

Experimental Analysis of Task-based Energy Consumption in Cloud Computing Systems

Feifei Chen, John Grundy, Yun Yang, Jean-Guy Schneider and Qiang He
Faculty of Information and Communication Technologies
Swinburne University of Technology
Melbourne, Australia 3122
{feifeichen, jgrundy, yyang, jschneider, qhe}@swin.edu.au

ABSTRACT

Cloud computing delivers IT solutions as a utility to users. One consequence of this model is that large cloud data centres consume large amounts of energy and produce significant carbon footprints. A common objective of cloud providers is to develop resource provisioning and management solutions that minimise energy consumption while guaranteeing Service Level Agreements (SLAs). In order to achieve this objective, a thorough understanding of energy consumption patterns in complex cloud systems is imperative. We have developed an energy consumption model for cloud computing systems. To operationalise this model, we have conducted extensive experiments to profile the energy consumption in cloud computing systems based on three types of tasks: computation-intensive, data-intensive and communication-intensive tasks. We collected fine-grained energy consumption and performance data with varying system configurations and workloads. Our experimental results show the correlation coefficients of energy consumption, system configuration and workload, as well as system performance in cloud systems. These results can be used for designing energy consumption monitors, and static or dynamic system-level energy consumption optimisation strategies for green cloud computing systems.

Categories and Subject Descriptors

C.4 [Computer System Organization]: Performance of Systems; K.4.1 [Public Policy Issues]: Use/abuse of power; B.8.2 [Performance Analysis and Design Aids]

General Terms

Measurement, Performance, Experimentation

Keywords

Cloud computing, green cloud, energy consumption, performance analysis, energy efficiency.

1. INTRODUCTION

Cloud computing is a new and promising computing paradigm which delivers computing as a utility [1]. It provides rented services for computation, application software, and data storage

via the Internet. Key advantages for consumers include flexible scaling on demand to their computing and data storage needs without the traditional and potentially large upfront investment in computing infrastructure. Over the last few years many large-scale data centres have been built due to the massive growth in demand for high performance cloud data and computational services. As cloud computing becomes more widespread, increasing data storage and computation needs raise the energy consumed by large cloud infrastructures. Thus, energy consumption has become a critical concern in designing modern cloud systems.

Firstly, a common economic objective of cloud providers is to minimise their total operational costs. High energy consumption directly contributes to operational costs, especially as energy unit costs continue to rise significantly. A utility provider in Virginia estimates that in 2012, 10% of all the energy it supplies to northern Virginia will be consumed by data centres [2]. Power consumption currently contributes up to 42% of a data centre's monthly expenses [3]. Secondly, due to the need to respond on-demand to customer load, many data centres consume electricity produced by "brown" generation facilities, such as coal. Thirdly, cloud system performance must not be jeopardised while minimising cloud system energy consumption. Therefore, energy consumption, as well as its impact on cloud system performance, operational cost and the environment, have become critical issues in green cloud computing systems [4].

Many efforts have been made to improve the energy efficiency of cloud systems. Some simple techniques provide basic energy management for servers in cloud systems, for example, turning on and off servers, putting them to sleep or using Dynamic Voltage/Frequency Scaling (DVFS) [5] to adjust servers' power states. DVFS adjusts CPU power (consequently the performance level) according to the workload. However, the scope of DVFS optimisation is limited to CPUs. Another approach for improving energy efficiency is to adopt virtualisation techniques to obtain better resource isolation and reduce infrastructure energy consumption through resource consolidation and live migration [6]. Based on virtualisation techniques, several energy-aware resource allocation policies and scheduling algorithms have been proposed to optimise energy consumption in cloud systems [7]. However, energy consumption and performance of cloud systems vary drastically with different system configuration and resource allocation strategies, as well as the workloads and types of runtime tasks in cloud systems [8]. By nature, the workloads of cloud systems are highly variable, application-specific and can often not be predicted in advance. Ideally, system performance should not be adversely impacted while energy consumption is being minimised. We thus require a thorough understanding of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICPE'13, April 21–24, 2013, Prague, Czech Republic.

Copyright © 2013 ACM 978-1-4503-1636-1/13/04...\$15.00.

energy consumption patterns of different types of tasks in cloud systems, for example, data retrieval and data processing tasks. We also need to understand how energy consumption of such tasks and cloud system performance are affected by different workloads and system configurations, respectively.

We proposed an energy consumption model to calculate the energy consumption of specific types of tasks in cloud systems, and to use this model to statically and/or dynamically organise cloud application load [9]. In our model, runtime cloud tasks are divided into three types: computation-intensive, data-intensive and communication intensive. The model identifies what factors determine the energy consumption of a specific task. However, the following major issues have yet to be addressed:

- How do we characterise and profile the energy consumption of different types of tasks?
- How do system configuration and resource allocation strategies affect the energy consumption?
- What is the relationship between energy consumption and workload of tasks?
- What is the relationship between energy consumption and cloud system performance?

In order to address these issues, we have conducted a series of experiments to empirically analyse the energy consumption in cloud systems. Based on our experimental results, we have identified the correlation coefficients of energy consumption, system configuration, workload, and system performance in cloud systems. Our analytical results provide guidelines for deriving energy consumption models based on different types of runtime cloud computing tasks. In addition, our analytical results also provide guidelines for statically planning tasks and scheduling on available cloud platforms, and dynamically monitoring energy consumption. They may be used to support system- or application-level energy consumption and performance optimisation (or both).

We briefly summarise the state-of-the-art of energy consumption models, energy-saving policies and analysis approaches in Section 2. In Section 3, we describe our energy consumption model to profile energy consumption. Our experimental energy consumption profiling setup and method are described in Section 4. In Section 5, we present our experimental profiling results and detailed analysis. We discuss observations derived from our analysis in Section 6. Finally, we summarise key findings and discuss directions for future work in Section 7.

2. RELATED WORK

Energy consumption of cloud systems has become a popular research topic in recent years. Several efforts have been made to build energy consumption models for cloud systems. Jung et al. [10] focused on the energy consumed by physical servers. Their energy consumption model does not take into account the impact of specific workloads running in cloud systems. Mach and Schikuta [11] proposed a method for energy consumption calculation based on the number of Java Virtual Machine (JVM) instances. However, it is difficult to measure the actual numbers of JVMs because of the dynamic nature of the JVM life cycle. Lee and Zomaya [12] proposed an energy model of cloud tasks for developing energy-conscious task consolidation algorithms that reduce energy consumption of cloud systems. However, in their model it is assumed that the relation between CPU utilisation and energy consumption is linear. This is a huge limitation

because the utilisation of other resources such as memory and hard disk can also greatly influence the energy consumption.

Energy-saving policies of cloud systems have also been investigated in the past few years. Liu et al. [13] described a new cloud infrastructure which can dynamically consolidate Virtual Machines (VMs) based on CPU utilisation of servers in order to identify idle physical servers. Identified idle physical servers can be turned off to save energy. Verma et al. [14] used the characteristics of VMs, such as cache footprint and the set of applications running on the VMs, to drive power-aware placement of VMs. VirtualPower [15] was proposed to exploit power management decisions of guest VMs on virtual power states. The virtual power states of guest VMs were considered as preconditions to run local and global energy management policies across the computation. These energy saving policies do not take the workload in cloud systems into consideration.

Research efforts have also focused on profiling and analysing the energy consumption of cloud systems. Most existing profiling efforts were conducted by using energy benchmarks or closely monitoring the energy profiles of individual system components at runtime, such as CPU, cache, hard disk and memory. A framework has been proposed by Stoess et al [16] for energy optimisation and development of energy-aware operation systems based on the availability of energy models for each hardware component. Chen et al. [17] developed a linear power model that presents the behaviour and power consumption of individual hardware components of a single physical server. Joulemeter, a power meter for VMs [18], makes use of software components to monitor the resource usage of VMs and then converts the resource usage to energy consumption based on the power model of each individual hardware component. Although some of the profiling and analysis are conducted based on specific applications in cloud systems, the evaluation only includes a single type of cloud application. For instance, Lefèvre and Orgerie [19] evaluated energy efficient cloud systems on a multicore platform. They focused only on CPU cores and conducted their evaluation of the energy consumption during migration of VMs only with computation-intensive cloud applications.

