Delft University of Technology
Parallel and Distributed Systems Report Series

Statistical Characterization of Business-Critical Workloads Hosted in Cloud Datacenters

Siqi Shen, Vincent van Beek, and Alexandru Iosup
{S.Shen,A.Iosup}@tudelft.nl, Vincent.vanBeek@bitbrains.nl

version 2014-July

report number PDS-2014-001

ISSN 1387-2109

Published and produced by:
Parallel and Distributed Systems Section
Faculty of Information Technology and Systems Department of Technical Mathematics and Informatics
Delft University of Technology
Mekelweg 4
2628 CD Delft
The Netherlands

Information about Parallel and Distributed Systems Report Series:
reports@pds.twi.tudelft.nl

Information about Parallel and Distributed Systems Section:
http://pds.twi.tudelft.nl/

Abstract

Business-critical workloads—web servers, mail servers, app servers, etc.—are increasingly hosted in virtualized datacenters acting as Infrastructure-as-a-Service clouds (cloud datacenters). Understanding how business-critical workloads demand and use resources is key in capacity sizing, in infrastructure operation and testing, and in application performance management. However, relatively little is currently known about these workloads, because the information is complex—large-scale, heterogeneous, shared-clusters—and because datacenter operators remain reluctant to share such information. Moreover, the few operators that did share data (Google, Cloudera, many supercomputing centers, etc.) have enabled studies in business intelligence (MapReduce), search, and scientific computing (HPC), but not in business-critical workloads.

To alleviate this situation, in this work we conduct the first comprehensive study of business-critical workloads hosted in cloud datacenters. We collect 2 large-scale and long-term workload traces corresponding to requested and actually used resources in a distributed datacenter servicing business-critical workloads. Our comprehensive datasets focus on four key types of resources, all of which can become bottlenecks for business-critical applications: CPU, disk I/O, and, rare among studies of datacenter workloads, memory and network I/O. We characterize the demand and use of these resources and conduct an analysis of basic statistics, including correlations, and of basic time-patterns.

Our study gives evidence that, for business-critical workloads, the workload is more dynamic than for other classes of hosted workloads: most of the virtual machines require less than 4 CPU-cores and less than 8 GB of memory, the requested amount of CPUs and memory size are correlated, the resource usage is low (10%) relative to the request, and and the resource usage is bursty but otherwise predictable even over the short-term. Last, we also release the traces for public use via the open-access Grid Workloads Archive.

Contents

List of Figures

List of Tables

1 Introduction

Spurred by a rapid development of hardware and of resource management techniques, cloud datacenters are hosting an increasing number of application types. Over a billion people access daily a diverse collection of free or paid cloud utilities, from search to financial operations, from online social gaming to engineering [1, 2, 3]. To continue the adoption of cloud datacenters, and to improve the ability of the datacenter operators to tune existing and to design new resource management techniques, understanding of the workload characteristics and the underlying datacenters is key for both datacenter operators and for cloud service providers. Although some of the largest datacenter operators, that is, Google, Facebook, Microsoft, and Yahoo, have contributed workload information that was used later in valuable studies [4, 5, 6, 7], the information they have contributed represents a relatively small sub-set of the cloud service market. To better understand the workloads of cloud datacenters, in this work we collect and analyze workload traces from a distributed datacenter servicing a fundamentally different workload, that of business-critical applications of financial institutions and engineering firms.

The rapid adoption of cloud datacenters is leading to significant changes in workload structure and, as a consequence, in system design and operation. It is likely that datacenter workloads are becoming increasingly data-intensive, which may put increasingly more stress on the networking, storage, and memory resources of the datacenter [8]. In a previous study of tens of grid workloads [9], we have observed that the workload units (jobs, requests, etc.) have decreased in size, and increased in amount and possibly also interdependency, over the last decade; this could be continued in cloud datacenters [10]. In response, resource management techniques have also evolved rapidly, with new approaches in computing [11], networking [12], storage [13], and memory [14] management.

To address various resource management challenges in cloud datacenters, comprehensive workload traces are vitally needed. Following ideas expressed in the unified formats used to publish tens of workload traces representative for parallel production environments [15] and for grid computing environments [16], a broadly useful workload trace should include long-term (e.g., at least for several months) information about both the requested and the actually used resources, where the resources include at least CPU, memory, network, and storage.

The characterization of workload traces is a long-established practice that supports innovation in the design, tuning, and testing of resource management approaches. A recent study [17] uses the characteristics of workloads observed in the Microsoft datacenters to propose and validate an energy-efficient scheduler. Others describe how workload characteristics could help test the robustness of stateful cloud services [18], how MapReduce workload traces could help understand the performance of big data frameworks [5], how the characteristics of traces could help the automated selection of the datacenter scheduling policies [19, 11], etc.

Although actual data and knowledge about workload characteristics are often beneficial for datacenter operation, remarkably few workload traces are publicly available or have even been publicly characterized. Moreover, the few existing examples, albeit seminal, are not comprehensive and, because of their data source, may not be representative for the cloud datacenter industry in general. Table 1 (which we will discuss in detail in Section 6) presents an overview of several of the highest-cited studies of cloud workload traces. Overall, the traces originate from Google, Microsoft, and other giant datacenter operators (column TS), and represent workloads that may be typical for the MapReduce and other operations specific to these companies (column Workload). We also observe that few studies include information about requested resources, and rarely include network and disk information at all.

To address paucity of data and knowledge about datacenter workloads, in this work we aim to characterize the workload of a distributed datacenter servicing enterprise customers with business-critical applications (detailed in Section 2.1). Our major contribution is four-fold:

1. We collect long-term and large-scale workload traces from a distributed cloud datacenter (Section 2). The traces include information about CPU, memory, disk I/O, and network I/O, for both requested and used resources. We make available these traces (Section 9), which are representative for business-critical applications running in cloud datacenters, through the public Grid Workloads Archive [16].

Table 1: Previous work in workload trace analysis, in contrast to this study. The trace source (TS) column: F=Facebook, G=Google, Y=Yahoo, T=Taobao, BB=Bitbrains. Nodes (N) column: k indicates thousands of items. The traces (Tr) column lists the number of traces. Time (T) column: y/m/d stand for year/month/day. Resources: Mem=Memory, Net=Network.

			Scale			Requested resources				Used resources			
Study	**TS**	**Workload**	**N**	**Tr**	**T**	**CPU**	**Mem**	**Disk**	**Net**	**CPU**	**Mem**	**Disk**	**Net**
Chen et al. [20]	F	MapReduce	5k	7	1 y	—	—	—	—	yes	—	yes	—
Reiss et al. [4]	G	Mixture	12.5k	1	1 m	yes	yes	—	—	yes	yes	—	—
Chen et al. [5]	F/Y	MapReduce	2.6k	2	7 m	—	—	—	—	yes	—	yes	—
Mishra et al. [8]	G	Mixture	?	5	4 d	—	—	—	—	yes	yes	—	—
Ren et al. [21]	T	MapReduce	2k	1	2 w	—	—	—	—	yes	yes	yes	yes
Di et al. [10]	G	Grid vs Google	12.5k	1	1 m	—	yes	—	—	yes	yes	—	—
This study	**BB**	**Business critical**	**1.75k**	**2**	**4 m**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**

Table 2: Business-critical workload traces collected in this work.

Name of the trace	# VMs	Period of data collection	Storage technology	Total memory	Total cores
`fastStorage`	1,250	1 month	SAN	17,729 GB	4,057
`Rnd`	500	3 months	NAS and SAN	5,485 GB	1,444
Total	1,750	5,446,811 CPU hours		23,214 GB	5,501

2. We analyze the basic statistics of the requested and actually used resources (Section 3). We report the basic statistics, such as quartiles, mean, and standard deviation. We also contrast the basic statistics of business-critical traces with those of parallel production environments, grids, and the search and data-mining workloads of Google, Microsoft, etc.

3. We conduct a correlation study to identify possible relationships between different resources (Section 4). We also contrast the results with results of previous datacenter studies.

4. We investigate the time-patterns occurring in the resource consumption (Section 5). Specifically, we investigate the peak to mean ratio in resource usage, which we compare with previous datacenter data, and conduct an autocorrelation study of each of the recorded characteristics.

2 Dataset Collection and Method of Characterization

In this section, we introduce two traces representative for business-critical workloads collected from a distributed cloud hosting datacenter. We also present a method for characterizing such traces.

2.1 A Typical Cloud-Hosting Datacenter for Business-Critical Workloads

In this work, we study operational traces representative for business-critical workloads, that is, workloads comprised of applications that have to be available for the business to not suffer significant loss.

Business-critical applications span a broad range of user-facing and back-end services, often including email, database, CRM and collaborative, and management services. When these services experience downtime or even just low performance, they often lead to loss of revenue, of productivity, etc., and may incur financial loss, legal action, and even customer departure. By nature, business critical workloads are significantly different from the applications that are running in datacenters used by Google's web search/services [4], and by Microsoft's Messenger, shared cluster, and Azure [6] datacenters. (Our study quantifies this difference.)

A typical mid-size datacenter hosting business-critical workloads is managed by Bitbrains. Bitbrains is a service provider that specializes in managed hosting and business computation for enterprises. Customers include many major banks (ING), credit card operators (ICS), insurers (Aegon), etc. For example, a customer would request a cluster of compute nodes to run a financial risk calculation. The following requirements would come with this request: data-transfers between the customer and the datacenter via secure channels, compute nodes leased as virtual machines (VM) in the datacenter that deliver predictable performance, and high availability for running business-critical simulations.

In general, Bitbrains hosts three types of VMs: management servers, application servers, and compute nodes. Management servers are used for the daily operation of customer environments (e.g., firewalls). Examples of application servers are database servers, web servers, and head-nodes (for compute clusters). Compute nodes are mainly used to do simulation and other compute-intensive computation, such as Monte Carlo-based financial risk assessment.

2.2 Collected Traces

From the distributed datacenter of Bitbrains, we collect two traces of the execution of business-critical workloads. For this we use the monitoring and management tools provided by VMware, such as vCloud suite[1]. For each trace, the vCloud Operation tools record 7 performance metrics per VM, sampled every 5 minutes: the number of cores provisioned, the provisioned CPU frequency, the CPU usage (average usage of CPU over the sampling interval), the provisioned memory capacity, the actual memory usage (the amount of memory that is actively used), the disk I/O throughput, and the network I/O throughput. Thus, we obtain traces that cover both requested and actually used resources, for four resource types (CPU, memory, disk, and network).

We collect between August and September 2013 two traces, whose overview we present in Table 2. Combined, the traces accumulate data for 1,750 nodes, with over 5,000 cores and 20 TB of memory, and operationally acumulate over 5 million CPU hours in 4 operational months; thus, they are long-term and large-scale time series. The first trace, `fastStorage`, consists of 1,250 VMs that are connected to fast storage area network (SAN) storage devices. The second trace, `Rnd`, consists of 500 VMs that are either connected to the fast SAN devices or to much slower Network Attached Storage (NAS) devices. The `fastStorage` trace includes a higher fraction of application servers and compute nodes than the `Rnd` trace, which is due to the higher performance of the storage attached to the `fastStorage` machines. Conversely, for the `Rnd` trace we observe a higher fraction of management machines, which only require storage with lower performance and less frequent access.

The two traces include a random selection of VMs from the Bitbrains datacenter, using an uniform distribution for the probability of selecting each VM. This is motivated by the need to guarantee the absolute anonymity of individual Bitbrains customers and to not reveal the actual scale of the Bitbrains infrastructure. A similar process is used by related work presenting the workloads of Google [4, 10], where the anonymization is achieved through a normalization of resource scales and by a selection of only a part of the infrastructure; in contrast, our study is more revealing, in that it presents the full characteristics of the virtualized resources.

Our traces do not include data about arrival processes, which in a cloud datacenter could be used to describe the lifetime of user jobs or of VMs. We investigate in this work resource consumption, which replaces the notion of user jobs with resource usage counters (this also protects the anonymity of Bitbrains' users and is in line with the approach taken many previous studies [4, 10]). For VMs, business critical workloads often use the same VMs for long periods of time, typically over several months. Thus, we do not have a proper arrival process to report on (the VMs we study run throughout the duration of our traces).

