: +886 26736293.
E-mail address: yeanfu@mail.ntpu.edu.tw

http://dx.doi.org/10.1016/j.jss.2016.01.032
0164-1212/

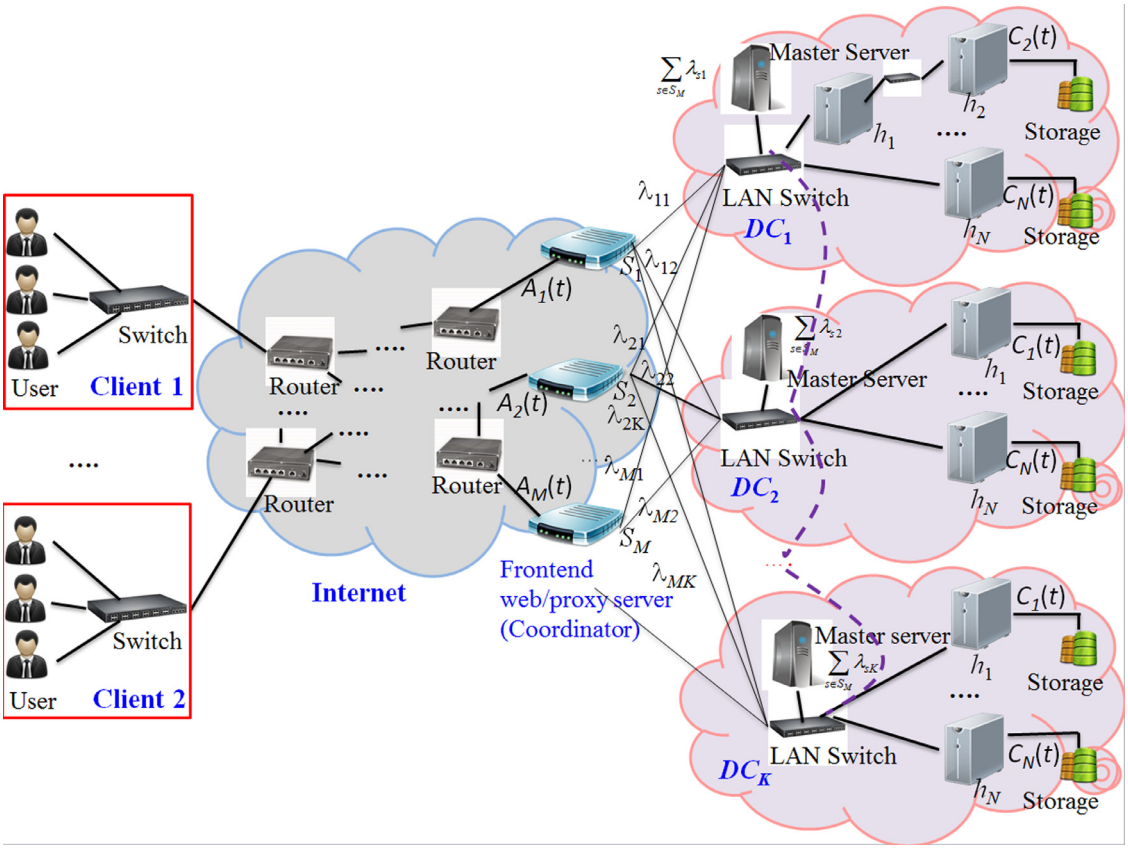

Fig. 1. A cloud network architecture.

How many hosts are allocated and how to assign the incoming tasks are addressed in this work. The problem is how to select a CH to be powered-on and a serving sever to be suspended. When the traffic load is high, the MS allocates a large number of powered-on hosts, and then, assigns the queued tasks or migrates tasks to new assigned active hosts from the neighboring hosts with VM techniques. When the traffic load is low, the MS selects some CHs to be suspended. In this step, the proposed scheme decides which host to be suspended and how to handle the processing tasks of these suspending hosts. A simple way is to select the hosts according to the lightest load. The proposed scheme migrates the remained in-progress tasks to the neighboring active hosts.

Only one threshold determines power-on and suspending a host leads to ping-pong effect, which is defined as a host power-on and suspending cycle in a short time (namely, the configuration or status changes with a high frequency of exchanging information. It is akin to a ping-pong ball traveling back and forth on a table. The ping-pong problem in this study emerged because of the minor traffic load switching between power-on and suspended CHs in a short period. Consequently, a large amount of power was wasted on power-on and suspended processes. A long adjustment cycle time might reduce ping-pong effect. However, if the adjustment cycle is too long, the method might not reflect the load status in real time especially for the busty traffic load that may result in system overload. The energy consumption is high if the traffic load cannot reflect in time. Thus, this work proposes a two-thresholds balancing strategic for power-on and suspending decision to reduce both the ping-pong effect and longtime adjustment cycle.

The gap between the power-on and suspending thresholds balances the established and low-load power consumption. The higher power-on threshold avoids wake-up a host but suspends a host in a short-time with the load is lower than the suspending threshold. As a result, an MS is busy to determine which host is waked-up and which host is suspended. The size of the gap might affect the cloud service responding time to satisfy the processing time. When the traffic load increases, the high-load level triggers wake up the required number of hosts that result in longer delay. When the traffic load decreases, several hosts are selected in low-load results in lower delay. As our best knowledge, no research discusses how to set the gap between the power-on and suspending thresholds to minimize energy consumption. Thus, this work adjusts the suspending threshold to reduce the short-term traffic load variance and satisfy the tolerable responding time.

From a viewpoint of CHs, several sets of hosts are set in several datacenters, which are located in various areas, as shown in Fig. 1. These datacenters DC_1–DC_K are connected through Internet and specific fiber network. Several frontend Web/proxy servers S_1–S_m are deployed to be the first level of coordinator. Each datacenter DC_k has one or more MSs to cooperatively process the tasks that the cloud structure is similar to the work (Lin et al., 2014), but the studied issue is different. When the load balancing among the CHs is considered, the energy consumption is not minimized. Because various power consumption rates among CHs result in varying amount of energy consumption, even though some hosts are idle. Furthermore, the amount of electric power consumed by transmitting data and processing tasks has to be considered. If a task is assigned to a CH with a long path, the processing energy should be low such that the nodal and link energy consumption are balanced. This work minimizes average energy consumption on host processing and network transmitting for all tasks.

From a viewpoint of network, multiple routing paths can be used to enhance the network performance as well as to reduce the energy consumption. If a standard shortest routing path algorithm is used, such as Dijkstra's algorithm (Cormen et al., 2001),