Some of the above work have made initial efforts in profiling the energy consumption and performance of cloud systems. However, none of them has identified the relationship between energy consumption and runtime tasks under different cloud system configurations and the correlation with cloud system performance. In order to address these issues, in this paper we present our energy consumption profiling results obtained from extensive experiments conducted on a cloud test-bed. Our goal is to investigate and characterise the impact of both system configurations and workload on the energy consumption and system performance.

3. ENERGY CONSUMPTION MODEL

Our energy consumption profiling and analysis is motivated by our existing energy consumption model proposed for cloud-wide energy analysis [9]. It provides a basis for characterising the energy consumption in cloud systems under different system configurations.

In this model, the total energy consumption of a cloud workload is divided into a fixed part and a dynamic part. The fixed part of energy consumption includes the energy consumption during idle time and energy consumption of the cooling system, defined as

EC_{Fix} . The dynamic energy consumption is the energy consumed by running tasks in the cloud system, defined as EC_{Var} . Thus, the total energy consumption defined as EC_{Total} is formulated as:

$$EC_{Total} = EC_{Fix} + EC_{Var} \quad (1)$$

Instead of individual hardware components, we set single task running in a cloud system as the fundamental unit for energy consumption calculation. All runtime tasks in cloud systems utilise computation, storage and communication resources. However, the percentage of each resource used by different types of tasks is significantly different. Based on the major type of resource consumed by the task, we characterise all cloud runtime tasks three types: *computation-intensive*, *data-intensive* and *communication-intensive*. For a computation-intensive task t_{comp}^i , $1 \leq i \leq n$, where n is the total number of computation-intensive tasks, the energy consumption of t_{comp}^i is defined as $EC(t_{comp}^i)$. For a data-intensive task t_{data}^i , $1 \leq i \leq m$, where m is the total number of data-intensive tasks, the energy consumption of t_{data}^i is defined as $EC(t_{data}^i)$. For a communication-intensive task t_{com}^i , $1 \leq i \leq k$, where k is the total number of communication-intensive tasks, the energy consumption of t_{com}^i is defined as $EC(t_{com}^i)$. Due to scheduling and other overheads, which is denoted as EC_{sche} , the total energy consumption of two tasks is usually not the sum of the energy individually consumed by the tasks. In our model, the total energy consumption is defined as:

$$EC_{Total} = EC_{Fix} + \sum_{i=1}^n EC(t_{comp}^i) + \sum_{i=1}^m EC(t_{data}^i) + \sum_{i=1}^k EC(t_{com}^i) + EC_{sche} \quad (2)$$

For each task, its energy consumption is dependent on the task workload. The factors related to workload influence energy consumption of the task directly and largely. In our model, the parameters of a task taken into account in the calculation of the energy consumption include: number of processes for the task, defined as PT ; the size of data to be processed, defined as DS ; and the size of data to be transmitted, defined as DT . In addition, system configurations, such as hardware of the physical server and the scale of the configured VMs, have significant impact on energy consumption. Hence, the energy consumed by each task is determined by the task parameters and the system configuration, denoted as C . Thus, the energy consumption of each type of task are defined as:

$$EC(t_{comp}^i) = f_{comp}(PT_{comp}^i, DS_{comp}^i, DT_{comp}^i, C_{comp}^i) \quad (3)$$

$$EC(t_{data}^i) = f_{data}(PT_{data}^i, DS_{data}^i, DT_{data}^i, C_{data}^i) \quad (4)$$

$$EC(t_{com}^i) = f_{com}(PT_{com}^i, DS_{com}^i, DT_{com}^i, C_{com}^i) \quad (5)$$

In order to determine the energy consumption model of a cloud system, it is essential to specify the factors that have impact on the energy consumption of each type of tasks.

4. EXPERIMENTAL SETUP

Our energy consumption profiling and analysis were performed based using the energy consumption model described in Section 3. Based on the energy consumption profiling results, the impact of the factors indicated in equations (3)-(5) on the energy consumption of different tasks will be verified. Our objective was to collect data and analyse the correlation coefficients of energy consumption, system configuration, workload, and system

performance. The analysis result will be adopted to refine and formalise our energy consumption model, used to support predictive analysis of workload energy consumption, and used to support static and dynamic optimisation of energy usage.

We profiled the energy consumption by creating heterogeneous Virtual Machines in a cloud system and running computation-intensive tasks, data-intensive tasks and communication-intensive tasks, respectively. For each type of task, we also profiled the energy consumption and system performance with various task parameters, that is, the number of processes, the size of data to be processed, the size of data to be transmitted, and task workload. We introduce the experimental setup in this section.

4.1 Testbed

SwinCloud is a private cloud that provides a common computational infrastructure to researchers at Swinburne University of Technology. Our experiments were conducted in SwinCloud as we have detailed knowledge of the cloud hardware, networking, operating system versions, and other application software. SwinCloud was experimented in the Energy Research Lab (ERL) at Swinburne University of Technology. This lab focuses on energy-related research and development. Using the extensive and sensitive power monitoring facilities provided by this lab, we could precisely monitor the power consumption of the SwinCloud server and network devices, including network cards, switches and routers. The power consumption measurement was realised and managed using PowerNode, a power usage profiling equipment developed by GreenWave Reality¹. It supports measurement of both immediate and average power consumption. Collected power readings were reported to the GreenWave Gateway, which is used for creating a mesh-based Home Area Network (HAN). The GreenWave Gateway then sent the data to the GreenWave Reality data centre, where the information could be viewed and analysed.

Table 1. Specifications of HP Z400

Basic Specification		Notes
Number of Cores	4	
Number of Threads	2	Intel Hyper-Threading Technology
CPU Frequency	2.8GHz	Fixed CPU Frequency
Memory	10GB	Memory Speed 1333 MHz
Hard Drive	1TB 7200 RTM SATA	
Network Interface	Intel e1000 Gb	

A cloud computing system is composed of multiple servers. The energy consumption of the cloud system includes the energy consumed by individual servers and the scheduling and communication overhead across different servers. In this paper, we focus on the energy consumption of individual servers as it is the predominant part [20]. Furthermore, the cross-server scheduling and communication overhead of one cloud system can be significantly different from the other, depending on the scheduling mechanism adopted by the cloud systems and the distribution of the constituent servers. Thus, we conducted our experiments by measuring energy consumption of tasks running

¹ <http://www.greenwavereality.com/>

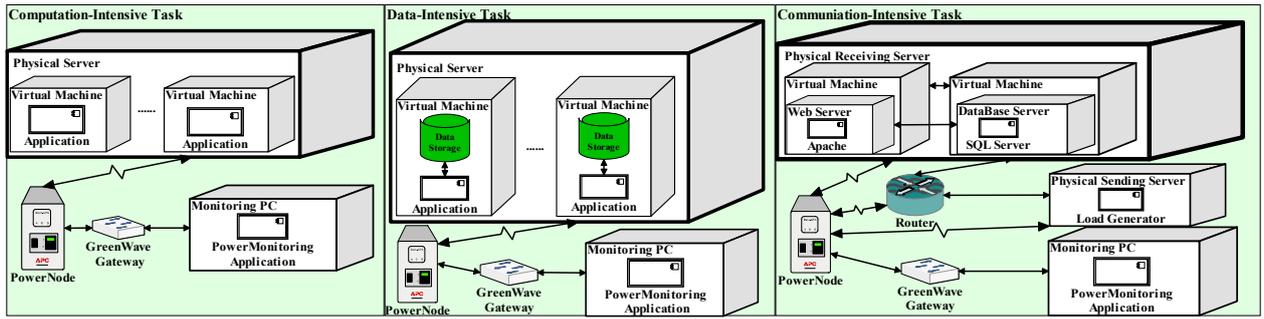


Figure 1. Energy profiling Framework

on a single discrete server. We left research to determine the energy consumption incurred by cross-server scheduling for future work. Moreover, we did not adopt any existing energy-saving policies in our experiments such as DVFS [5] or VirtualPower [15], in order to be able to isolate the factors causing unexpected energy consumption. Our experimental set up can however be reused to include these options enabled.

