# An Analysis of Virtual Machine Performance for Inter- and Intra- Datacenter Resource Management under Cloud Content Delivery Network Environment

Heejae Kim, Myeongseok Hyeon, Hyungyu Jang, Chan-Hyun Youn
*Department of Electrical Engineering, KAIST*
*{kim881019,tidusqoop,h.g_jang,chyoun}@kaist.ac.kr*

## Abstract

*In this paper, we present an analysis of virtual machine (VM) performance for inter- and intra-datacenter resource management under cloud content delivery network environment. We firstly overview cloud CDN environment and present the detail description of inter- and intra- datacenter VM management. Then, an analysis of VM performance is studied by experiments for identifying performance interference of CPU, memory, disk I/O, and network I/O in various cases. The experiment result showed that performance degradation tends to be more severe if resource sharing increases. In addition, network bandwidth of a video streaming cache server converged with time, and it achieved more rapidly as resource usage is smaller. Finally, settling time of network bandwidth and the corresponding performance degradation followed the same trend in the almost all lines in the result.*

## 1. Introduction

A combination of cloud computing and content delivery networks (CDNs) is emerging in IT industry, and vendors such as Amazon, Akamai, Google, Limelight, Netflix, and Rackspace are providing related services. Recently, the vendors show the tendency to construct both cloud computing and CDN environment. For example, Amazon provides CloudFront [1] as a CDN service using its EC2 [2], and Rackspace CDN [3] is serviced on its cloud [4]. We refer to this kind of CDNs operated on clouds as cloud CDNs [5].

The mechanism of cloud CDNs includes both inter- and intra- datacenter resource management. For example, in case of cache server placement, datacenter selection and request routing are operated for multiple cloud datacenters, and virtual machine (VM) placement follows in each datacenter.

Performance interference is one of the most important considerable factors to achieve the operations effectively. The performance interference is caused by resource sharing in nodes. Especially in virtualized nodes, the modern virtualization technology fails to guarantee performance isolation perfectly. Resources such as last level cache (LLC), memory bandwidth, disk buffer, and network bandwidth are not partitioned for VMs in the same node while hypervisors guarantees independent usage of cores, memory, and disk [6], [7]. Therefore, an analysis of VM performance is necessary to achieve effective cache server placement under cloud CDN environment in terms of inter- and intra-datacenter resource management.

In this paper, we firstly overview cloud CDN environment and present the description of inter- and intra- datacenter VM management in detail. Then, an analysis of VM performance is studied by experiments for identifying performance interference of CPU, memory, and disk I/O, and network I/O related resources in various cases. In our previous work [8], we presented an analysis of network I/O performance of VMs as cache servers. As extension of it, we study CPU, memory, and disk I/O performance additionally in this paper.

The remainder of this paper is organized as follows. Section 2 describes resource management under cloud CDN environment. Section 3 presents an analysis of resource performance. Finally, section 4 concludes the paper.

## 2. Resource management under cloud CDN environment

In cloud CDN environment, cache servers are placed as VMs in cloud datacenters. In the figure, the cloud CDN provider operates cache server placement as responses of cache server requests of content providers. For the cache server placement, the cloud CDN provider operates datacenter selection and VM placement sequentially. As responses of content requests of end users, the cloud CDN provider allocates the requests to suitable cache servers via request routing, and the

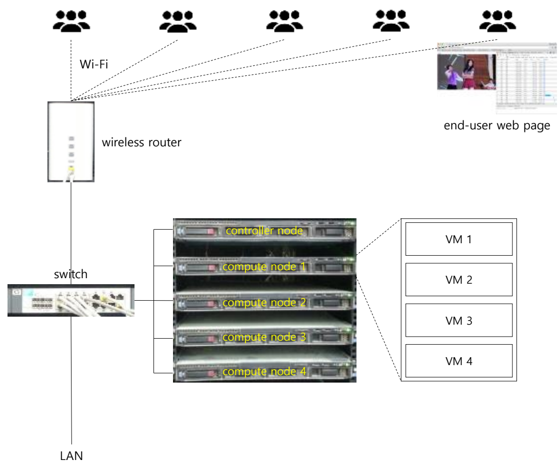



Fig. 1. Experiment environment.

selected cache servers is used to provide content for the end users.