2.3 Method for Workload Characterization

We conduct in this work a comprehensive characterization, of both requested and actually used resources, and using data corresponding to CPU, memory, disk, and network resources. Although VMs may change

[1]For more details, we refer to the official metrics documentation: https://www.vmware.com/support/pubs/vcops-pubs.html

Table 3: Statistics of requested and used resources for Bitbrains, and requested CPU cores (which as the same as used) for grid and parallel traces. The same information is very difficult to assemble for the studies listed in Table 1.

Trace source	Properties	Mean	Min	Q1	Median	Q3	Max	CoV	SDev
Bitbrains	Memory requested [GB]	10.7	0.0	1.27	3.98	15.59	511	2.8	29.3
Bitbrains	Memory usage [GB]	0.6	0.0	0.03	0.10	0.29	384	3.0	1.8
Bitbrains	CPU requested [GHz]	8.9	2.4	2.93	5.20	10.40	86	1.3	11.1
Bitbrains	CPU usage [GHz]	1.4	0.0	0.02	0.08	0.20	64	3.3	4.4
Bitbrains	Disk, Read throughput [MB/s]	0.3	0.0	0.00	0.00	0.00	1,411	14.9	5.2
Bitbrains	Disk, Write throughput [MB/s]	0.1	0.0	0.00	0.00	0.01	188	14.4	1.1
Bitbrains	Network receive [MB/s]	0.1	0.0	0.00	0.00	0.00	859	11.3	0.7
Bitbrains	Network transmit [MB/s]	0.1	0.0	0.00	0.00	0.00	3,193	24.0	1.5
Bitbrains	CPU cores	3.3	1	1	2	4	32	1.2	4.0
Bitbrains	`Rnd` CPU cores	2.8	1	1	2	4	32	1.1	2.9
Grid	`DAS2` [16] CPU cores	4.3	1	1	2	4	128	1.5	6.4
Grid	`Grid5000` [16] CPU cores	5.8	1	1	1	2	342	3.6	21.0
Grid	`NorduGrid` [16] CPU cores	1.1	1	1	1	1	64	1.2	1.3
Parallel	`CEA Curie` [15] CPU cores	713.3	1	4	32	256	79,808	5.8	4,116.7
Parallel	`LLNL Atlas` [15] CPU cores	423.4	8	8	64	256	9,120	2.9	1,249.0

configuration during the trace, the chance of this happening is rare in our trace (under 1%), so we show only the initial configuration of each VM present in our traces.

For the statistical characterization we use in this study three main statistical instruments[2]: basic statistics, correlations, and time-pattern analysis. For the basic statistics, we report the min and the max, the quartiles, the mean and the standard deviation (SDev), and the unitless "Coefficient of variation" (CoV, defined as the ratio of standard deviation and mean). We also report the cumulative distribution function (CDF) of the values observed for all VMs, and for the CoV observed per VM (a measure of *dynamicity* that extends previous work [4]).

To understand the correlation between the different resources, and between requested and used resources, we look at two traditional instruments: the Pearson correlation coefficient (PCC), which measures the linear relationship between two variables, and the Spearman rank correlation coefficient (SRCC), which measures the dependence between two ranked series (e.g., ranked by time). We report here overall results that summarize all VMs but also, where the process is dynamic (e.g., resource usage), the CDF and the probability density function (PDF).

To identify time patterns in our time series, we analyze for each resource type the aggregate usage over time, by summing, each hour, the average resource usage observed for all the VMs. This *aggregate resource usage* can be used to assist resource capacity planning. We plot the auto-correlation function (ACF, a strong indicator for the existence repeating patterns) of the workload traces for each aggregate resource usage.

Last, we also analyze *dynamicity* [4], expressed as the ratio of peak to mean values, which we compute for hourly and daily intervals.

3 Characterization using Basic Statistics

In this section, we analyze in turn the requested resources, the used CPU and memory resources, and the used disk and network resources. Understanding the basic statistics can lead to interesting insights into the operation of the datacenter and into the structure of business-critical workloads, and can help create benchmarks and tune

[2]For details, we refer to the free Statistics Textbook: `http://www.statsoft.com/Textbook`

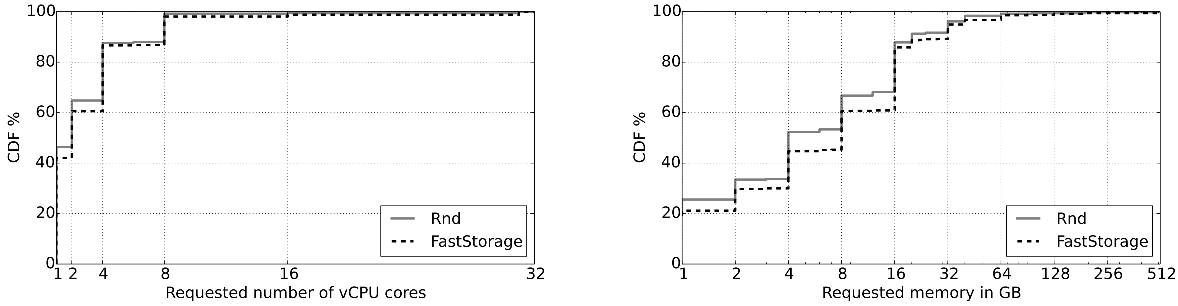

Figure 1: CDF of (left) the number of requested CPU cores, and (right) the amount of requested memory.

resource-management approaches. Table 3 summarizes the results, which are further analyzed in this section. The main findings are:

For Bitbrains, unless otherwise specified we present here only results obtained for the `fastStorage` trace; for `Rnd` results, which are very similar (for example, see Figure 1), we refer to Section 10.1 (Figure 26 to 32).

1. Over 60% of the VM requests are for no more than 4 CPU cores and 8 GB of memory (Section 3.1). Q3 of the number of requested CPU cores is 4, which indicates indicates that business critical applications domain do not fully adopt high-performance computing (HPC, especially parallel) techniques.

2. The resource usage for most VMs is dynamic. The mean CoV for resource usage range from around 1 to over 20. The lowest CoV is observed for CPU and memory—CoV values under 5 (Section 3.2).

3. On average, VMs read 4 times more than write, and use the network to send as much as they receive (Section 3.3).

3.1 Requested Resources

In this section, we analyze the requested resources (only CPU and memory, as disk and network do not record such requests). We find that VMs in our traces require on average similar amounts of CPU cores as typical grid workloads, that most of the VMs have modest requirements for CPU cores (at most 4) and allocated memory (at most 8GB), and that power-of-two requests are common.

First, we compare the CPU characteristics of VMs supporting business-critical workloads (rows labeled Bitbrains in Table 3) and of representative traces from grid and parallel production environments. The rows including "CPU cores" in Table 3 list the number of CPU cores requested (and reported as used by all resource managers) in these workloads: `fastStorage` and `Rnd` representing business-critical workloads; the `DAS2`, `Grid5000`, and `NorduGrid` datasets representing grid workloads [16]; and the `CEA CURIE` and `LLNL Atlas` datasets representing production parallel workloads [15]. If we view the VM as the unit of submitting workload, our workloads requires, on average, slightly more CPU cores than production grids (`NorduGrid`) and slightly less cores than experimental grids (`DAS2` and `Grid5000`), but significantly less CPU cores than the parallel workloads. This may indicate that business critical applications do not fully adopt HPC techniques.

We further characterize the requested resources, in terms of number CPU cores and bytes of memory allocated to each VM. Figure 1 (left) shows the cumulative distribution function (CDF) of the number of CPU cores requested per VM. For both the number of CPU cores and the amount of memory our results show that a large percentage (more than 60%) of the VMs have low requirements (2 or 4 CPU cores for our two traces, and less than 8 GB of memory). Most of the VMs (over 90%) use power-of-two cores.

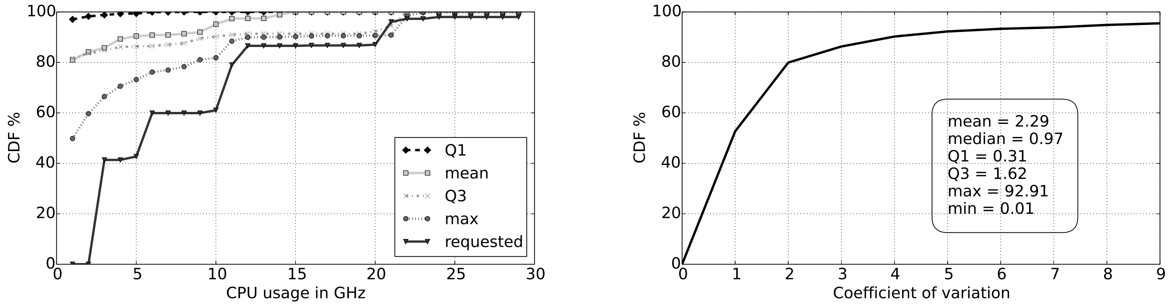

Figure 2: CPU usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

Other studies [22] show the power-of-two scale-up behavior, which seems to be historically an artifact of the architecture of parallel architectures and algorithms. VMs in our datasets use from 1 up to 32 cores (small-scale HPC), but over 95% of the VMs use at most 8 cores, and over 85% of the VMs use at most 4 cores. On average, VMs in the `Rnd` dataset use slightly fewer cores, which we ascribe to the higher density of management VMs in the `Rnd` trace—typically, management VMs require only 1, rarely more cores.

Regarding memory requirements, we observe similar patterns as for CPU requirements. Figure 1 (right) shows the CDF for the requested memory of each VM. Memory is often provisioned in power-of-two quantities (around 90% for memory). For the `fastStorage` dataset, the requested memory can range from 1GB to 512GB per VM, but most VMs use a relatively small amount of memory: over 95% of the VMs use at most 32 GB of memory, and over 70% of the VMs use at most 8 GB of memory. The VMs in the `Rnd` dataset demand slightly less memory than in the `fastStorage` dataset, which we ascribe again to the difference in management VM density—typically, management VMs use 1GB or less memory.

3.2 CPU and Memory Usage

In this section, we analyze the CPU and memory resource usage, for which we report both the CDF observed across all VMs, and the CDF of CoV observed per VM. We find that CPU usage is low on average and can be dynamic, and much lower (around 10% for most VMs) than the requested CPU bandwidth. We also find that memory usage is even lower on average but less dynamic than CPU usage.

We study first the CDF of CPU usage, across all VMs. Figure 2 (left) shows the CDF, computed across all VMs, from their observed CPU usage—first-quartile (Q1), mean, third-quartile (Q3), and maximal (Max) CPU usage—and from their requested *CPU bandwidth* (computed as the product of the number of CPU cores and the bandwidth of each core, e.g., 4×2.6 GHz for 4 cores at 2.6 GHz each). For most (about 80%) of the VMs, the mean CPU usage (curve "mean" in Figure 2 (right)) is lower than 0.5 GHz and lower than 10% of the requested CPU bandwidth. Only for less than 5% of the VMs, the mean CPU usage is higher than 50% of the requested CPU bandwidth. About 50% of the VMs have a maximal usage (curve "max" in Figure 2 (left)) lower than 1.3 GHz. Only 30% of the VMs have a maximal usage higher than 2.8 GHz. These observations suggest that for most VMs *the usage is low most of the time.*

We study next the CDF of the CoV in the observed CPU usage, observed per VM. Figure 2 (right) depicts this statistic, indicating that half (50%) of the CoV for CPU usage is lower than 1. This indicates that, for half of the VMs in our traces, the CPU usage is stable and centered around the mean—these VMs have predictable CPU usage. However, there is still a significant amount (about 20%) of VMs whose CoV for CPU usage is higher than 2—the CPU usage of these VMs is dynamic and unpredictable.