A type of the servers currently deployed in SwinCloud is HP Z400. Table 1 lists the specifications of the HP Z400. The Virtual Machine Manager (VMM) utilised for VM management is VMware ESX 4.1 and the operating systems running on the virtual machines are either Window Server 2008 or Windows XP Professional.

Figure 1 presents the energy profiling framework for different tasks used in our experiments. For computation-intensive tasks and data-intensive tasks, stand-alone applications were deployed. For communication-intensive tasks, a Web Application was deployed. The user requests were sent to the Web server through the router from a client PC. The major energy consumption of computation-intensive tasks was introduced by the computation of the deployed application. For data-intensive tasks and communication-intensive tasks, the energy consumption of the deployed application caused by computation is minimal. The major energy consumption of data-intensive tasks is introduced by reading/writing data in data storage. For communication-intensive tasks, the major energy consumption includes the energy consumed to send user request, the energy consumed to serialise/deserialise data and the energy consumed to process user requests. In order to focus on the communication aspects, the workload applied for communication-intensive tasks in our experiments was arranged to minimize load on sender/receiver so almost all energy is the encoding and decoding overhead.

Table 2. Type of VM

Virtual Machine	Number of Cores	RAM	Hard Disk
Small	1	2GB	80GB
Medium	2	4GB	80GB
Large	3	6GB	80GB
XLarge	4	8GB	80GB

In the experiments, all VMs were assigned with 2GB, 4GB, 6GB or 8GB memory. The number of virtual CPUs (vCPUs) of each VM varied from 1 to 4 in steps of 1. The number of vCPUs corresponded to the number of physical cores assigned to the VM. The configuration of VMs in different scales is presented in Table 2. System configurations related to specific task types will be introduced in Sections 5.1, 5.2 and 5.3, respectively.

4.2 Profiling Method

The energy consumption result of a task equals the difference of average power consumption between the server with and without workload multiplied by the execution time of the task. We firstly retrieved the average power consumption measured by PowerNode with no hosted application workload in the cloud system as our idle state benchmark. Then, we retrieved the average power consumption measured by PowerNode during the running time of each task. Finally, we multiplied the average power consumption by the execution time of the task to obtain the energy consumption of the task.

As discussed in Section 3, the task parameters, including the number of processes, the size of data to be processed, the size of data to be transmitted, have impact on the task energy consumption and system performance based on the energy consumption model. These task parameters determine the workload of a single task. In addition, the number of running tasks determines the overall workload in the cloud system. Energy consumption is highly influenced by the workloads in the cloud system as higher workloads consume more resources that leads to higher energy consumption. Moreover, system configurations, such as the number and scale of the configured VMs, also impact energy consumption significantly. As such, we took the cloud system workload and system configuration as inputs of our experiments. We set energy consumption and system performance as outputs of our experiments. We selected the throughput of the system as the profiled system performance measure. This is because throughput is often a key performance parameter monitored in cloud systems. It has the advantage of reflecting resource usage accurately [21]. For computation-intensive tasks, throughput is defined as the number of computation tasks completed per hour. For data-intensive tasks, throughput is defined as the amount of data transferred per second. For communication-intensive tasks, throughput is defined as the total number of user interactions requested and completed successfully per second.

4.3 Test Case Design

Again, we used three types of tasks: computation-intensive, data-intensive and communication-intensive tasks. We designed and conducted three series of experiments, described as follows:

1. **Computation-intensive tasks:** The major resources consumed by computation-intensive tasks are the CPU cores. In order to make sure the computation workload would be distributed to all available CPU cores, an application which calculates Fibonacci sequence has been developed and parallelism has been applied to the application. We deployed multiple

processes to calculate Fibonacci sequences to implement computation-intensive tasks. Each calculation was considered as a computation-intensive task. As the largest number of the Fibonacci sequence determines the duration of each calculation task, we mapped this number to the workload of each computation-intensive task – defined as LN . As this application purely consumes CPU resources, the energy consumption caused by other resources such as memory can be eliminated. In order to analyse the impact of workload and cloud system configurations on the total energy consumption and throughput, we designed three sets of test cases in this series of experiments. We first ran a single task to calculate Fibonacci sequence and increased the LN of the task gradually with fixed system configuration in test set 5.1.1 as described in Section 5.1. Then, we increased the resource allocated to the task while keeping LN constant. Computation-intensive tasks mainly consume CPU resources and Hyper-Threading (HT) technology is usually applied to improve the overall performance of the CPU. Thus, we measured the energy consumption and system performance with HT enabled and disabled in test set 5.1.2 as described in Section 5.1, to analyse the impact of HT technology on energy consumption and system performance. In these tests we kept the resources allocated constant and increased the LN of the tasks gradually and turned on and off HT. In test set 5.1.3 as described in Section 5.1, we ran multiple tasks with the same LN simultaneously and kept the system configuration and resource allocation constant to analyse the scheduling overhead.

2. Data-intensive tasks: As a data-intensive task mainly consumes the storage resources in cloud systems, we ran IOzone benchmark² in our tests, a disk and file system benchmarking application. Using IOzone, we generated a large number of I/O operations and stressed the disk I/O as required. Each run of IOzone benchmark was considered as a data-intensive task. The total amount of data read/writer determined the workload of a task. In each task, we set the read and write ratio to 50% - 50%. The parameters of a data-intensive task to be changed include the process number and the size of the data record. We set the size of the data record to be transferred at 4K, 8K and 64K bytes as our basic test suite. 4K is the memory page size, 8K is what Microsoft Windows uses for network transfers, and 64K is the typical record size that Windows uses when applications try to transfer blocks of data bigger than 64K [22]. We also tested a few very large data records, including 8MB and 16MB data records. A total of five sets of tests were designed and run in this series of experiments. Test set 5.2.1 as described in Section 5.2 aimed to reveal the impact of workload on energy consumption and system throughput while keeping the task parameters and system configuration constant. The purpose of test set 5.2.2 as described in Section 5.2 was to analyse the impact of system configuration on energy consumption and system throughput. So we kept workload and task parameters constant and changed the number and type of virtual machines. In test set 5.2.3 as described in Section 5.2, we aimed to evaluate the impact of task parameters on energy consumption and system throughput by keeping workload and system configuration constant. Furthermore, we evaluated the impact of scheduling overhead on energy consumption and system throughput in test set 5.2.4 as described in Section 5.2. Similar to computation-intensive tasks, resource allocated to data-intensive tasks would affect the energy

consumption and system performance. Therefore, we measured the energy consumption and system throughput by changing the total resource allocation in test set 5.2.5 as described in Section 5.2.

3. Communication-intensive tasks: We deployed a Java Web Application named JPetstore³ in SwinCloud as the communication-intensive application. It is a Web e-commerce application which has been used as a representative application for a transactional workload that stresses the servers and network devices as required in our tests. JPetstore uses a Web server to handle the user requests and a database server to process the database queries in response to the user requests. Therefore, it is a composite task which is composed of communication-intensive task, data-intensive task and a small level of computation-intensive task. For our experiment, we used its communication aspects only. As mentioned in Section 4.1, we arranged the workload of communication-intensive task to minimize load on sender/receiver in order to measure the energy consumption of the communication aspects only of this reference application. All network traffic generated by customers was emulated using JMeter⁴. JMeter is a load testing tool for analysing and measuring the performance of variety of Web applications. Each user session generated by JMeter consisted of a series of sequential user requests, such as browsing the list of products, adding items to the shopping cart, checking out, and so on. Each run of JMeter was considered as a communication-intensive task and the number of user sessions generated by JMeter determined the workload of the task. All user sessions were started with fixed time interval in each test set. The time interval decreased when the number of user sessions increased. The parameters of a communication-intensive task to be changed included the user request type, the number of requests in each user session and the packet size of each user request. We developed three test sets for communication-intensive tasks. We applied different workloads for each test set by mixing browsing requests or shopping requests in different percentages. In test set 5.3.1 as described in Section 5.3, we fixed the task parameter and the workload and changed virtual machine types. In test set 5.3.2 as described in Section 5.3, we changed the workload of the task while fixing the task parameter, the system configuration and the resource allocation. We only changed the task parameter and kept all other factors constant to evaluate the impact of task parameters on energy consumption and system performance in test set 5.3.3 as described in Section 5.3.