## 2.1. Inter-datacenter VM management

In Kim et al. [9], the authors presented a Nash bargaining solution (NBS)-based datacenter selection algorithm. The optimization problem of the algorithm is formulated as depicted in (1), (2), and (3). In (1), $u(S)$ represents a utility function where $I$ is the set of cloud datacenters, $s_i$ in the vector $S$ is a decision variable and represents the number of cache servers to be placed in datacenter $i \in I$, $c$ is control parameter, $\delta_i$ is predicted end user demand in datacenter $i$ with clustering end user requests by geographical location of the datacenters, and $e$ is the number of capable end user requests in a VM respectively. (2) is the constraint to guarantee the sum of $s_i$ is equal to that of the number of cache server requests and reserved VMs where $s$ and $\rho_i$ are the number of cache server requests and reserved VMs in datacenter $i$ respectively. We note that reserved VMs represent existing VMs in datacenters when operating the algorithm. (3) is the constraint to limit the minimal value of $s_i$ in each datacenter $i$.

$$\text{maximize } u(V) = \sum_{i \in I} \log\left(v_i - \frac{c \cdot \delta_i}{e}\right) \tag{1}$$

$$\text{subject to } \sum_{i \in I} v_i - \left(s + \sum_i \rho_i\right) = 0 \tag{2}$$

$$v_i \geq \max\left(\frac{c \cdot \delta_i}{e}, \rho_i\right), \forall i \in I. \tag{3}$$

In the algorithm, $e$ affects VM performance because increase of $e$ intensifies performance interference. Therefore, it is necessary to determine $e$ in consideration of performance interference.

## 2.2. Intra-datacenter VM management

VM consolidation for reduction of power consumption and VM dispersion for reduction of performance interference are two opposite goals of VM placement. Therefore, it is necessary to mix them for effective intra-datacenter VM management.

To achieve it, Patal and Shah [10] argued the need of cost modeling of a datacenter, and constructed the cost model. In the model, the total cost in a cloud datacenter includes the space, the power and the cooling recurring, and other cost. The other cost consists of maintenance and amortization of power and cooling system, personnel, software licenses, compute equipment depreciation, and so on.

In this paper, we adds IaaS service level agreement (SLA) cost to the model. Although almost all cloud service providers only consider availability of VMs as IaaS SLA today, necessity of considering performance in IaaS SLA is raising as in [11], [12]. To achieve it, we define the SLA penalty cost (*PC*) of node $j$ as depicted in (4) where *UPC* is the unit *PC*, $v_j(t)$ is resource usage of VMs in timeslot $t$, and $\bar{d}_j(t)$ is the average performance degradation in timeslot $t$.

$$PC_j(t) = UPC \cdot v_j(t) \cdot \bar{d}_j(t). \tag{4}$$

Finally, we describe the partially total cost (*PTC*) which is composed of costs only depending on VM placement in the total cost as depicted in (5). We note that the *OC* is an abbreviation of the PM operating cost including the power and the cooling recurring cost. In (5), $J$ is the set of PMs in a cloud datacenter, *UOC* is the unit *OC*, $V_j(t)$ is a vector of $v_j(t)$, and $p_j(V_j(t))$ is power consumption of node $j$ when resource usage in timeslot $t$ is $V_j(t)$.

$$\begin{aligned} PTC(t) &= \sum_{j \in J} OC_j(t) + PC_j(t) \\ &= \sum_{j \in J} UOC \cdot p_j(V_j(t)) + UPC \cdot v_j(t) \cdot d_j(t). \end{aligned} \tag{5}$$

Based on the cost model, we formulate an optimization problem as depicted in (6) and (7).

$$\text{minimize } \limsup_{t \to \infty} \frac{1}{t} \sum_t \sum_{j \in J} OC_j(t) \tag{6}$$

$$\text{subject to } \limsup_{t \to \infty} \frac{1}{t} \sum_t \sum_{j \in J} PC_j(t) < \delta. \tag{7}$$

To solve the optimization problem, we should consider time variance of the *OC* and the *PC*. Therefore, we need to predict power consumption and performance degradation in next timeslot. To achieve accurate prediction of performance degradation, it is necessary to identify performance interference.