We now study the CDF of memory usage, across all VMs. We construct Figure 3 (left) similarly to Figure 2 (left), but with data about used memory. We find that the memory usage is low: on average, 80% of the

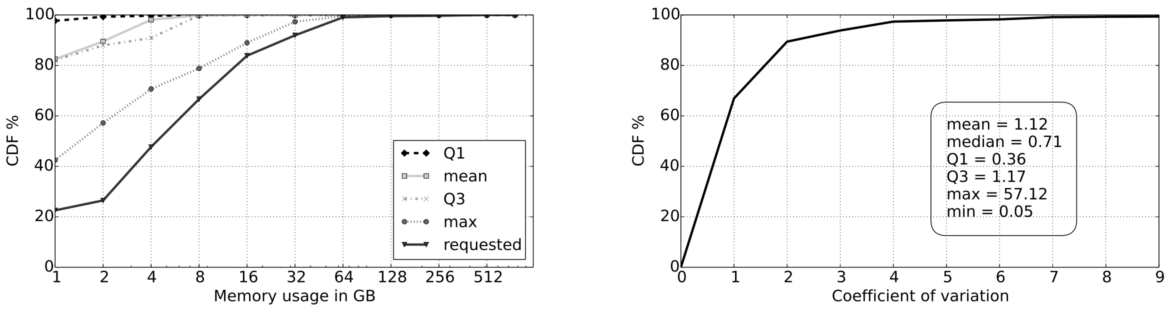

Figure 3: Memory usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

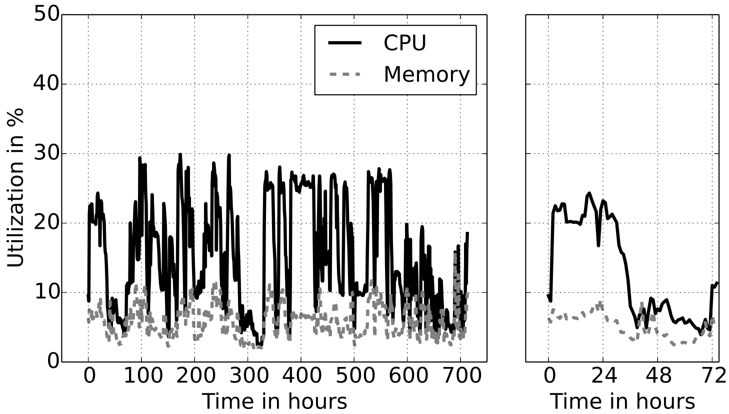

Figure 4: CPU and memory usage over time.

VMs use less than 1 GB of memory, and most (about 80%) of the VMs have maximal memory usage lower than 8 GB of memory. In Figure 3 (left), The large gap between the “mean” and the “max” curves indicates that the peak memory usage of each VM is much higher than its average usage; we investigate this in more detail in Section 5.1.

Similarly to our study for CPU usage, we investigate next the CDF of the CoV in the observed memory usage, per VM. As Figure 3 (right) shows, the memory usage is less dynamic than CPU usage: about 70% of VMs (vs only 50% for CPU usage) have a CoV for memory usage lower than 1. Most (about 90%) of the VMs have a CoV lower than 2. A similar observation has been reported for the Google trace [4].

It is interesting to study the CPU and memory usage, together; their progress over time can indicate opportunities for VM consolidation and datacenter efficiency. We depict these metrics, over time, in Figure 4. We find that CPU utilization is higher than memory utilization, which is the opposite of the finding of Di et al. [10] for the Google trace. This finding is consistent with our earlier observation that business-critical workloads are more in line with grid workloads, where CPU utilization is the typical bottleneck, and indicates that different strategies for datacenter efficiency may be needed for business-critical workloads, in contrast to Google-like search and services workloads.

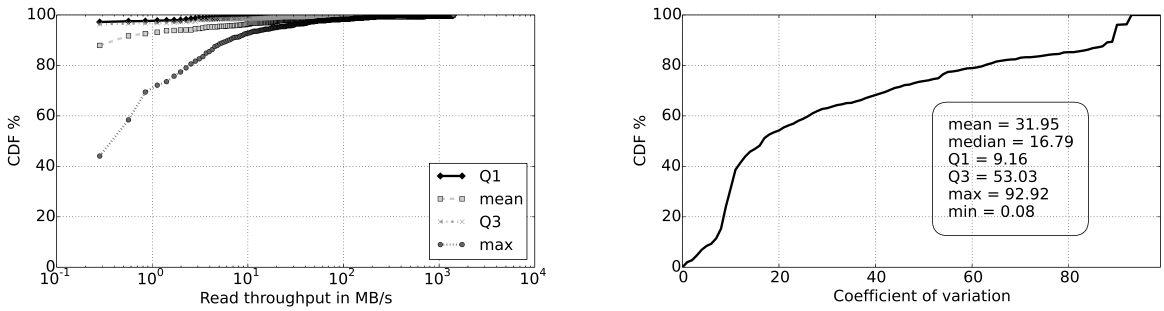

Figure 5: Disk read usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

3.3 Disk and Network Usage

Similarly to Section 3.2, in this section we analyze the disk and network resource usage, for which we report both the CDF observed across all VMs, and the CDF of CoV observed per VM. We study, in turn, the disk read and write usage. For network usage we find similar results as for disk usage, the graphs can be found in Figure 7 and 8. We find that most VMs have bursty disk and network accesses.

We study the the CDF of disk read usage, across all VMs, which we depict in Figure 5 (left). We find that most of the VMs only read sporadically: about 95% of the VMs perform three-quarters of their disk reads ("Q3" curve) at less than 0.1 MB/s. The mean value and especially the maximal value of disk reads of most VMs is much higher than the Q3 value, which indicates that disk reads are bursty. The CDF for CoV of disk reads is plotted in Figure 5 (right). The disk read usage is much more dynamic than the CPU usage: only 15% of VMs have their disk-read CoV under 1, and about 50% of the VMs have their disk-read CoV higher than 2. This may be due to application behavior, e.g. backup tools may act periodically, financial modeling tools read large volumes of financial data into memory at the start of simulations, etc.

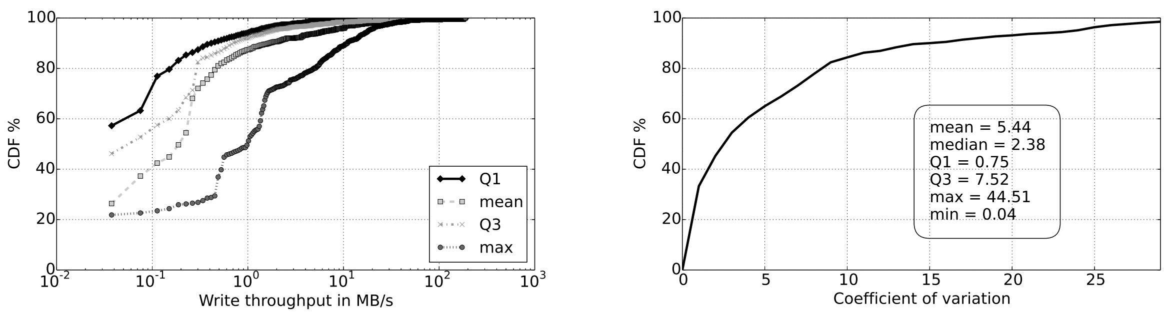

Figure 6: Disk write usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

Similarly to disk reads, we study now the disk write usage. The results, which we depict in Figure 6 (left), are similar in trend for disk reads and writes: most of the VMs do not write most of the time, but some VMs show very high peak disk write usage. On average, each VM's disk write usage is about 0.1 MB/s, which is about one fourth of the disk read usage (0.4 MB/s). Comparing to disk reads, we observe that disk writes are less dynamic, as shown in Figure 6 (right).

Similarly to disk behavior analysis, we study the network usage, expressed in terms of received and transmitted data. Figure 7 (left) shows the receive data over the network, for each VM. Most of the VMs mean

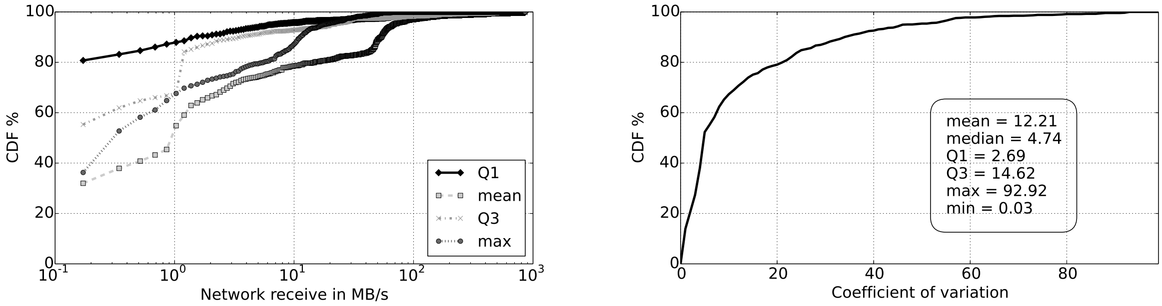

Figure 7: Network receive usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

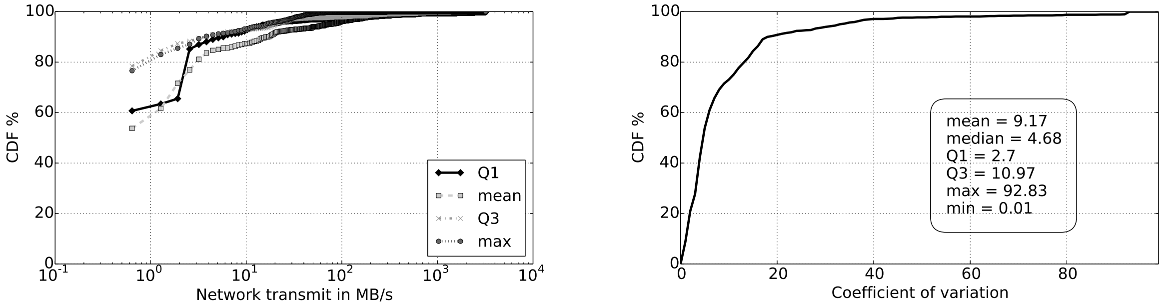

Figure 8: Network transmit usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

amount of data received (Figure 7) or transmitted (Figure 8) over the network is low. Most of the VMs mean amount of data received or transmitted over the network is low. About 80% of the VMs receive less than 30 KB/s and transmit less than 10 KB/s. The large gap between the max and the other percentiles, as observed per VM, indicates the bursty nature of network traffic. The amounts of both the received and transmitted are much more dynamic than the CPU usage.

4 Characterization of Correlations

In this section, we analyze the pair-wise correlation between the requested resources (e.g., requested CPU and memory), the correlation between the request demand and the actual usage, and the pair-wise correlation between used resources (e.g., between the CPU and memory usage). The main findings are:

1. CPU and memory are strongly correlated for requests (Section 4.1), but much less correlated for usage (Section 4.2).
2. Request and use are very weakly correlated (Section 4.1).

4.1 Correlation of Requested Resources

In this section, we investigate the correlation between the two types of requested resources, CPU and memory, and find a strong correlation between them. We also investigate the correlation between requested and used resources, and find a very weak correlation.

For the `fastStorage` dataset, the PCC and SRCC between the number of CPU cores and memory are 0.81 and 0.90, respectively. For the `Rnd` dataset, the PCC and SRCC are 0.82 and 0.85, respectively. This indicates that VMs with high values for the requested CPU tend to also have high values for the requested memory, especially for VMs (in the `fastStorage` dataset). We confirm this result through an interview with the engineers of Bitbrains, confirming that Bitbrains typically maps either 2 GB or 4 GB memory to a CPU core, depending on the physical CPU-to-memory ratio of the underlying physical infrastructure. For memory-intensive workloads, they set the memory to 16 GB per core. At the other extreme of the CPU-to-memory ratio, small VMs (1 GB or less memory) are typically management VMs that are needed to operate the customer environments.

For both the `fastStorage` and the `Rnd` datasets, the requested and the used resources are weakly correlated. This is indicated visually by the top plots of Figures 2, 3, and 5. We analyze here the data for the former two, in turn; the analysis for the latter reveals similar trends.

Figure 2 shows that: about 80% of VMs have an average CPU usage lower than 10% of their allocated CPU bandwidth, and less than 5% of the VMs have a mean CPU usage that is higher than 50% of the allocated CPU bandwidth. About 50% of the VMs have a maximal usage lower than 20% of allocated CPU bandwidth and only 30% of the VMs have a maximal usage higher than 70% of allocated CPU bandwidth. These observations suggest that most of the VMs' CPUs are idle most of the time.

Figure 3 shows that: 60% of the VMs have an average memory usage lower than 10% of the requested amount of memory, the most memory-intensive VM use only 80% of its requested memory; only 14% of the VMs have a peak memory usage that reaches their requested amount, about 50% of the VMs have a peak memory usage lower than 50% of the requested memory. This indicates that the memory utilization is low most of the time.