5. EXPERIMENTAL RESULTS

We conducted three major sets of tests to analyse the energy consumption of three types of cloud-hosted application tasks in order to analyse the relationship between the system energy consumption and system performance. For each test set, we took cloud system workload and system configuration as inputs. Energy consumption of each task and the system throughput were the outputs of our experiments. We modified the configuration of SwinCloud, including the number of VMs, the hardware and software resources allocated to the VMs. We then measured the energy consumption of SwinCloud running different types of tasks. Each set of tests was repeated ten times to reduce measurement error. We analysed the impact of system

² <http://www.iozone.org>

³ <http://java.sun.com/developer/releases/petstore/>

⁴ <http://jmeter.apache.org/>

configuration, task parameters and workload on the energy consumption and performance of the cloud system based on the experimental results.

5.1 Computation-intensive Tasks Results

A computation-intensive task usually requires a number of isolated processes to perform the computation in a cloud system. The total energy consumption might increase with the number of processes since the increased overhead of scheduling will cause extra energy consumption. Moreover, the energy consumption of the cloud system is subject to change under different system configurations.

We deployed multiple processes that calculate Fibonacci sequences to implement computation-intensive tasks. Parallelism was applied to make sure the computation workload would be distributed to all available vCPUs. As the largest number of the Fibonacci sequence determines the duration of each calculation task, we mapped this number to the workload of each computation-intensive task – defined as LN (see Section 4.3). In order to control the execution time of each task within a reasonable value, we set LN from 52 to 56 in the following tests.

Test Set 5.1.1: Keeping the number of tasks constant, while gradually increasing the number of active CPU cores allocated to the task, and the workload of the task. The total number of tasks was set to one. This set of tests was run on an XLarge virtual machine (see Table 2 for specification details). The server power usage is presented in Figure 2. We observed increasing power consumption caused by increasing CPU usage. The power consumption was linear to CPU usage. The energy consumption per task is displayed in Figure 3. We noticed that the energy consumption of the task was impacted by the number of CPU cores allocated to the task. Moreover, the largest LN of the Fibonacci sequence also affected the energy consumption. As shown in Figure 3, the energy consumption of each task increased with the workload of the task. Moreover, the energy consumption of each task decreased dramatically as the number of cores allocated to the task increased. This is because the execution time of a task will decrease as more computation resources are allocated to the task. However, the increase in average energy usage rate caused by extra CPU cores is not as much as the execution time of the task. Therefore, the energy consumption will decrease accordingly. In addition, we observed a slight turning point of the energy consumption when the number of cores allocated to the task reaches three. For instance, when we set the largest number of the Fibonacci sequence LN to 56, the energy consumption with four cores allocated increased 4% compared to energy consumption with three cores. This shows that the overhead of scheduling an extra core can cancel out the task running time saved and will also cause more energy consumption. Therefore, the energy consumption as a function of number of cores is highly nonlinear, with a minimum at three. In addition, we observe that most of the reduction of energy consumption is in the changing from one core to two cores, and then it does not change much any more. The system throughput is presented in Figure 4. As expected, the more resources allocated to the task the better the system throughput obtained. This result shows that, for computation-intensive tasks, the system throughput rises with the number of allocated cores and the increase of system throughput is nonlinear. Together with the results of Figure 3, we can conclude that energy consumption of

computation-intensive tasks is a more complex nonlinear function of allocated cores and execution time.

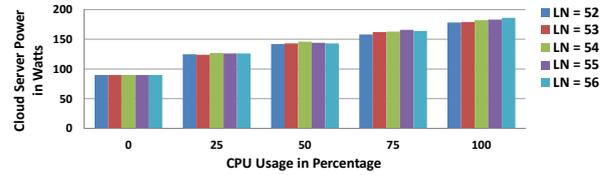


Figure 2. Server Power Consumption with Different Workload.

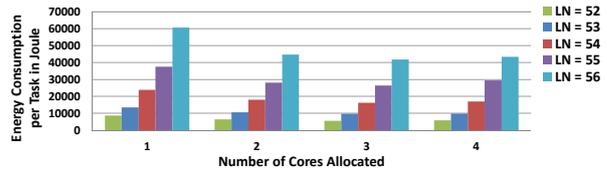


Figure 3. Energy Consumption per Task with Different Workload.

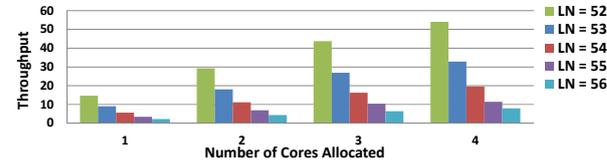


Figure 4. System Throughput with Different Workload.

Test Set 5.1.2: Keeping the number of tasks and resources allocated constant, gradually increasing the workload of the task, and turning Hyper-Threading on and off. Only one task was running on the test-bed at one time. This set of tests was run on an XLarge virtual machine. The results are presented in Figures 5 and 6. As shown by Figure 5, with Hyper-Threading enabled, there is a significant increase in the energy consumption of the task – the Hyper-Threading causing extra power consumption. In the case of “ $LN = 56$ ”, the extra energy consumption caused by Hyper-Threading is 8%. However, as shown in Figure 6, the extra system throughput obtained by enabling Hyper-Threading is not as much as expected. In cases of “ $LN = 53$ ”, “ $LN = 54$ ” and “ $LN = 55$ ”, Hyper-Threading even led to decrease in system throughput. This suggests that Hyper-Threading be disabled for computation-intensive tasks since there is little to gain from Hyper-Threading if a core’s execution resources are already well utilised.

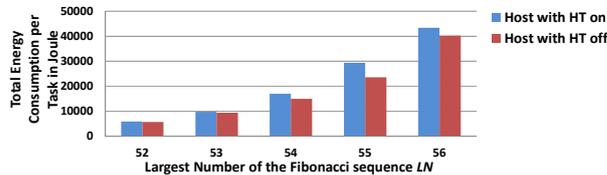


Figure 5. Energy Consumption per Task with HT on and off.

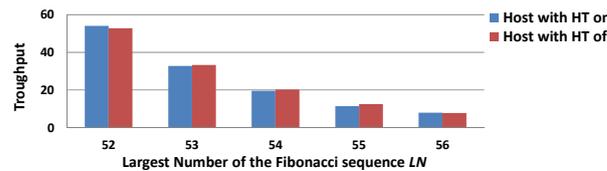


Figure 6. System Throughput with HT on and off.

Test Set 5.1.3: Keeping the workload of each task constant, and increasing the total number of tasks and number of VMs allocated to tasks. In this set of tests, the type of each VM was set to Small. Figure 7 shows the results. The number of running tasks can influence the average energy consumption of a single task: a larger number of tasks lead to higher average energy consumption of individual tasks. When the total number of tasks increased, more consequential scheduling overhead caused extra energy consumption. Moreover, it is clear that the increase in number of VMs significantly resulted in increase in per-task energy consumption. For instance, in the case of 6 tasks, the per-task energy consumption increased by 5% when we increased the number of VMs from one to two. As more VMs were deployed, the VM maintenance overhead increased as well. This caused more memory consumption, resulting in more power consumption. On the other hand, as depicted in Figure 8, the throughput was minimal when only one VM was configured. The throughput of two VMs and three VMs were at the same level. However, three VMs consumed 3% more energy than two VMs on average, as presented in Figure 7. Thus the two VMs configuration demonstrates higher energy efficiency than other configurations when there are multiple computation-intensive tasks running simultaneously.

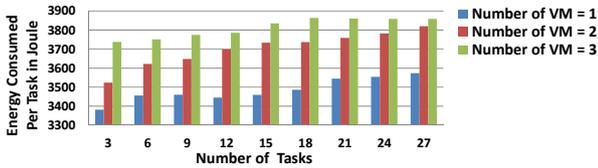


Figure 7. Energy Consumption per Task with Different VM Numbers.

