Performing benchmarking to compare the performance of Google Cloud Platform VM's

Suchina Parihar
Dept. of Electrical and Computer Engineering
University of Calgary
Calgary,Canada
Suchina.parihar2@ucalgary.ca

Abstract: Cloud computing today has now been growing as new technology and new business models. In distributed technology perspective, cloud computing most like client-server services uses virtual resources to execute. Currently, cloud computing relies on the use of elastic virtual machines and the use of network for data exchange. In this paper, an experiment will be conducted to measure the quality of service received by cloud computing customers in terms of different types of services provided to them by Google Cloud Platform Virtual Machines. We interest to know about the functionality of different Virtual machine types. This paper presents a comparative analysis between different machine types in terms of cost, performance, memory, bandwidth and availability. The results of this study can be used as a reference model for the Google Cloud users in choosing services from different types of Virtual machines according to their choice of requirements to be fulfilled in order to obtain optimal quality services [1].

Keywords—Benchmarking; Perfkit Benchmarker; Virtual Machines; latency; throughput; bandwidth.

I. INTRODUCTION

Cloud computing has emerged as a very important commercial infrastructure that promises to reduce the need for maintaining costly computing facilities by organizations and institutes [8]. It is a complex environment which is dependent on a number of IT infrastructures, including components that are often widely geographically dispersed, with shared elements and running diverse applications [2][3]. Through the use of virtualization and time sharing of resources, clouds serve with a single set of physical resources as a large user base with altogether different needs. Thus, the clouds have the promise to provide to their owners the benefits of an economy of calibration and, at the same time, become a substitute for scientists to clusters, grids, and parallel production conditions. In this paper, the performance of cloud computing virtual machines is analyzed. Our effective method demonstrates to yield comparative results of different machine types [4][8]. Cloud computing delivers infrastructure, platform, and software as services, which are made available as subscription-based services in a pay-as-you-go model to consumers. These services in industry are respectively referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [5].

1.1 CLOUD COMPUTING PERFORMANCE:

Bautista et al. [6] propose a performance measurement framework for cloud computing systems, which integrates the software quality concepts proposed by ISO 25010, which states: "*The performance of a Cloud Computing system is determined by analysis of the characteristics involved in performing an efficient and reliable service that meets requirements under stated conditions and within the maximum limits of the system parameters*" [7].

This paper proposes a framework of cloud service performance measurement system to provide criteria for choosing cloud service with confidence according to needs by providing objective and quantitative comparison and evaluation results on cloud service. In particular, this paper defines the scope of system performance measurement of virtual machine (VM) of infrastructure service (IaaS) among various areas of cloud service and deduces performance measurement items by analyzing requirements. Based on this, this paper will develop a framework of performance measurement system of VM to provide assistances in the performance management of IaaS [9]. IaaS as a cloud service is providing a service for easily and quickly creating, managing and monitoring server resources such as CPU, memory, disk, network and OS via internet browser, and hardware platform can be divided into several logical units that are respectively referred to as virtual machines (VM) [10]. VM refers to server resources such as CPU, memory and disk that are ready to be used through allocation or appropriate arrangement, and analyzes the requirements of measuring VM performance from the VM creation to extinction according to users' request in the virtualization area of cloud service infrastructure system.

1.2 GOOGLE CLOUD PLATFORM:

Google Cloud Platform is a cloud computing service by Google that offers hosting on the same supporting infrastructure that Google uses internally for end-user products like Google Search and YouTube. Cloud Platform provides developer products to build a range of programs from simple websites to complex applications [11]. Google Cloud Platform is a part of a suite of enterprise services from Google Cloud and provides a set of modular cloud-based services with a host of development tools. For example, hosting and computing, cloud storage, data storage, translations APIs and prediction APIs. Google Cloud Platform offers many Virtual Machine instances which are to be compared in this paper.