Similarly, the bursty nature and the poor visual correlation between the curves depicted in Figure 4, indicates poor correlation between requested and used resources.

4.2 Correlation of CPU and Memory Usage

We analyze the correlation of CPU usage and memory usage, for which we report an average correlation. Because both CPU and memory usage vary over time, we also report CDFs and PDFs of the correlation observed over time, per VM. We find strong correlation between high CPU and memory usage, that is, VMs that exhibit high CPU usage are very likely to also exhibit high memory usage. However, the temporal correlation is much weaker: it is less likely that VMs exhibit high CPU and memory usage *at the same time*. This gives, for the future, interesting opportunities to host business-critical workloads more efficiently inside the datacenter.

We analyze first the average correlation, that is, the correlation between mean CPU and mean memory usage. For the `fastStorage` dataset, the PCC and SRCC of the mean CPU usage and mean memory usage, per VM, are 0.83 and 0.84, respectively. For the `Rnd` dataset, the PCC and SRCC are 0.72 and 0.83, respectively. This indicates that VMs with high CPU usage tend to have high memory usage. Ren et al. [21] report that for the Taobao system the PCC between CPU usage and memory usage is 0.76, which falls within the same range as our overall result.

Although the average correlation indicates strong correlation, we observe that the temporal nature of both CPU and memory usage requires more in-depth analysis. We thus report CDFs and PDFs of the correlation observed over time, per VM, e.g., we collect data about the CPU-memory correlation for each sampling point (every 5 minutes, as indicated in Section 2.2), and we analyze the CDF and the PDF of this dataset.

In Figure 9 we show the distribution of PCC and SRCC for CPU usage and memory usage. The mean PCC for CPU and memory usage for the `fastStorage` dataset is 0.4, this is much lower than Ren et al. [21] report and also much lower than we found if we compare CPU and memory required or used on average.

From both the results of the correlation analysis and from the analysis of resource usage in Section 3.2, it follows that there is a big discrepancy between requests and actual usage. A similar result was found by Reiss et al. [4] for the Google trace, where only about 50% of the requested resource are actually used by

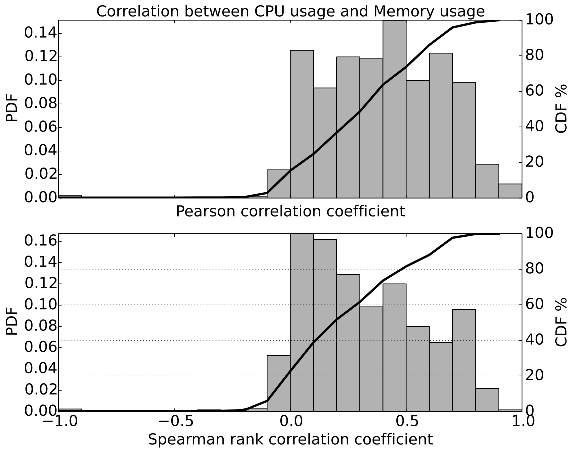

Figure 9: Correlation between CPU usage and memory usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

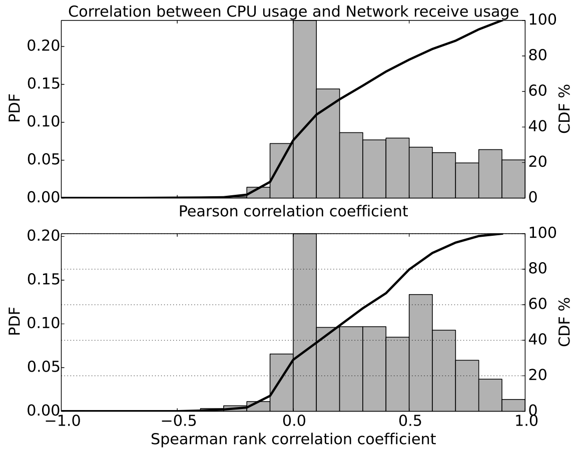

Figure 10: Correlation between CPU usage and network receive usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

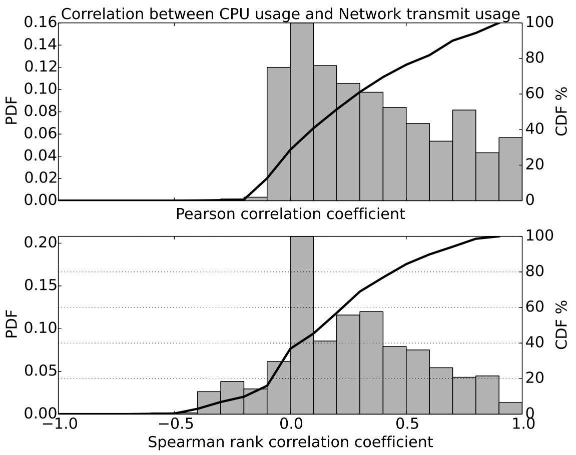

Figure 11: Correlation between CPU usage and network transmit usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

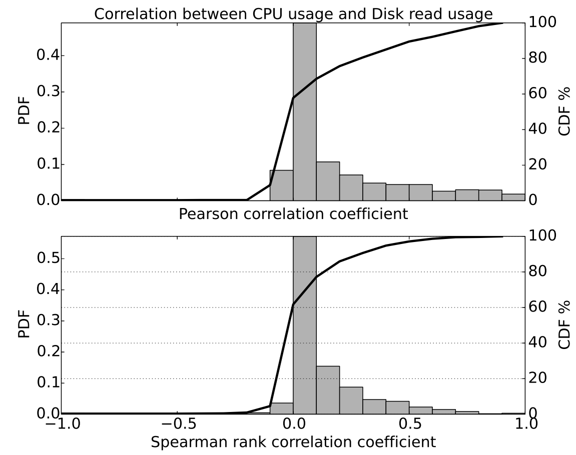

Figure 12: Correlation between CPU usage and disk read usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

jobs. As we have shown in Section 4.1, this ratio is even lower for business-critical workloads. This implies, for business-critical workloads, excellent opportunities for efficient resource management approaches exist.

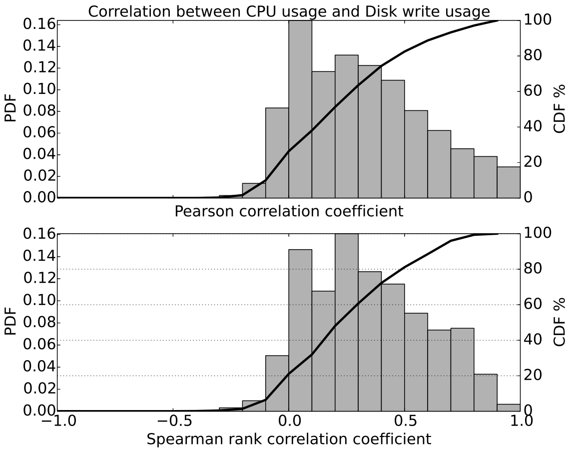

Figure 13: Correlation between CPU usage and disk write usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

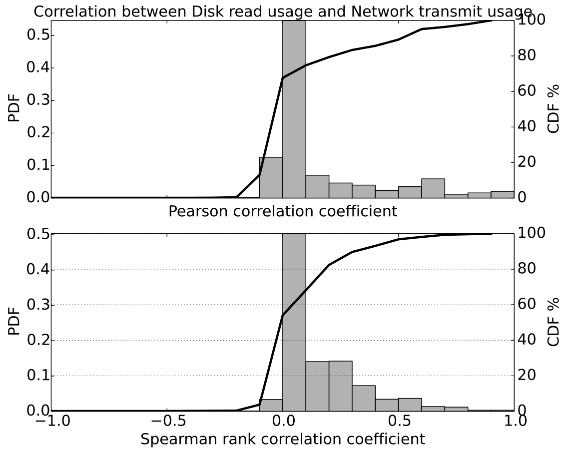

Figure 14: Correlation between disk read and network transmit: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

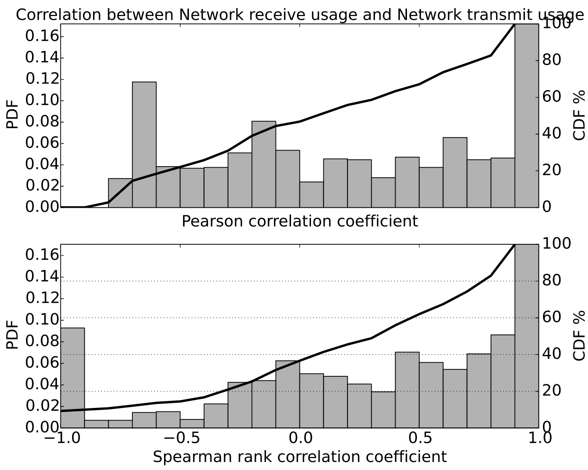

Figure 15: Correlation between network receive usage and network transmit usage: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

4.3 Correlation of CPU and Other Resource Usage

To get a better understanding of correlations between the usage of different resource types, we conduct a comprehensive analysis of all the possible pair-wise combinations (as throughout this work, the resource types considered are CPU, memory, disk read, disk write, network transmit, and network receive). We discuss in the following a selection of these results, depicted in Figures 10 to 15 (all similar in construction process and structure to Figure 9). We find low correlations between CPU usage and the usage of other resource types, and even lower correlation between disk and network resources. We also find that about 25% of the VMs in our study exhibit strong network transmit and network receive correlation, but either strongly positive or strongly *negative*; the remaining VMs exhibit the low correlation trend we have observed for other resources.

We start by investigating the correlation between the usage of CPU and of all the other resource types. We find that the correlation between CPU usage and disk write, disk read, network receive, and network transmit is very low. From both Figure 10 and Figure 13, we observe that the majority of the pair-wise correlations of the type CPU usage-other resource usage are between 0.0 and 0.5. These values are much lower than, for example, the correlation values between 0.8 and 0.9 found for in Section 4.1.

Next, we investigate potential correlations between storage and network resources. Figure 14 depicts the correlation between disk read and network transmit. The observed correlations are even lower than the what we observe for CPU usage, and the usage of disk and network resources. This observation holds for all other pair-wise correlations of disk and network usage.

Last, we investigate the correlation between the two network-related resource types, network receive and network transmit. Figure 15 shows the correlation between network transmit and network receive. From the figure, we observe that for the majority of the VMs the correlation between sending and receiving network traffic is very low. However, about 16% of the VMs have a strong *positive* correlation between sending and receiving network traffic, and about 8% of the VMs have a strong *negative* correlation between sending and receiving network traffic. We conclude that network receive and network transmit have more diverse pattern of correlations than other resources.

5 Characterization of Time-Patterns in Resource Usage

In this section, we analyze the time patterns of resource usages. Understanding the time patterns of the resource usages can help to build smart predictors that estimate upcoming resource usage, and can lead to improved datacenter efficiency. The main findings are:

1. The aggregate resource usage of the VMs fluctuates significantly over time.
2. The peak CPU resource usage is 10–100 times higher than the mean (Section 5.1).
3. CPU and memory resource usage can be predicted in short-term.
4. The usage of disk I/O and network I/O show daily patterns, for the `fastStorage` dataset. (Section 5.2).

5.1 Peak vs Mean Resource Usage

In this section, we analyze how dynamic are business-critical workloads, and contrast our findings with previously described workloads. To this end, following [23, 4] we study the peak and mean resource usage, and their ratio, over time. We report both hourly and daily intervals, for all the resources investigated in this work. (Previous studies report this value of intervals that range from 30 seconds [6] to 1 day [4], which makes it difficult to compare results across studies.) The results for CPU bandwidth, for disk read, and for amount of network-transmitted data are summarized in Figure 16 to 20. Overall, we find that business-critical workloads are much more dynamic than most previously described datacenter workloads, and more in line with the volatile grid workloads. This emphasizes the opportunity to design more efficient resource management approaches, such as the dynamic change of the number of active physical resources underlying the leased VMs.