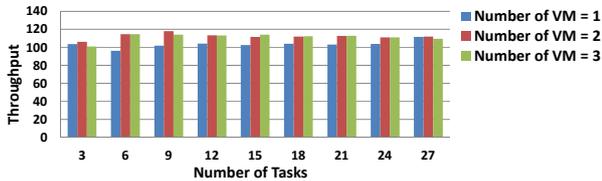


Figure 8. Throughput with Different VM Numbers.

5.2 Data-intensive Tasks Results

A data-intensive task needs to retrieve or store a large amount of data stored in one or more data storage servers. It requires high local disk I/O bandwidth in order to meet performance requirements. In reality the storage servers can be deployed in different data centres located in different geographic locations. However, we only consider the energy consumption in one data centre for the purpose of simplicity as the characteristics of data centres of the same service provider are often very similar.

We focused on the correlation between the energy consumption and the amount of data transferred in and out of the storage server. To profile and analyse the energy consumption caused by different data sizes and system configurations, we stress the cloud test-bed with data-intensive tasks generated using IOzone.

In every set of tests, we observed major spikes of energy consumption at the beginning and at the end of each task. They corresponded to the moments when disk I/O were stressed. We conducted five sets of tests in this series of experiments.

Test Set 5.2.1: Keeping the total number of tasks, process number of each task, resources allocated and VM

configuration constant, while increasing the total amount of data transferred and record size. This set of tests was performed on an XLarge virtual machine. Only one task was running on the cloud test-bed at one time and the number of processes was fixed at one. Therefore, the impact of scheduling overhead caused by multiple tasks and processes on energy consumption were avoided. As presented in Figure 9, server power consumption increased as the data record size decreased. However, the effect of record sizes on power consumption is nonlinear as displayed in Figure 9, with larger record sizes leading to convergence to a certain value. For instance, server power consumption did not change too much when we increased the record size from 8MB to 16MB. Smaller data record sizes led to generation of more overhead information for keeping track of where the data were located on the storage media. The overhead information consisted of the directory information, the space allocation and any other data that was not a part of the data to be transferred. As a result, the tasks used more computational resources, consuming more energy.

The results of system energy consumption and system throughput are displayed in Figure 10 and Figure 11, respectively. From Figure 10, we see that the energy consumption increases proportionally to the total amount of transferred data. The per-task energy consumption increases in a linear manner. As the total amount of data transferred increased, smaller record size would cause higher gradient when the total energy consumption increased because the execution time and the energy consumption increased simultaneously. As the record size decreased, more overhead information needed to be processed and stored. In summary, if only one process is configured to run the data-intensive task, bigger record size is more energy-efficient. As shown in Figure 11, the system throughput was higher when we set the record size to 64K, compared to other record sizes. As 64K is the typical record size Windows uses when applications try to transfer blocks of data bigger than 64K, data is transferred most efficiently in this size.

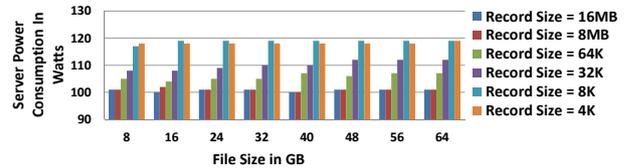


Figure 9. Server Power Consumption with Different File Size.

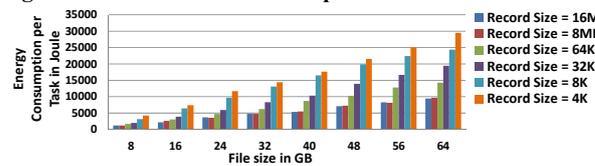


Figure 10. Energy Consumption per Task with Different File Size.

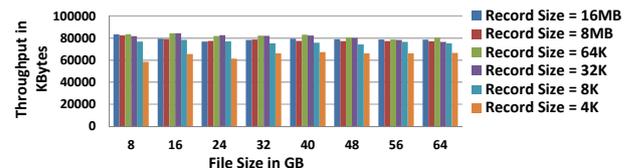


Figure 11. Throughput with Different File Size.

Test Set 5.2.2: Keeping total number of tasks, process number per task, total amount of data transferred and resource allocated constant, while changing VM configurations and record size of the task. This set of experiments was running on one XLarge VM, two Medium VMs and four Small VMs respectively so as to fix the size of the total virtual memory and the total number of virtual cores allocated to the VMs. The total number of tasks was set to four and the number of processes instantiated for each task was set to one. All the tasks were distributed on the VMs evenly. The total amount of data to be transferred was 64GB.

The system energy consumption and system throughput are presented in Figure 12 and Figure 13. We observed that the energy consumption increased and the throughput decreased as we changed the VM deployment from one XLarge VMs to four Small VMs. On one hand, VMs compete with each other for I/O bandwidth. On the other hand, more VMs will cause more read/write latency that lead to longer execution time of each task. As shown in Figure 12, the per-task energy consumption increased in a sub-linear manner. As shown in Figure 13, the best system throughput presented when we ran the workload on one XLarge VM. From these tests we conclude that running one VM for the entire I/O operation is most energy-efficient.

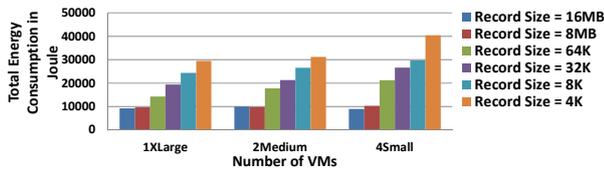


Figure 12. Energy Consumption with Different VM Configurations.

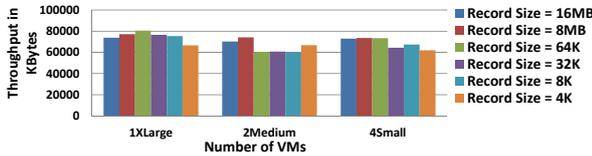


Figure 13. Throughput with Different VM Configurations.

Test Set 5.2.3: With single task, keeping the total amount of data transferred stable, resource allocation and VM configuration constant while increasing the process numbers for task and the record size of the task. An XLarge VM was deployed to run this set of tests. Only one task was running on the VM and it was assigned to transfer a 64GB file. As depicted in Figure 14, the per-task energy consumption increased with the number of processes. Although the power consumption of the server decreased slightly as the number of processes increased, as presented by Figure 15, multiple processes caused scheduling and synchronisation overhead. Thus, the total execution time of one data-intensive task increased accordingly. As a result, the system throughput decreased, as shown in Figure 16. Our conclusion is

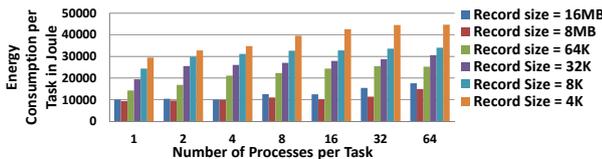


Figure 14. Energy Consumption per Task with Multiple Processes and Single VM.

that running data-intensive tasks with one sequential I/O operation is the most energy-efficient when the total amount of data transferred is fixed.

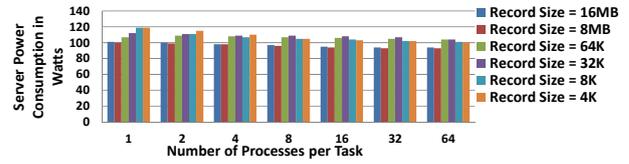


Figure 15. Server Power Consumption with Multiple Processes and Single VM.

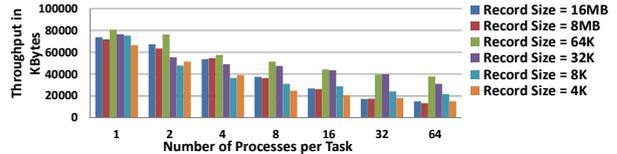


Figure 16. Throughput with Multiple Processes and Single VM.

Test Set 5.2.4: With multiple tasks, fix the total amount of data transferred, the resource allocation and number of processes for each task, changing VM configuration and record size for each task. The total number of tasks was set to four and the tasks were distributed to all the VMs evenly. We fixed the number of processes for each task running on each virtual machine at four and the total amount of data to be transferred of all the tasks at 64GB. The system energy consumption and system throughput are shown in Figure 17 and Figure 18, respectively. As we already observed in previous tests, energy consumption is greatly impacted by data record size. Smaller record size results in more energy consumption. The highest throughput was approximately 30% less than the highest throughput we obtained in test set 5.3.2. The numbers of process in both test sets were the same. However, there were multiple tasks running simultaneously this test set while only one task was running in test set 5.3.2. Therefore, multiple task scheduling causes extra scheduling overhead.