2. PERFKIT BENCHMARKER

PerfKit Benchmarker is an open source benchmarking tool used to measure and compare cloud offerings. PerfKit Benchmarker is licensed under the Apache 2 license terms. PerfKit Benchmarker is a community effort involving over 500 participants including researchers, academic institutions and companies together with the originator, Google. PerfKit Benchmarker (PKB) is a community effort to deliver a repeatable, consistent, and open way of measuring Cloud Performance. It supports a growing list of cloud providers including: AliCloud, AmazonWebServices, CloudStack, Digita lOcean, GoogleCloudPlatform, Kubernetes, MicrosoftAzure, OpenStack, Rackspace[27]. In addition to Cloud Providers to supports container orchestration including Kubernetes and Mesos and local "static" workstations and clusters of computers. The goal is to create an open source living benchmark [framework] that represents how Cloud developers are building applications, evaluating Cloud alternatives, learning how to architect applications for each cloud. Living because it will change and morph quickly as developers change [25].

PerfKit Benchmarker measures the end to end time to provision resources in the cloud, in addition to reporting on the most standard metrics of peak performance, e.g.: latency, throughput, time-to-complete, IOPS. PerfKit Benchmarker reduces the complexity in running benchmarks on supported cloud providers by unified and simple commands. It's designed to operate via vendor provided command line tools. PerfKit Benchmarker contains a canonical set of public benchmarks shown in Table 1 and Table 2. All benchmarks are running with default/initial state and configuration [25][26].

	Big Data/IoT	**High Performance Computing** **Scientific Computing**	**Simulation**	**Web Benchmarks**
Workloads	AerospikeYCSB Hadoop Terasort Cassandra YCSB HBase YCSB MongoDB YCSB Redis YCSB	HPCC Scimark2	Oldisim etcd	EPFL CS Web Search EPFL CS Web Serving Tomcat

Table 1: List of benchmarks provided by PerfKit Benchmarker

	Storage Benchmarks	**CPU Benchmarks**	**Network Benchmarks**	**System**
Micro Benchmarks	Bonnie File Copy Fio Google Cloud BigTable Object Storage Synthetic Storage Sysbench OLTP	Coremark Spec CPU 2006	Iperf Netperf Ping Mesh Network	Cluster Boot UnixBench

Table 2: List of micro benchmarks provided by PerfKit Benchmarker

3. RELATED WORK

Pavlo et al. [13] presented a micro benchmark for big data analytics. It is used for compaing Hadoop-based analytics to a row-based RDBMS system and a column-based RDBMS one. Spark [14] and Shark [15] systems inspire the AMP Lab big data benchmarks [16], which targets real-time analytics. GridMix[17] is a benchmark specially designed for Hadoop MapReduce, which includes only micro benchmarks for text data. Yahoo released their cloud benchmark specially for data storage systems, NYCSB [18]. Bautista et al. [6] propose a performance measurement framework for cloud computing systems. Simon Ostermann, Alexandria Iosup, Nezih Yigitbasi, Radu Prodan, Thomas Fahringer, Dick Epema [19] presented an evaluation of the usefulness of the current cloud computing services for scientific computing. HiBench [20] is a benchmark suite for Hadoop MapReduce and Hive. Armstrong et al. [19] characterized the social graph data and database workloads for Facebook's social network, and presented the motivation, design, and implementation of LinkBench, a database benchmark that reflects real-world database workloads for social network applications. The TeraSort or GraySort benchmark [21] considers the performance and cost involved in sorting a large number of 100- byte records, and its workload is not sufficient to cover the various needs of big data processing. BigBench [22] is the recent effort towards designing big data benchmarks. BigBench focuses on big data offline analytics, thus adopting TPC-DS as the basis and adding atop new data types like semi-/un-structured data, as well as non-relational workloads. Recently, architecture communities also proposed CloudSuite [23] for scale-out cloud workloads, and DCBench [24] for datacenter workloads.

4. PERFKIT BENCHMARKER METHODOLOGY

4.1 BENCHMARKING REQUIREMENTS:

PerfKit Benchmarker provides wrappers and workload definitions around popular benchmark tools. It instantiates VMs on the Cloud provider of your choice, automatically installs benchmarks, and runs the workloads without user interaction [26]. Before proceeding with the execution of the benchmarks, we need to follow certain installation steps as follows:

STEP 1: Accepting the license of each of the benchmarks individually, and take responsibility for using them before you use the PerfKit Benchmarker.

STEP 2: Some of the benchmarks invoked require Java. We must also agree with the corresponding license.

STEP 3: Before we can run the PerfKit Benchmarker, we need accounts on the cloud providers we want to benchmark.

STEP 4: We also need the software dependencies, which are mostly command line tools and credentials to access our accounts without a password.