We begin with a focus on CPU usage. Figure 16 shows the peak and mean CPU usage, and their peak-to-mean ratio, per hour and per day. CPU usage fluctuates significantly overtime. The daily peak usage can be 10 to 100 times higher than the daily mean usage. This phenomenon is commonly observed in other related workloads: in the Google trace (daily peak-to-mean ratio: 1.3), in the Microsoft Azure trace (15 minute sample, peak-to-mean ratio 1.7), and in the Microsoft Messenger trace (30 second sample, peak-to-mean ratio range from 2.5 to 6.0). The peak-to-mean ratios observed in business-critical workloads are even higher than the

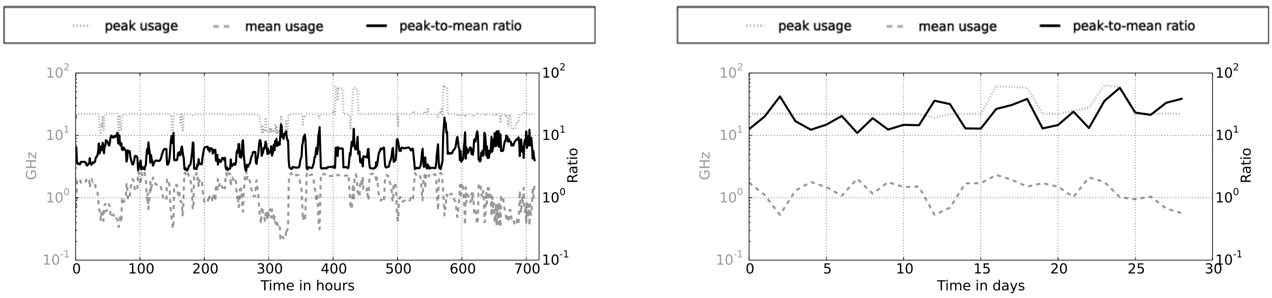

Figure 16: Peak to Mean resource usage, over time, CPU: (top) hourly data and (bottom) daily data.

ratios observed in these traces. Iosup et al. [23] analyze 5 grid traces and find hourly peak-to-mean ratios of up to 1,000:1. Similarly, Chen et al. [20] analyze 7 workload traces (from Facebook and Cloudera) and find peak-to-mean ratios ranging from 9:1 to 260:1. These ratios are more in line with the ratios we observe.

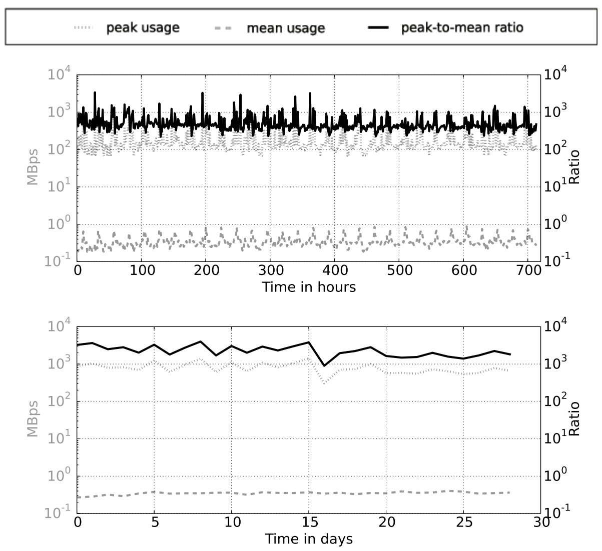

Figure 17: Peak to Mean resource usage, over time, Disk read: (top) hourly data and (bottom) daily data.

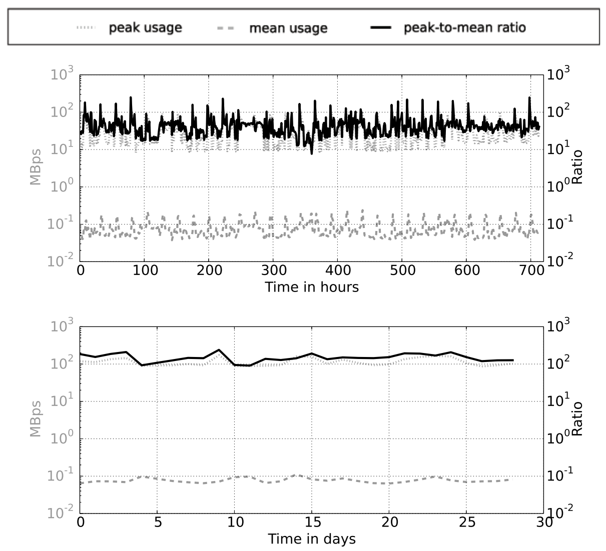

Figure 18: Peak to Mean resource usage, over time, Disk write: (top) hourly data and (bottom) daily data.

Similarly to CPU usage, we analyze the other resources, and find similarly high or even higher peak-to-mean ratios. Figure 17 shows the peak-to-mean ratio for disk-read usage. Both the hourly and daily ratios are much higher than the ratios observed for CPU usage: we observe 1:1000 and even 1:10,000 ratios. We find similar numbers for disk-write usage in Figure 18. Figure 20 depicts the peak-to-mean ratio for network-transmit usage. We find the ratios in the same order of magnitude as for disk usage (including the occasional 1:10,000). We find similar numbers for network-receive usage in Figure 19.

5.2 Time Patterns through Auto-correlation

In this section, we investigate the presence of time patterns in the usage of resources observed for business-critical workloads. To this end, we conduct an analysis using the auto-correlation (ACF) tool. For all resources,

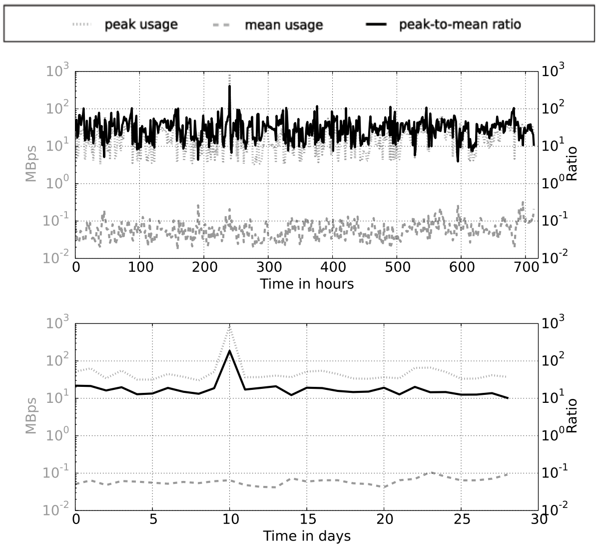

Figure 19: Peak to Mean resource usage, over time, Network receive: (top) hourly data and (bottom) daily data.

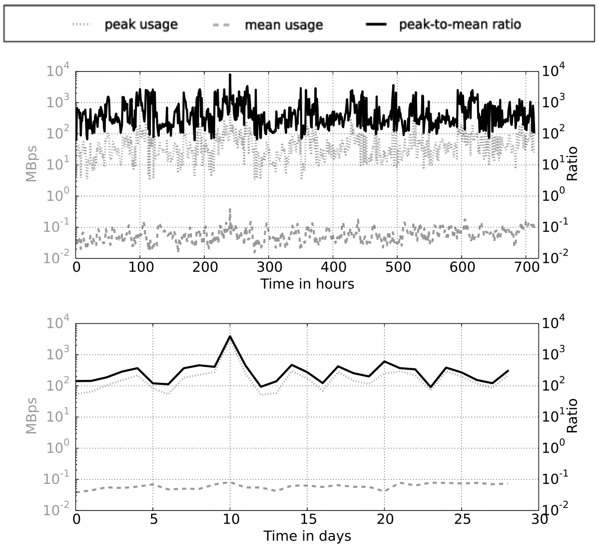

Figure 20: Peak to Mean resource usage, over time, Network transmit: (top) hourly data and (bottom) daily data.

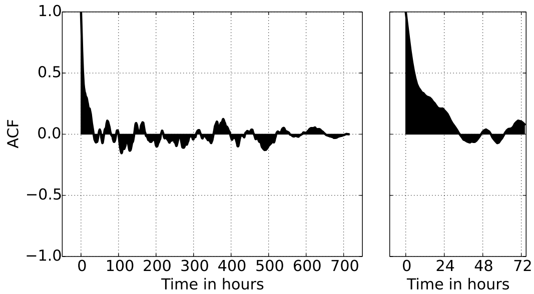

Figure 21: Auto-correlation of CPU usage.

we identify high ACF for small lag, which indicates predictable resource usage in the short term (that is, a few hours). We also find strong daily patterns in disk and, somewhat less, in network activities.

We analyze the ACF for all types of resource usage, for lag values from 0 hours up to 1 month, with 1-hour step. Figure 21 to 25 depicts the ACF values for all resource usage: CPU, disk read, and network transmit. The ACF values for the first 10 lags ranges, for all resource usage types, from 0.7 to 0.8, which is high and indicates strong auto-correlation. This indicates that, for all resource usage types, the resource usage is predictable in the short-term (up to a few hours).

For disk read (Figure 22), the ACF curve has local peaks at lag multiples that correspond to days; this indicates that the disk read has a strong daily pattern. We also observe that the disk write (not shown) and the network I/O (Figure 25) follow daily patterns, albeit less pronounced for the network I/O.

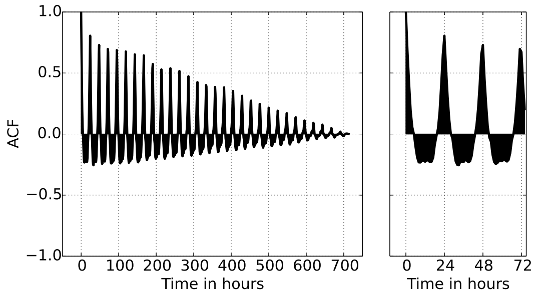

Figure 22: Auto-correlation of Disk read.

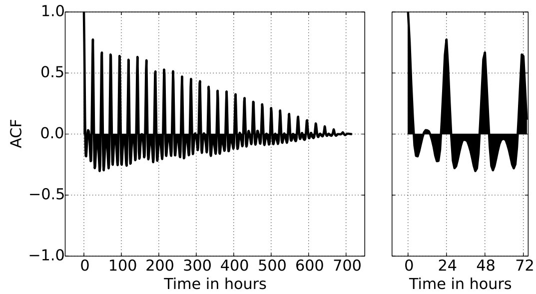

Figure 23: Auto-correlation of Disk write.

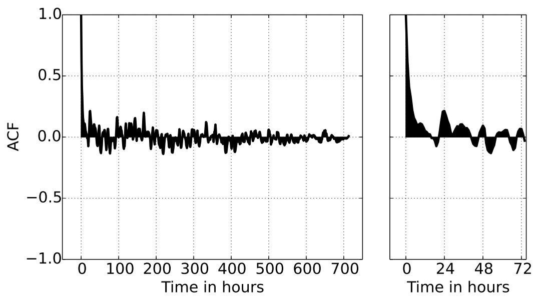

Figure 24: Auto-correlation of Network receive.

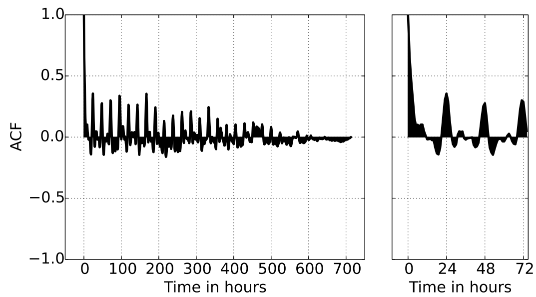

Figure 25: Auto-correlation of Network transmit.

6 Related work

In this section we present a comprehensive comparison between our and related work, along two axes: contributions related to public datasets and contributions related to workload characterization in datacenters.

Dataset release Our data release (see Section 9) complements well the few datasets that are publicly available. Many previous datacenter studies have used the workloads of distributed systems, from parallel [15] and grid [16] environments. The seminal Google workload dataset [4], released in 2011, includes only CPU and memory characteristics, and only *normalized*, rather than actual, values. The public SWIM workloads repository includes 5 workload traces, possibly extracted from publicly characterized Facebook MapReduce traces [20], but very short (only 1 day) and with no information about memory, network, or number of CPUs. Our main contribution here is the release of a dataset representative for a new type of workload, that is, business-critical jobs in cloud datacenters.

Workload characterization Table 1 summarizes the comparison of previous studies with our work. Overall, our study is derived from an averagely-sized dataset, but focuses on a different workload, and includes a more comprehensive resource view (four types of resources, including the rarely studied disk and network I/O). Our study also conducts a detailed study of both requested and used resources, something that most public datacenter-studies are lacking. We have already compared throughout this work, whenever possible, the results obtained in this work with results from previous studies.