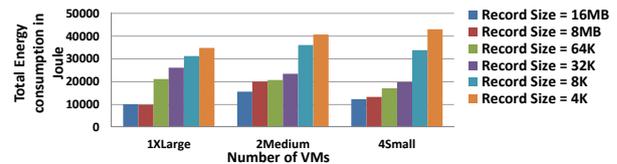


Figure 17. Energy Consumption with Multiple Processes and Multiple VMs.

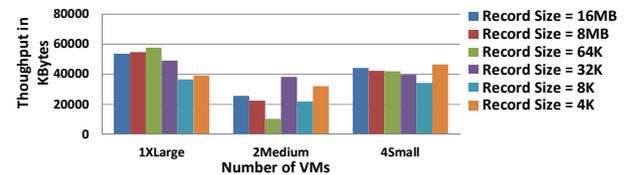


Figure 18. Throughput with Multiple Processes and Multiple VMs.

Test Set 5.2.5: Keeping the total number of tasks, the total amount of data transferred and the number of processes for each task constant, while changing the resources allocated, VM configuration and record size of each task. Only one task was running on the cloud test-bed at one time. We fixed the number of processes for the data-intensive task at one and the

total amount of data transferred at 64GB and scaled up the size of VM from Small to XLarge. These results are shown in Figure 19 and Figure 20. Compared to computation-intensive tasks, the VM type did not impact the energy consumption and throughput significantly when the record size was fixed. Running on an XLarge virtual machine was most energy-efficient when the record size was smaller than 64K. When the record sizes were 8M and 16M, running the data-intensive task on a Small virtual machine was most energy-efficient.

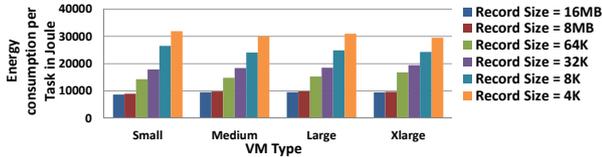


Figure 19. Energy Consumption per Task with Different

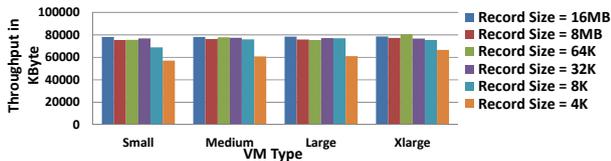
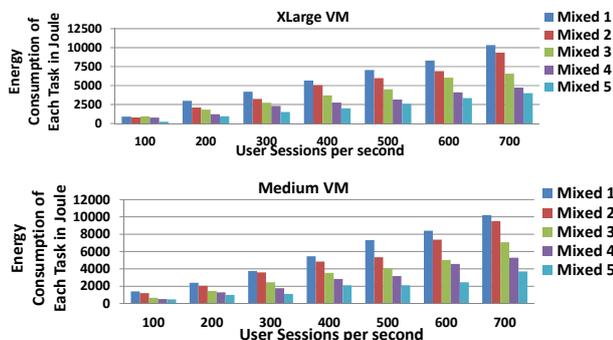


Figure 20. Throughput with Different VM.

5.3 Communication-intensive Tasks Results

A communication-intensive task requires a lot of network resources to transmit large amounts of data. Switches form the basis of the interconnection fabric of a private cloud network. Thus, switches, plus network cards, are the main energy consumers of the network resources. Traditionally, energy consumption of a switch depends on hardware parameters, such as the type of switch, the number of ports and the port transmission rates. However, the energy consumption may increase with the amount of data transferred over the network because of the data processing overhead. In addition, the total energy consumption might be impacted by network congestion because of the imbalance between the computation speed and the communication speed. The computation speed of the switch may be slower than the data transmission speed. We investigated this issue by applying different network workloads. A Cisco 2960 switch connected the cloud servers and client PCs. All parameters of the switch were set as default to avoid potential impact caused by switch configuration. We deployed the Web server and database server of JPetstore onto two different VMs to isolate the impact of different user requests.

In order to reveal the impact of the network packet size, the types of user requests and the system configuration on the energy



consumption of communication-intensive tasks, we conducted three sets of tests in this series of experiments, described as follows:

Test Set 5.3.1: Keeping the total number of user requests each user session, the resources allocated and the VM configuration of the Web server constant, while changing the total number of user sessions and workload type. In this set of tests, each user session consisted of mixed browsing requests or shopping requests in different percentages. There are five types of mixed workload, as presented in Table 4. Browsing requests include checking home page, viewing catalogue, viewing products, searching products and so on. Shopping requests include checking out, updating shopping cart, filling order forms, ordering inquiry, and so on. For each type of the mixed workload, we increased the number of concurrent user sessions from 100 to 700 in steps of 100. The resulting system energy consumption and system throughput are shown in Figures 21 and 22.

Table 4. Mixed Workload

	Mixed 1	Mixed 2	Mixed 3	Mixed 4	Mixed 5
Browsing Request	100%	80%	50%	20%	0%
Shopping Request	0%	20%	50%	80%	100%

As the number of concurrent user sessions increased, the energy consumption increased but the throughput decreased when we fixed the type of VM as presented in Figures 21 and 22. This was due to the increase in the number of concurrent user sessions causing extra scheduling overhead as each user request was processed. However, the decrease in throughput was not proportional to the increase in energy consumption. For instance, when running Mixed 1 on an XLarge virtual machine and after we changed the number of concurrent user sessions from 600 to 700, the system energy consumption increased by 24.7% and the system throughput decreased by 3.7%. In addition, we noticed that Mixed 1 consumed the most energy while Mixed 5 consumed the least energy among all five workload types, despite the type of VM. Meanwhile, we observed that Mixed 1 resulted in the lowest throughput compared to the other four types of mixed workload. Since Mixed 1 contained the most browsing requests and Mixed 5 contained the most shopping requests, Mixed 1 had the most disk and memory access of all as opposed to Mixed 5. Thus, it took longer to complete requests of type Mixed 1 compared to the other four workload types. As a result, the energy efficiency of Mixed 1 was lower than the other four workload types.

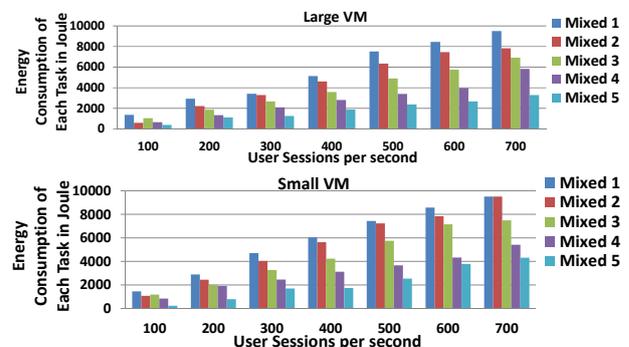


Figure 21. Energy Consumption per Task with Different VMs.

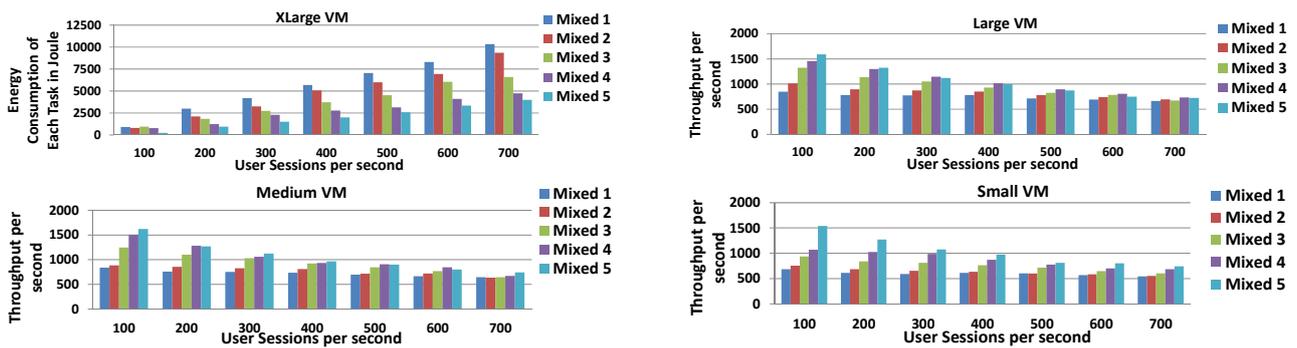


Figure 22. Throughput per Task with Different VMs.