STEP 5: We need to install Python 2.7 and pip.

STEP 6: Download PerfKit Benchmarker from GitHub and install PerfKit Benchmarker dependencies.

STEP 7: Install gcloud and setup authentication

STEP 8: Create and configure a .boto file for object storage benchmarks and install Image prerequisites for Docker based clouds.

STEP 9: We are ready to run the PerfKit Benchmarker.

4.2 METHODOLOGY:

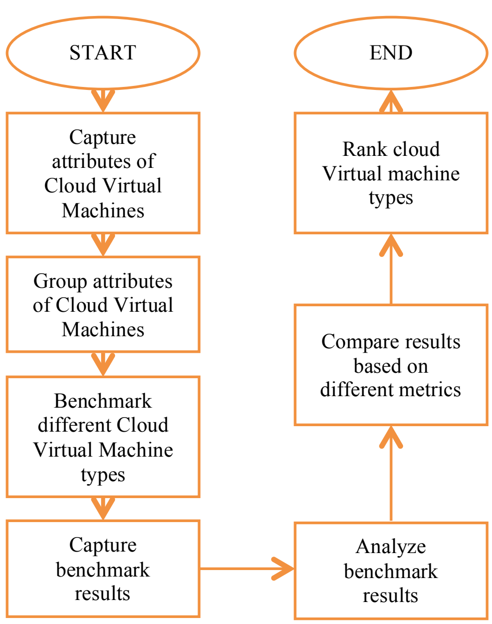

Figure 1: Methodology of the project

5. BENCHMARKING RESULTS

The experimental setup for obtaining the attributes of VMs by benchmarking and then grouping the attributes are presented in this section. The Google Cloud Platform(GCP) is a rapidly growing public cloud infrastructure offering a variety of VMs with different performance capabilities. Hence, GCP is the platform chosen for this research. From the benchmarks provided by Perfkit Benchmarker, four tools relevant for my study were chosen, namely (1) Iperf is used to provide throughput and execution time information (2) Ping tool can provide latency and bandwidth information (3) HPCC tool, commonly referred to as the High-Performance Linpack Benchmark, is also used for combining several benchmarks to test a number of independent attributes of the performance of high-performance computer (HPC) systems (4) Unixbench is used to test various aspects of System's performance. All experiments to gather the VM attributes were performed between four to six times. The following analysis were done:

I. CPU Performance
II. IO Performance
III. Cost analysis
IV. Throughput analysis
V. Computation analysis
VI. Latency analysis
VII. Availability of VMs
VIII. Memory analysis

5.1: CPU PERFORMANCE:

CPU performance is measured using a subset of the UnixBench tests, namely:

1. dhry2reg -- Dhrystone CPU using two register variables
2. whetstone-double -- Whetstone double precision CPU test
3. pipe -- Unix pipe throughput
4. context1 -- Pipe based context switching throughput
5. shell8 -- 8 bash shells executing simultaneously

A higher score means better CPU performance as shown in Figure 3. Additionally, Figure 2 shows the number of CPUs used by the different machine types.