Closest to our work, Reiss et al. [4] analyze the workloads of Google. They use a dataset that is limited in comparison to ours; notably, it does not cover disk and network I/O. Because the Google workloads do not match the profile of business-critical applications, we observe significantly different results. For example, in the Google trace, the actual workload is relatively stable, whereas our results indicate that CPU and memory workloads are very unpredictable for business-critical applications; we have indicated other differences throughout this work.

Also related to our work, Di et al. [10] analyze the workloads of Google (with the same dataset limitations as Reiss et al. [4]) and compare them with Grid/HPC systems regarding job length and host load, Mishra et al. [8] propose a workload classification approach and apply it to a four-days trace from a Google datacenter, and Gmach et al. [7] analyze workload demands in term of number of CPUs from an HP datacenter.

Other types of datacenter workloads are complemented by our study: Chen et al. [5] analyze MapReduce traces from Yahoo and Facebook regarding the input/output ratio, job count, job submission frequencies, etc.; Guenter et al. [6] analyze the workload traces from Microsoft's Live Messenger, Azure, and a shared computing cluster; and Chen et al. [17] analyze the workloads of login rates and connection counts in Microsoft's Live Messenger cluster.

7 Threats to validity

In this section we list possible threats to the validity of this work and the measures we took to mitigate them.

Timelessness of the dataset: As datasets age, they may become unrepresentative. Our dataset is not only more recent than that of previous work but, based on our experience, rather stable over time—business-critical applications have changed relatively little in the past few years. Thus, our datasets could still be relevant for the next 5-10 years.

Dataset size: Unrepresentative datasets can lead to misleading characterization. Compared to other workload traces surveyed in this work (see Table 1 and Section 6), but not necessarily public, our traces are of medium size, in both the period and the number of nodes they cover. Our traces are also of medium size in comparison with the public traces collected from parallel [15] and grid [16] environments. Thus, our results suffer from this set as much as results of studies derived from other traces in the field. Because this information is not publicly available, we can only argue that the datacenter size we considered in this work is more common in the industry as a whole than the Google, Facebook, and Microsoft datacenters.

Data collection tools: The data collection tool can cast doubts on the validity of the dataset. We rely on the tools provided by VMware, which are currently used by thousands of medium and large businesses, and thus can be considered a de-facto industry standard.

Trustworthy analysis: Mistakes in analysis occur often, in many fields of applied statistics. To alleviate this problem, in lack of a validation study conducted by a third-party laboratory, our statistical analysis is conducted by two of the authors, independently; the results have matched fully.

Collaboration with an industry partner: Analysis in which a participant has a vested interest could lead to biased results. To alleviate this problem, in lack of a multi-party industry consortium, we have collected and analyzed two traces. We note that the studies presented in Section 6 have the same limitation, but most rely on a single trace.

8 Conclusion and Ongoing Work

Understanding the workloads of cloud datacenters is important for many datacenter operations, from efficient capacity planning to resource management. In this work, we collect from a distributed datacenter hosting business-critical workloads 2 large-scale and long-term workload traces from 1,750 virtual machines. We analyze these traces for both requested resources and actual resource usage, in terms of CPU, memory, and disk I/O and network I/O; we also compare these findings with previous studies of workloads from search datacenters, parallel and grid environments, etc.

Our main findings, as reported in this technical report, are:

1. More than 60% of the VMs use less than 4 cores and 8 GB of memory.

2. There is a strong positive correlation between requested CPU and memory.

3. Resource usage is low, under 10% of the requested resources, and the correlation between requested and used resources is also low.
4. Peak workloads can be 10–10,000 times higher than mean workloads, depending on resource type.
5. The CPU and memory resource usage is often predictable over the short-term. Disk and network I/O follow daily patterns.

We are currently extending this work with more in-depth statistical and time-series analysis. We plan to use the findings to improve the datacenter-wide scheduler at Bitbrains.

9 Data Availability

The two traces will be made public during ICPE'15, as part of the archive Grid Workloads Archive [16].

References

[1] Schwiegelshohn, U,. Badia, Rosa M., Bubak , M. et al: Perspectives on grid computing. FGCS 2010 **26**(8) (October 2010) 5

[2] Talia, D.: Clouds for scalable big data analytics. IEEE Computer **46**(5) (2013) 5

[3] Olteanu, A.C., Iosup, A., Tapus, N.: Towards a workload model for online social applications: Icpe 2013 work-in-progress paper. In: ICPE. (2013) 319–322 5

[4] Reiss, C., Tumanov, A., Ganger, G.R., Katz, R.H., Kozuch, M.A.: Heterogeneity and dynamicity of clouds at scale: Google trace analysis. In: SoCC. (2012) 7 5, 6, 7, 8, 11, 14, 17, 20, 21

[5] Chen, Y., Ganapathi, A., Griffith, R., Katz, R.H.: The case for evaluating mapreduce performance using workload suites. In: MASCOTS. (2011) 390–399 5, 6, 21

[6] Guenter, B., Jain, N., Williams, C.: Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning. In: INFOCOM. (2011) 1332–1340 5, 6, 17, 21

[7] Gmach, D., Rolia, J., Cherkasova, L., Kemper, A.: Workload analysis and demand prediction of enterprise data center applications. IISWC 2007 5, 21

[8] Mishra, A.K., Hellerstein, J.L., Cirne, W., Das, C.R.: Towards characterizing cloud backend workloads: insights from google compute clusters. SIGMETRICS Performance Evaluation Review **37**(4) (2010) 34–41 5, 6, 21

[9] Iosup, A., Epema, D.H.J.: Grid computing workloads. IEEE Internet Computing **15**(2) (2011) 19–26 5

[10] Di, S., Kondo, D., Cirne, W.: Characterization and comparison of cloud versus grid workloads. In: CLUSTER. (2012) 230–238 5, 6, 7, 11, 21

[11] Deng, K., Song, J., Ren, K., Iosup, A.: Exploring portfolio scheduling for long-term execution of scientific workloads in iaas clouds. In: SC. (2013) 5

[12] Beaumont, O., Eyraud-Dubois, L., Larchevêque, H.: Reliable service allocation in clouds. In: IPDPS. (2013) 55–66 5

[13] Bautista-Gomez, L.A., Nicolae, B., Maruyama, N., Cappello, F., Matsuoka, S.: Scalable reed-solomon-based reliable local storage for hpc applications on iaas clouds. In: Euro-Par. (2012) 313–324 5

[14] Agmon Ben-Yehuda, O., Posener, E., Ben-Yehuda, M., Schuster, A., Mu'alem, A.: Ginseng: market-driven memory allocation. In: VEE. (2014) 41–52 5

[15] Feitelson, D.: Parallel Workloads Archive. http://www.cs.huji.ac.il/labs/parallel/workload/ 5, 8, 9, 20, 21

[16] Iosup, A., Li, H., Jan, M., Anoep, S., Dumitrescu, C., Wolters, L., Epema, D.H.J.: The grid workloads archive. FGCS **24**(7) (July 2008) 672–686 5, 8, 9, 20, 21, 22

[17] Chen, G., He, W., Liu, J., Nath, S., Rigas, L., Xiao, L., Zhao, F.: Energy-aware server provisioning and load dispatching for connection-intensive internet services. In: NSDI. (2008) 337–350 5, 21

[18] Bodík, P., Fox, A., Franklin, M.J., Jordan, M.I., Patterson, D.A.: Characterizing, modeling, and generating workload spikes for stateful services. In: SoCC. (2010) 241–252 5

[19] Shen, S., Deng, K., Iosup, A., Epema, D.H.J.: Scheduling jobs in the cloud using on-demand and reserved instances. In: Euro-Par. (2013) 242–254 5

[20] Chen, Y., Alspaugh, S., Katz, R.H.: Interactive analytical processing in big data systems: A cross-industry study of mapreduce workloads. PVLDB **5**(12) (2012) 1802–1813 6, 18, 20

[21] Ren, Z., Xu, X., Wan, J., Shi, W., Zhou, M.: Workload characterization on a production hadoop cluster: A case study on taobao. In: IISWC, IEEE Computer Society (2012) 3–13 6, 14

[22] Feitelson, D.: Packing schemes for gang scheduling. In: JSSPP. (1996) 89–110 10

[23] Iosup, A., Tannenbaum, T., Farrellee, M., Epema, D.H.J., Livny, M.: Inter-operating grids through delegated matchmaking. Scientific Programming **16**(2-3) (2008) 233–253 17, 18

10 Appendix

In this section, the results for the RND dataset are shown. In Section 10.1, the resource usages of the RND set are shown. We analyze the per VM usage and the variation of usage. In Section 10.2, we analyze the correlations between resource usages. In Section 10.3, we analyze the resource usages over time by looking at the peak-to-mean ratio. In Section 10.4, we analyze the time pattern of the resource usages. In summary, we find that the resource usage patterns of the RND are similar to the fastStorage set.

10.1 Resource usage, RND set

We analyze the resource usages in this section. For each resource usage, we first look at the resource usage per VM and then look at CoV of usage.

Figure 26 shows the CPU usage for the RND set. As is shown in Figure 26 (left), the large gap between the mean curve and the max curve indicates that the CPU usage is low most of the time. Figure 26 (right) shows the CDF of the CoV in the observed CPU usage per VM. About 50% of the CoV for CPU usage are lower than 1. And about 20% of the CoV are higher than 3. The CPU usage of these VMs is dynamic and unpredictable.

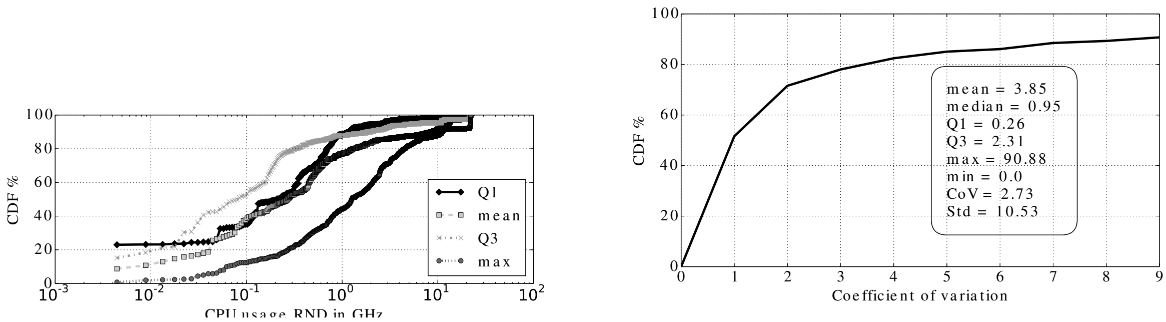

Figure 26: CPU usage RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

For the memory usage, as it is shown in Figure 27 (left), the memory usage is low: about 80% of the VMs whose average memory usage are lower than 1 GB. Figure 27 (right) shows the CDF of the CoV of memory usage per VM. The memory usage is stable: about 70% of the VMs' CoV of memory usage is lower than 1.

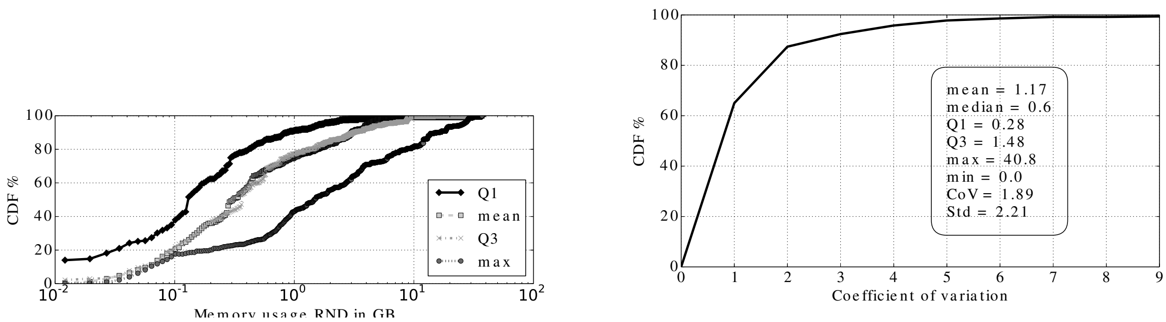

Figure 27: Memory usage RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

Figure 28 shows the CPU and memory usage over time, for the RND set. Similar to the fastStorage set, the CPU utilization is higher than the memory utilization.