Test Set 5.3.2: Keeping the total request number of each user session and the workload type constant, while changing the number of user sessions, the resources allocated and VM deployment on the Web server. The energy consumption and system throughput are presented in Figures 23 - 27. We observed that when the size of the VM deployed for the Web server increased from Small to Large, the system throughput increased while the energy consumption decreased in general. Intuitively, the more resources used the greater the energy consumption. However in this case, the smaller the instance the higher the disk accesses due to the thrashing of the cache, which leads to increase in energy consumption. Even for the Medium VM, which has two cores, the extra energy consumption by the additional core was

cancelled out by the much larger added memory, reducing the number of accesses to the database server. Noticeably, when the size of the VM deployed for the Web server changed from Large to XLarge, the system throughput decreased and the system energy consumption increased in general.

When we set the type of the VM to Large, the total capacity of the Web server and the database server reached the full capacity of the physical server. However, when we set the size of the VM for the Web server at XLarge, the total capacity of the Web server and the database server exceeded the full capacity of the host. In this situation, extra overhead occurred because of the scheduling of VMs, which caused extra energy consumption.

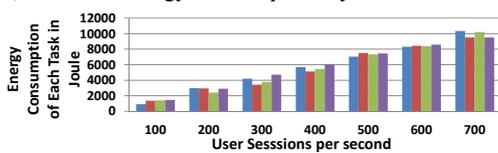


Figure 23. Energy Consumption and Throughput per Task with Mixed 1 Workload Type.

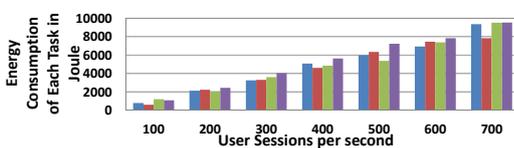
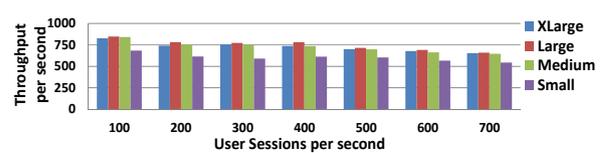


Figure 24. Energy Consumption and Throughput per Task with Mixed 2 Workload Type.

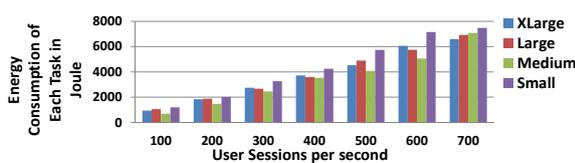
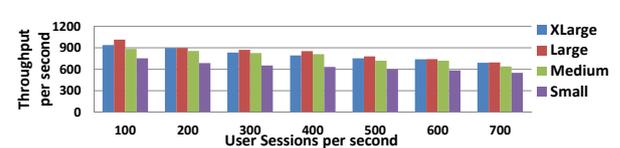


Figure 25. Energy Consumption and Throughput per Task with Mixed 3 Workload Type.

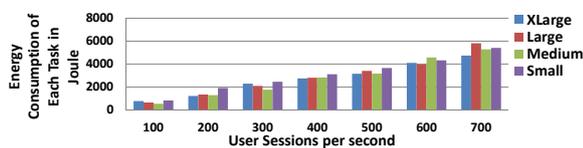
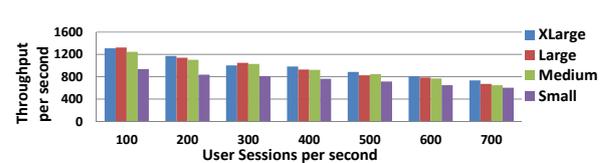


Figure 26. Energy Consumption and Throughput per Task with Mixed 4 Workload Type.

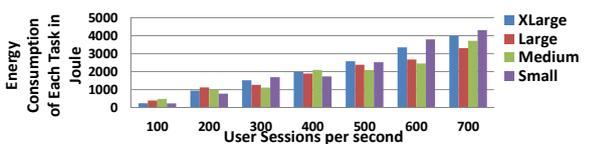
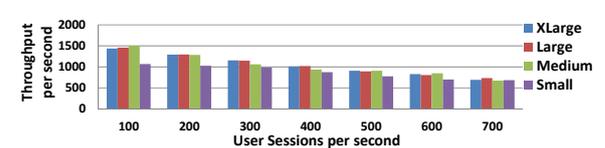
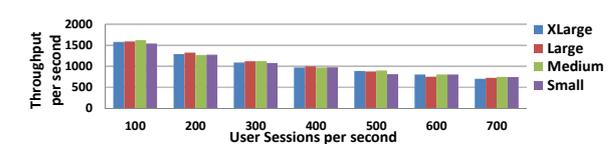


Figure 27. Energy Consumption and Throughput per Task with Mixed 5 Workload Type.



Test Set 5.3.3: Keeping the workload type, the number of user sessions constant, while changing the packet size, the resource allocation and the VM configuration. We fixed the number of concurrent user sessions of the workload at 300. First, we ran the test with only browsing requests and gradually increased the packet size of the browsing requests on different VM of different types. The packet size of each browsing request is listed in Table 5. These packet sizes represent the majority of all the browsing requests of JPtstore. The packet size of request “Browsing 1” is the smallest while packet size of request “Browsing 3” is the largest. The corresponding system energy consumption and system throughput are presented in Figures 28 and 29. As demonstrated, there was a slight increase in the energy consumption of each task when we increased the packet size of the requests. Furthermore, the throughput decreased as the packet size increased. Bigger packet size usually leads to more transmission time over the network and more processing time on both the servers and switches. Accordingly, throughput decreases and energy consumption increases for each communication-intensive task.

Table 5. Browsing Request Packet Size

	Browsing 1	Browsing 2	Browsing 3
Packet Size(KB)	3758	4190	4853

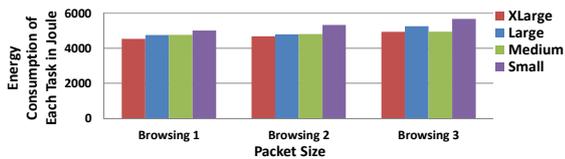


Figure 28. Energy Consumption per Task with Increased Packet Size.

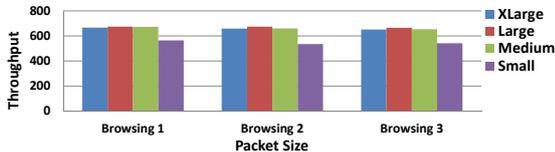


Figure 29. Throughput per Task with Increased Packet Size.

6. DISCUSSION

Based on our observations, we have derived a set of guidelines which can be adopted to achieve energy efficient resource provisioning and management for green cloud computing.

The more resources used by a single task, the more energy it consumes, and the better the system performance tends to be. However, based on the type of the runtime task, the declining throughput results show that overcommitted resource allocation would result in significant increase in energy consumption and decrease in overall system performance. For instance, energy consumption increased by 11% on average and the overall system performance decreased by 4% when the resource allocation exceeded the full capacity of the host during the tests of communication-intensive tasks. In cloud systems, the resource allocation should dynamically adapt to the customers’ needs, taking both performance (and other) SLAs and energy efficiency needs into account. Dynamic scaling-up is needed when the peak workload is likely to exceed the capacity of the cloud system. Based on our results, it is worth finding the trade-off between the energy cost caused by the overcommitted resource allocation and the cost introduced by adding new resources in cloud systems.