Table. 1. Measured VM performance. $x$ = 0, 1, 2, 3 represents executing Compress-7zip, Cachebench, Bonnie++, and the video streaming cache server in VM 1 respectively, and $y$ = 0, 1, 2, 3 represents executing each of the applications in the other VMs respectively. $n$ represents the number of VMs which run the corresponding $y$.

| $x$ | | 0 (MIPS) | | | | 1 (MB/s) | | | | 2 (KB/s) | | | | 3 (ms) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $n$ | | 0 | 1 | 2 | 3 | 0 | 1 | 2 | 3 | 0 | 1 | 2 | 3 | 0 | 1 | 2 | 3 |
| $y$ | 0 | 2801 | 2624 | 2563 | 2332 | 7353 | 7006 | 6949 | 6446 | 32428 | 29970 | 29248 | 29237 | 220595 | 223811 | 222687 | 223674 |
| | 1 | | 2698 | 2673 | 2020 | | 6493 | 4314 | 2913 | | 31932 | 31449 | 29481 | | 221652 | 223008 | 220740 |
| | 2 | | 2664 | 2613 | 2600 | | 6269 | 6144 | 6160 | | 15818 | 7033 | 4978 | | 231943 | 253414 | 258503 |
| | 3 | | 2777 | 2698 | 2679 | | 7123 | 7021 | 6775 | | 29540 | 26839 | 19490 | | 230158 | 234090 | 268448 |

# 3. An analysis of VM performance

## 3.1. Experiment setting

For an analysis of VM performance, we constructed the experiment environment as shown in Fig. 1. In the experiment, OpenStack [13] was used to build cloud environment. The cloud consists of one controller node and four compute nodes, and each node has two quad-core process with hyper threading (Intel® Xeon® Processor E5620), 14 GB of memory, and 1 TB of disk. In compute node 2, we created four VMs with 1 VCPU, 2 GB of memory, and 80 GB of disk. The experiment was conducted by executing Compress-7zip (CPU-intensive job) [14], Cachebench (memory-intensive job) [15], Bonnie++ (disk-intensive job) [16], a video streaming cache server (network-intensive job) using NGINX [17]. To identify performance interference of CPU, memory, and disk I/O related resources, we measured performance of VM 1. Also, in case of network I/O, we measured streaming completion time and network bandwidth of video streaming.

## 3.2. Result and discussion

Table 1 shows measured VM performance with respect to $x$ and $n$. In the figure, the result shows that performance degradation of VM 1 tends to be more severe as the number of VMs which runs the same application with VM 1 increases. Performance degradations of VM 1 where $x = y$ results 28 %, 60 %, 85 %, and 18 % for $x$ = 0, 1, 2, 3 respectively. However, if $x \neq y$, performance degradation is much smaller than the case of $x = y$.

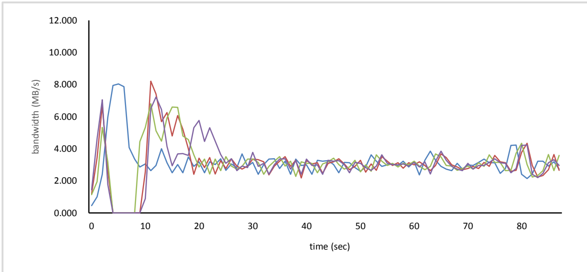
Fig. 2 shows network bandwidth of the video streaming cache server with respect to time. In the figure, network bandwidth converges with time, and it achieves more rapidly as $n$ is smaller. The result shows that the server tries to keep the certain speed which guarantees seamless execution in this environment. In addition, settling time and the corresponding performance degradation in Table 1 follows the same trend in the almost all lines.

# 4. Conclusion

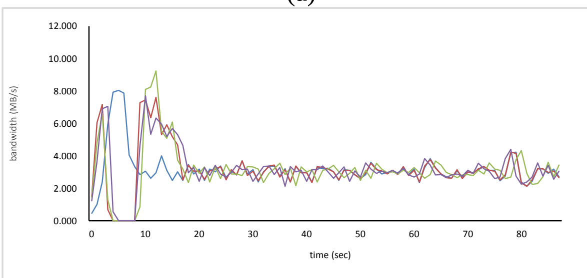
In this paper, we presented an analysis of VM performance for inter- and intra- datacenter resource management under cloud content delivery network environment. The experiment result showed that performance degradation tends to be more severe if resource sharing increases. In addition, network bandwidth of the video streaming cache server converged with time, and it achieved more rapidly as resource usage is smaller. Finally, settling time of network bandwidth and the corresponding performance degradation followed the same trend in the almost all lines in the result. We expect that the result is applied to design the algorithm for inter- and intra- datacenter resource management.