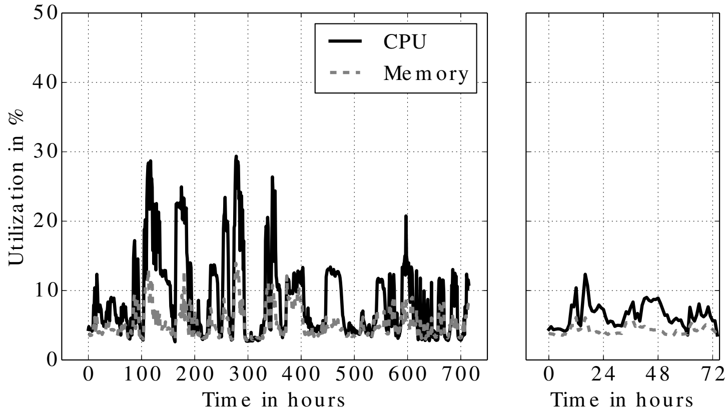

Figure 28: CPU and memory usage over time, RND set.

Figure 29 shows the disk read usage for the RND set. The VMs in RND set, read sporadically: more than 90% of the VMs perform three-quarters of their disk reads ("Q3" curve) at less than 1 MB/s. The CDF for CoV of disk reads is plotted in Figure 29 (right). The disk reads are bursty. The disk read usage is much more dynamic than the CPU usage.

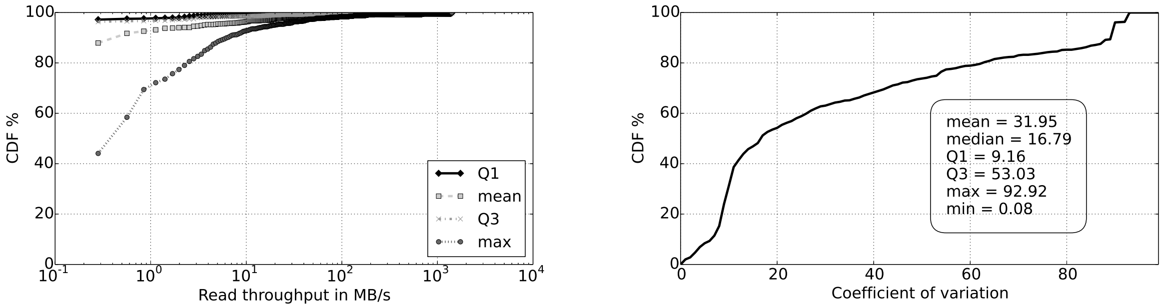

Figure 29: Disk read usage, RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

Figure 30 shows the disk write usage for the RND set. Most of the VMs do not write at all, and the disk write is less dynamic than then disk read.

For the network usages, as they are shown in Figure 31 and 32. The network usages of the RND set. The network usages are low and dynamic.

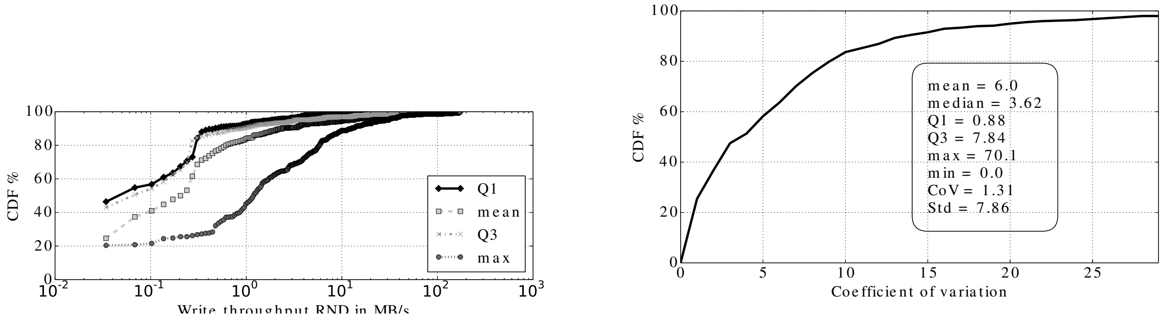

Figure 30: Disk write usage, RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

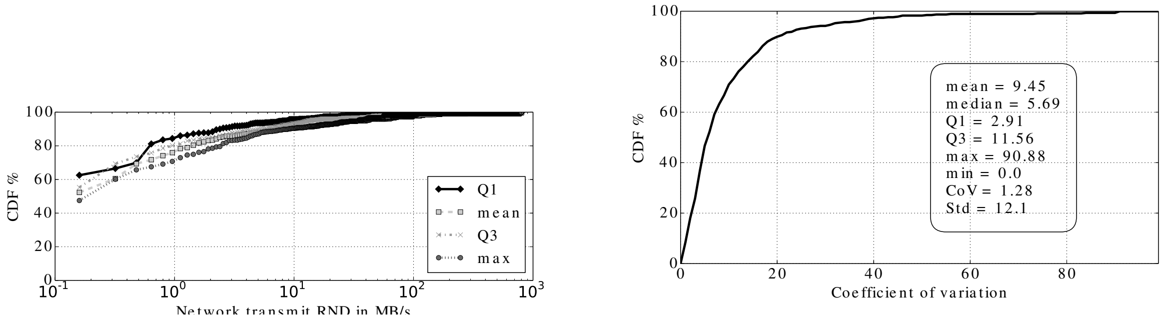

Figure 31: Network transmit usage RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

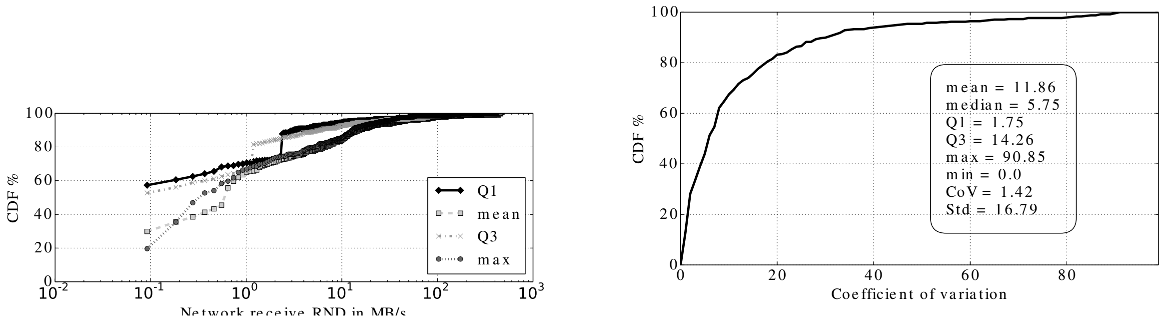

Figure 32: Network receive usage RND set: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

10.2 Correlation of Resource usage, RND set

In this section, to study the possible correlations between CPU usage and the usages of the other resources. We look at the PCC and SRCC between: CPU and memory, CPU and network receive, CPU and network transmit, CPU and disk read, CPU and disk write are shown. We find low correlations between CPU usage and the usage of other resource types, and even lower correlation between disk and network resources.

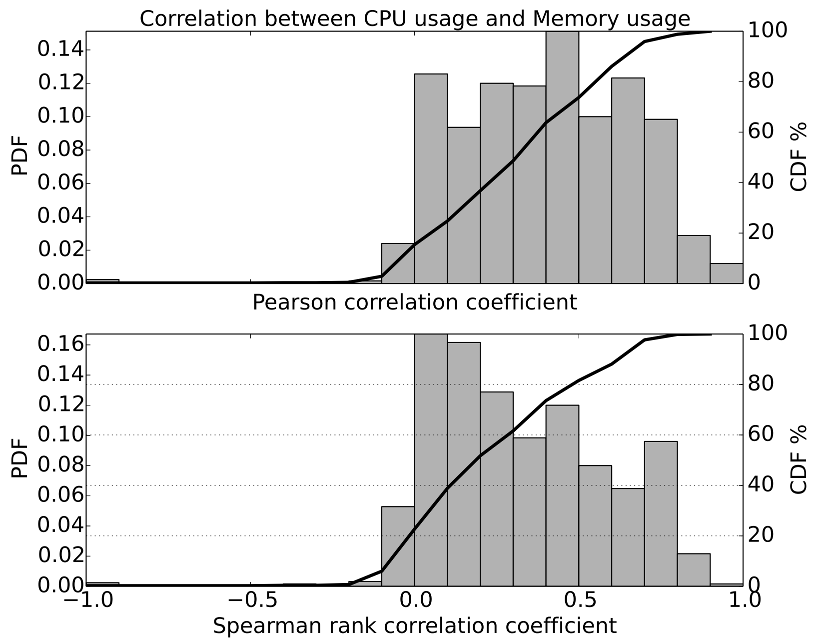

Figure 33: Correlation between CPU usage and memory usage, RND set: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

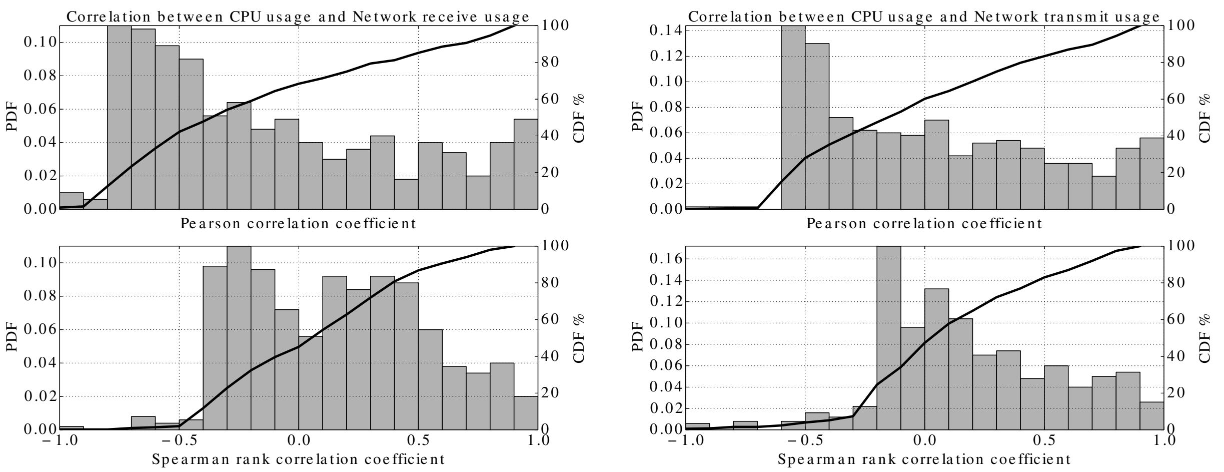

Figure 34: Correlation between CPU usage, RND set: (left) CDF and PDF of PCC over time, and (right) CDF and PDF of SRCC over time.