The types and workload of runtime tasks impact energy consumption significantly. The energy consumption of each task is highly coupled with the resource utilisation in cloud systems. Thus, it is important to predict the required resources accurately based on the types and workloads of runtime tasks. For some applications, their workload is either known or can be empirically determined and is relatively constant. However, due to the dynamic nature of many cloud applications and the demand of different hosting platforms, the workload of different runtime tasks in cloud systems can drastically change over time. The need to find out the workload patterns for different runtime tasks in cloud systems, in order to schedule them for optimal performance and energy consumption, has emerged.

For a specific type of task, the various configuration parameters associated with the task, that is, the number of processes, the size of data to be processed, and the size of data to be transmitted, can greatly affect the task’s energy consumption. These task parameters, that may originally come from application requirements, are closely linked to system configurations. Even with the same resource allocation, different system configurations can result in different energy consumption based on our observation. Therefore, dynamically changing cloud systems configurations is needed to adapt to different tasks based on their various configuration parameters.

Differing task types, task workload, task parameters, and cloud platform configurations can dramatically affect task throughput performance and energy consumption. As discussed, some of these effects are predictable, while others are counter-intuitive. Certain configurations give optimal balance of maximising workload vs minimising energy consumption. However, there may still be times that customers and/or cloud providers choose to prioritise one over the other, that is, to sacrifice performance to maximise energy efficiency further, or to sacrifice energy efficiency for improved performance.

7. CONCLUSIONS AND FUTURE WORK

Understanding cloud system energy consumption dynamics is valuable for developing efficient energy-aware resource management techniques for green cloud computing. We conducted a number of empirical experiments and profiled the energy consumption and performance with different types of runtime tasks on a controlled, representative cloud system. We treated a single task as a unit and measured the energy consumed by the task under various system configurations and task workload. The correlation of system throughput and energy consumption were analysed based on our experimental results. These results provide guidelines for developing energy management techniques for cloud systems that aim to reduce the energy consumption while achieving sufficient system performance to meet customer Service Level Agreements (SLAs).

Currently, we are determining the functional shape of our energy consumption model based on the experimental results analysis. Moreover, we are running experiments including large scale composite workload on larger cloud platform. We compare the energy consumption of individual task against the energy consumption of composite load predicted by our model. For computation-intensive tasks, we are selecting CPU-intensive and memory-intensive applications which inherently have single threaded and multi threaded algorithms instead of developing a new application. The experiment results will be together to help us improve our energy consumption model. In addition, we will

integrate an energy cost rate and an “energy dirtiness rate” into our energy consumption model to factor in the costs – monetary and environmental – of cloud energy generated by different resources. This enhanced energy cost model will be investigated in order to minimise total energy costs while meeting system performance needs.

8. ACKNOWLEDGMENTS

We thank Professor Ryszard Kowalczyk for providing the facilities of Swinburne Energy Research Lab. This research is supported by the Australian Research Council under Discovery Project DP110101340.

9. REFERENCES

- [1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M. Above the clouds: a Berkeley view of cloud computing. Technical Report UCB/EECS-2009-28, UC Berkeley Reliable Adaptive Distributed Systems Laboratory, USA, 2009.
- [2] Fanara, A., Haines, E. and Howard, A. The State of Energy and Performance Benchmarking for Enterprise Servers. In *Proceedings of the 1st TPC Technology Conference(TPCTC2009)*, pages 52-56, Lyon, France, 2009.
- [3] Hamilto, J. Cooperative expendable micro-slice servers (CEMS): low cost, low power servers for internet-scale services. In *Proceedings of the 4th Biennial Conference on Innovative Data Systems Research(CIDR2009)*, pages 1-8, Asilomar, California, USA, 2009.
- [4] Baliga, J., Ayre, R. W. A., Hinton, K. and Tucker, R. S. Green cloud computing: balancing energy in processing, storage, and transport. *Proceedings of the IEEE*, 99 (1): 149-167, 2011.
- [5] Shang, L., Peh, L.-S. and Jha, N. K. Dynamic voltage scaling with links for power optimization of interconnection networks. In *Proceedings of the 9th International Symposium on High-Performance Computer Architecture(HPCA2003)*, pages 91-102, Anaheim, California, USA, 2003.
- [6] Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., Pratt, I. and Warfield, A. Live migration of virtual machines. In *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation(NSDI2005)*, pages 273-286, Boston, Massachusetts, USA, 2005.
- [7] Raghavendra, R., Ranganathan, P., Talwar, V., Wang, Z. and Zhu, X. No "power" struggles: coordinated multi-level power management for the data center. In *Proceedings of the 13th International Conference on Architectural Support for Programming Languages and Operating Systems(ASPLOS2008)*, pages 48-59, Seattle, WA, USA, 2008.
- [8] Zhang, Z. and Fu, S. Characterizing power and energy usage in cloud computing systems. In *Proceedings of the 3rd IEEE International Conference on Cloud Computing Technology and Science(CloudCom2011)*, pages 146-153, Athens, Greece, 2011.
- [9] Chen, F., Schneider, J.-G., Yang, Y., Grundy, J. and He, Q. An energy consumption model and analysis tool for Cloud computing environments. In *Proceedings of the 1st International Workshop on Green and Sustainable Software(GREENS2012)*, pages 45-50, Zurich, Switzerland, 2012.
- [10] Jung, G., Hiltunen, M. A. and Joshi, K. R. Mistral: dynamically managing power, performance, and adaptation cost in cloud infrastructures. In *Proceedings of the International Conference on Distributed Computing Systems(ICDCS2010)*, pages 62-73, Genova, Italy, 2010.
- [11] Mach, W. and Schikuta, E. A consumer-provider cloud cost model considering variable cost. In *Proceedings of the 9th IEEE International Conference on Dependable, Autonomic and Secure Computing(DASC2011)*, pages 628-635, Sydney, Australia, 2011.
- [12] Lee, Y. C. and Zomaya, A. Y. Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*, 60 (2): 268-280, 2012.
- [13] Liu, L., Wang, H., Liu, X., Jin, X., He, W., Wang, Q. and Chen, Y. GreenCloud: a new architecture for green data center. In *Proceedings of IEEE Conference on Autonomic Computing(ICAC2009)*, pages 29-38, Barcelona, Spain, 2009.
- [14] Verma, A., Ahuja, P. and Neogi, A. Power-aware dynamic placement of hpc applications. In *Proceedings of the 22nd Annual International Conference on Supercomputing(ICS2008)*, pages 175-184, Island of Kos, Greece, 2008.
- [15] Nathuji, R. and Schwan, K. VirtualPower: coordinated power management in virtualized enterprise systems. In *Proceedings of the 21st ACM Symposium on Operating Systems Principles(SOSP2007)*, pages 265-278, Stevenson, Washington, USA, 2007.
- [16] Stoess, J., Lang, C. and Bellosa, F. Energy management for hypervisor-based virtual machines. In *Proceedings of the 2007 USENIX Annual Technical Conference(USENIX2007)*, pages 1-14, Santa Clara, CA, USA, 2007.
- [17] Chen, Q., Grosso, P., van der Veldt, K., de Laat, C., Hofman, R. and Bal, H. Profiling energy consumption of VMs for green cloud computing. In *Proceedings of the 9th IEEE International Conference on Dependable, Autonomic and Secure Computing(DASC2011)*, pages 768-775, Sydney, Australia, 2011.
- [18] Kansal, A., Zhao, F., Kothari, N. and Bhattacharya, A. A. Virtual machine power metering and provisioning. In *Proceedings of the 1st ACM Symposium on Cloud Computing(SoCC2010)*, pages 39-50, Indianapolis, Indiana, USA, 2010.
- [19] Lefèvre, L. and Orgerie, A.-C. Designing and evaluating an energy efficient Cloud. *Journal of Supercomputing*, 51 (3): 352-373, 2010.
- [20] Zhang, Q., Cheng, L. and Boutaba, R. Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1 (1): 7-18, 2010.
- [21] Koller, R., Verma, A. and Neogi, A. WattApp: An Application Aware Power Meter for Shared Data Centers. In *Proceedings of the 7th International Conference on Autonomic Computing(ICAC2010)*, pages 31-40, Reston, VA, USA, 2010.
- [22] Schuster, A. Searching for processes and threads in Microsoft Windows memory dumps. *Digital Investigation*, 3 (1): 10-16, 2006.