# Acknowledgement

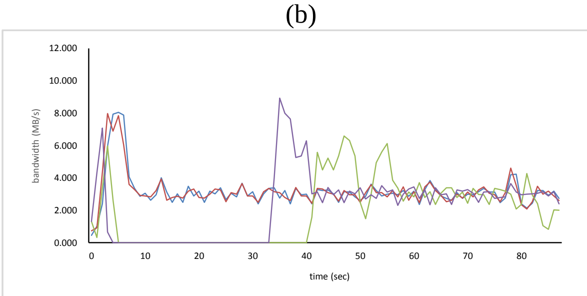
This research was supported by Next-Generation Information Computing Development Program through the NRF funded by the Ministry of Education, Science and Technology (2010-0020732), the MSIP under the ITRC (Information Technology Research Center) support program (NIPA-2014(H0301-14-1020)) supervised by the NIPA (National IT Industry Promotion Agency), and the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014.



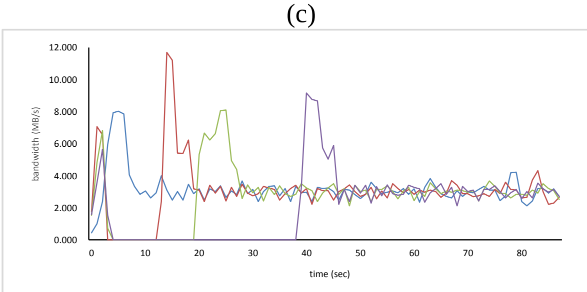

(a)



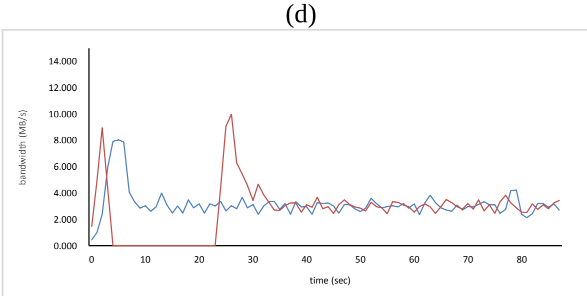

(b)




(c)




(d)




(e)

Fig. 2. Network bandwidth of the video streaming server. Blue, red, green, and purple represent $n$ = 0, 1, 2, 3 respectively. (a) $y$ = 0. (b) $y$ = 1. (c) $y$ = 2. (d) $y$ = 3. (e) executing Compress-7zip, Cachebench, Bonnie++, and the video streaming cache server.

## References

[1] CloudFront, available: http://aws.amazon.com/ko/cloudfront/.

[2] Amazon EC2, available: http://aws.amazon.com/ec2/.

[3] Rackspace CDN, available: http://www.rackspace.com/cloud/cdn-content-delivery-network/.

[4] Rackspace Managed Cloud, available: http://www.rackspace.com/cloud/.

[5] F. Chen, K. Guo, J. Lin, and T. L. Porta, "Intra-cloud Lightning: Building CDNs in the Cloud," in IEEE INFOCOM, 2012.

[6] Y. Koh, R. Knauerhase, P. Brett, M. Bowman, W. Zhihua, and C. Pu, “An Analysis of Performance Interference Effects in Virtual Environments,” in IEEE ISPASS, 2007.

[7] L. Wang, J. Xu, and M. Zhao, “Modeling VM Performance Interference with Fuzzy MIMO Model,” in USENIX Feedback Computing, 2012.

[8] M. Hyeon, H. Kim, and C. H. Youn, “An analysis of Network I/O Performance for Effective VM Management under Cloud CDN Environment,” in KIPS Spring Conference, 2015.

[9] H. Kim, Y. K. Ha, and C. H. Youn, “Nash Bargaining Solution-based Datacenter Selection under Cloud Content Delivery Network Environment,” submitted to IEEE COMSOC MMTC E-Letter, 2015.

[10] C. D. Patal and A. J. Shah, “Cost Model for Planning, Development and Operation of a Data Center,” Technical report HPL-2005-107(R.1), 2005.

[11] S. Baset, “Cloud SLAs: Present and Future,” ACM SIGOPS, vol. 46, no. 2, 2012.

[12] J. M. Myerson, “Best Practices to Develop SLAs for Cloud Computing,” IBM developerWorks, 2013.

[13] OpenStack, available: http://www.openstack.org/.

[14] Phoronix Test Suite, available: http://www.phoronix-test-suite.com/.

[15] LLCbench, available: http://icl.cs.utk.edu/projects/llcbench/cachebench.html/.

[16] Bonnie++. Available: http://www.coker.com.au/bonnie++/.

[17] NGINX, available: http://www.nginx.org/.