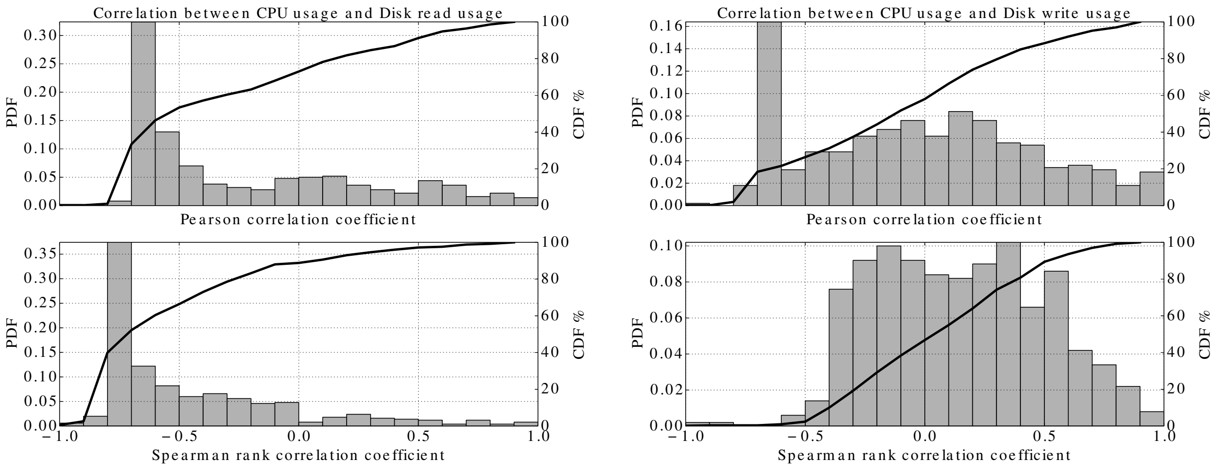

Figure 35: Correlation between CPU usage and (left two plots) disk read usage, and (right two plots) disk write usage. For each group of two plots: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

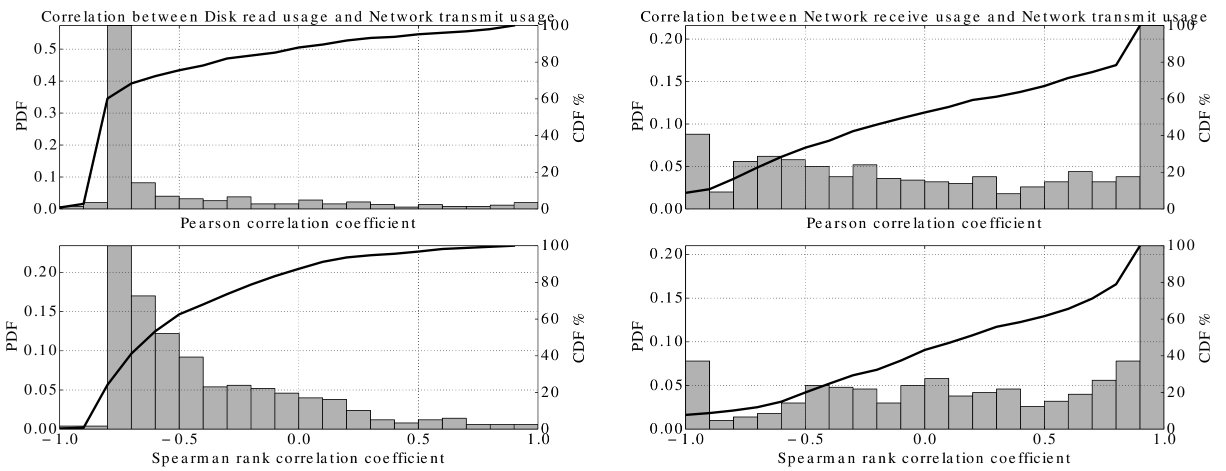

Figure 36: Correlation: (left) disk read usage and network transmit usage, and (right) network receive usage and network transmit usage, RND set: (top) CDF and PDF of PCC over time; (bottom) CDF and PDF of SRCC over time.

10.3 Peak to Mean ratio RND set

In this section, we analyze the resource usages over time. We look at the hourly and daily resource usages, and plot the ratio of peak and mean. Figure 37 to 41 show the resource usages overtime. We find that the resource usages fluctuate significantly over time.

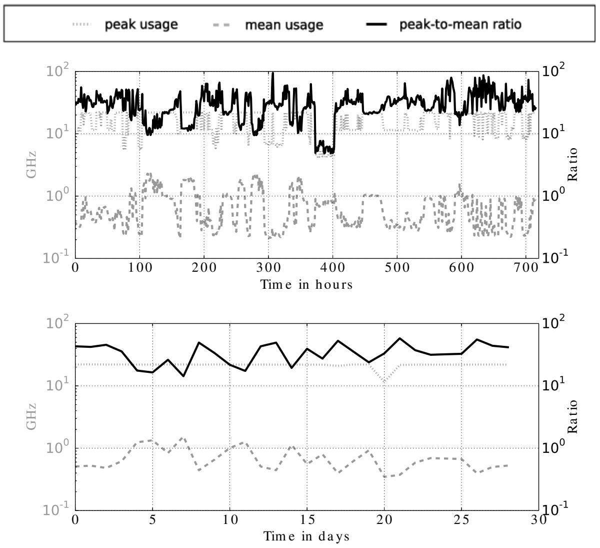

Figure 37: Peak to Mean CPU usage, over time, RND set: (top) hourly data; (bottom) daily data.

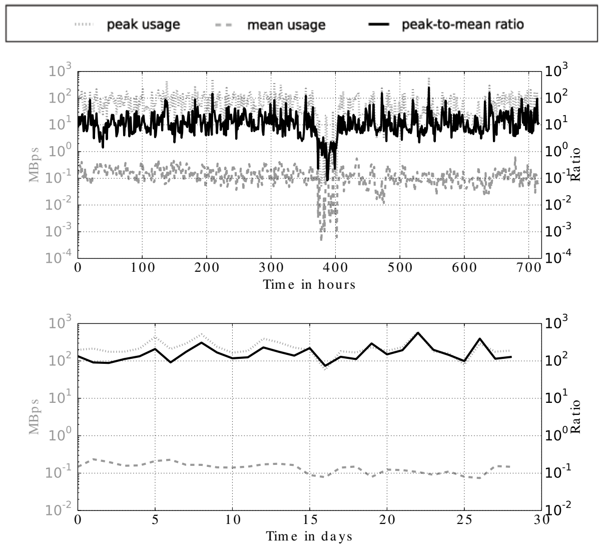

Figure 38: Peak to Mean Disk read usage, over time, RND set: (top) hourly data; (bottom) daily data.

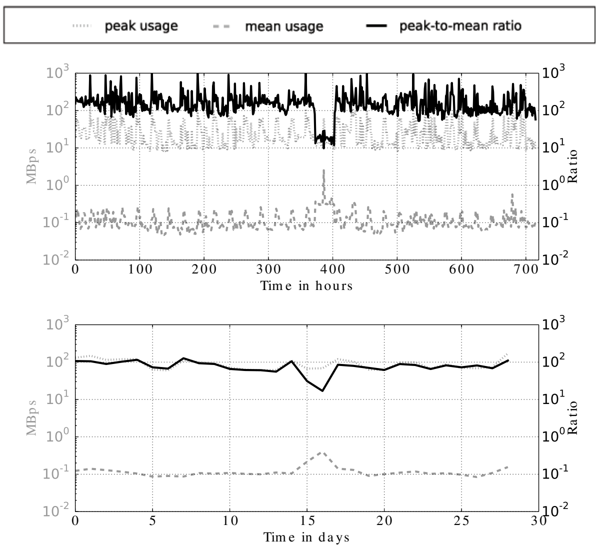

Figure 39: Peak to Mean Disk write usage, over time, RND set: (top) hourly data; (bottom) daily data.

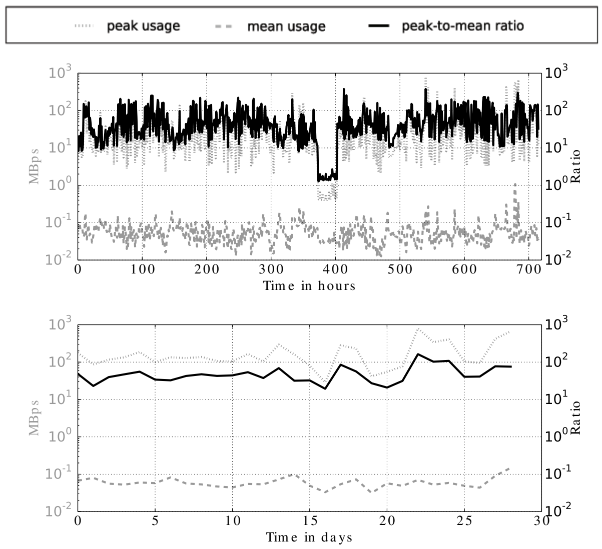

Figure 40: Peak to Mean Network transmit usage, over time, RND set: (top) hourly data; (bottom) daily data.

10.4 Autocorrelation RND set

In this section, we analyze the time pattern of the resource usages. Figure 42 to 46 show the auto-correlation function of the resource usages. We find that the resource usages are predictable in short-term, and the disk usages exhibit periodical patterns.

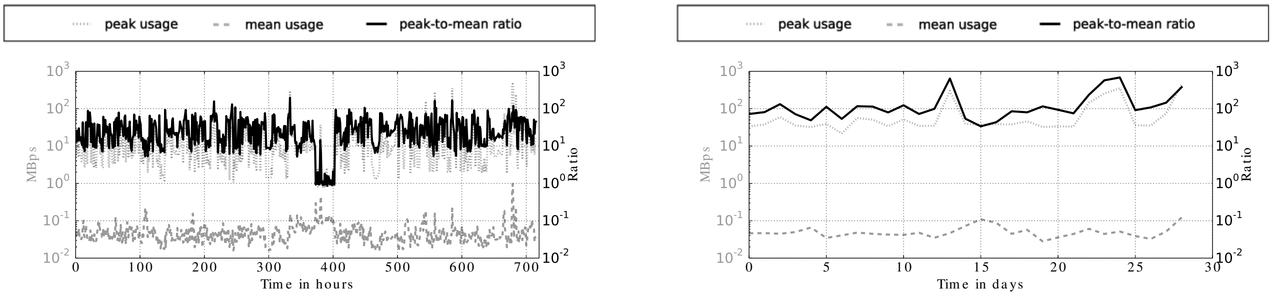

Figure 41: Peak to Mean Network receive usage, over time, RND set: (top) hourly data; (bottom) daily data.

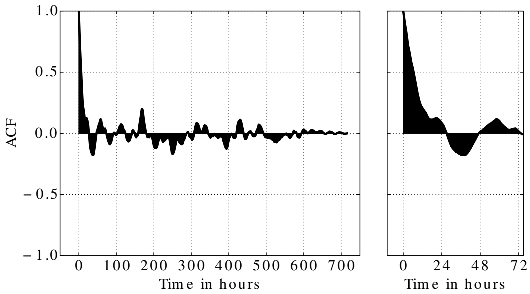

Figure 42: Auto-correlation, CPU, RND set.

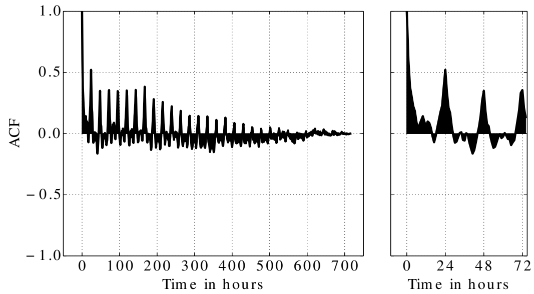

Figure 43: Auto-correlation, Disk read, RND set.

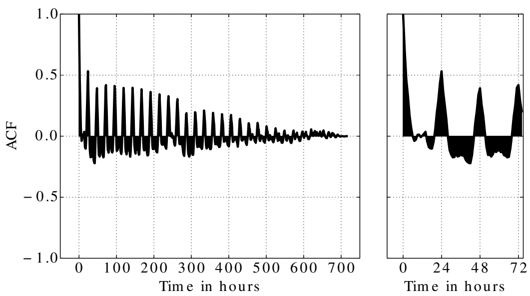

Figure 44: Auto-correlation, Disk write, RND set.

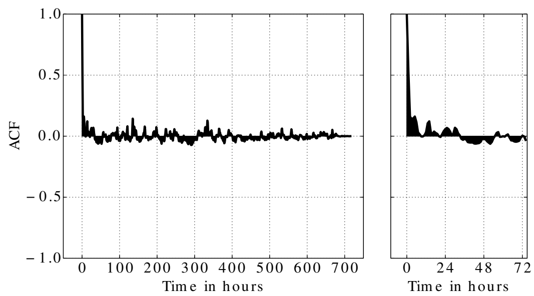

Figure 45: Auto-correlation, Network transmit, RND set.

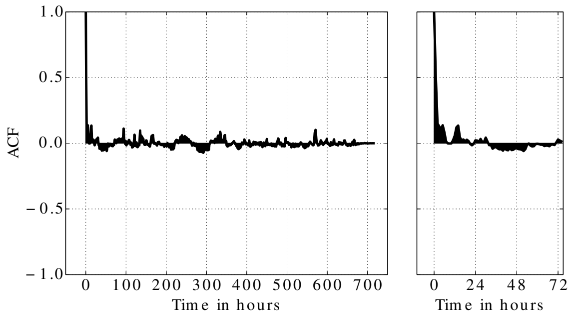

Figure 46: Auto-correlation, Network received, RND set.