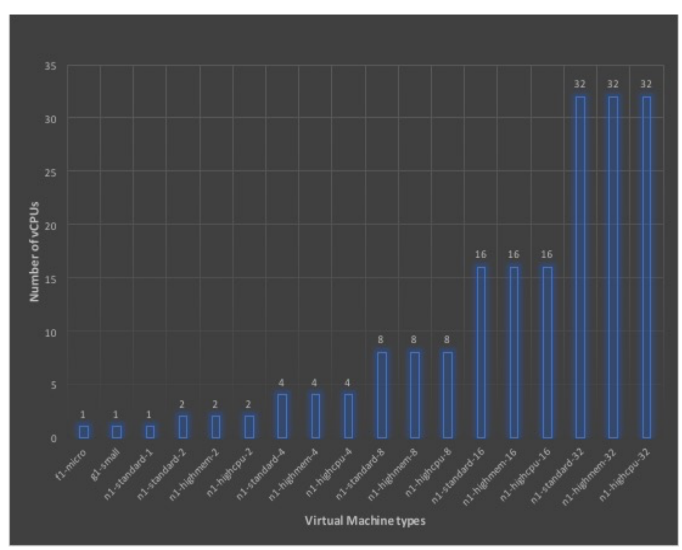

Figure 2: Number of vCPUs for different VM types

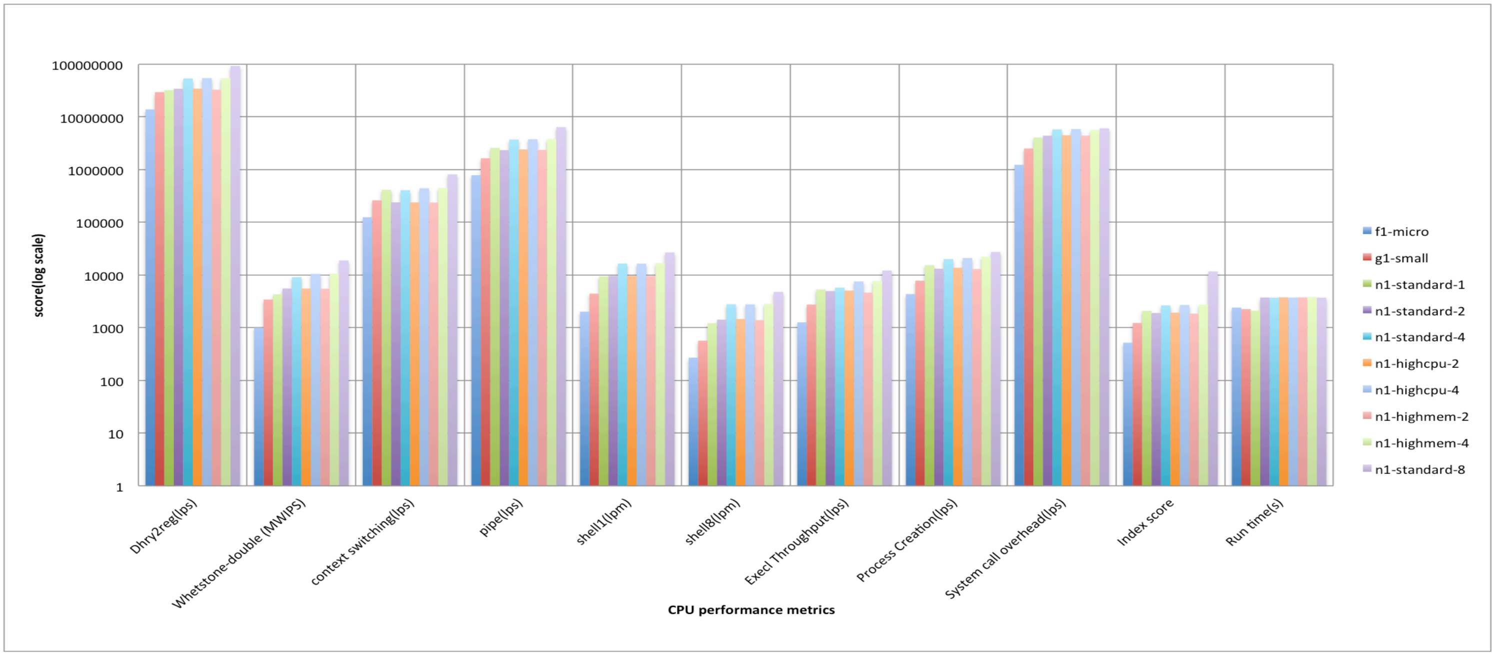

Figure 3: CPU Performance results obtained from Unixbench Benchmark

RESULTS: If we consider, Figure 3 alone then it is clearly visible that out of all the VM's, n1-standard-8 shows better CPU performance than others. But we also need to consider variance (pie-chart in appendix B) of the results. Thus, n1-standard-8 is 48% variable and thus we cannot consider it to be the best among other machine types.
Thus, overall we can say that n1-standard-4 shows better results as well and is 15% variable. Also n1-highcpu-4 shows good performance with 17% variability.

5.2: IO PERFORMANCE:
IO performance is measured using a subset of the UnixBench tests, namely:
1. fstime -- file copy, 1024 byte buffer size, 2000 maxblocks
2. fsbuffer -- file copy, 256 byte buffer size, 500 maxblocks
3. fsdisk -- file copy, 4096 by buffer size, 8000 maxblocks
The pie chart for variance is in Appendix B

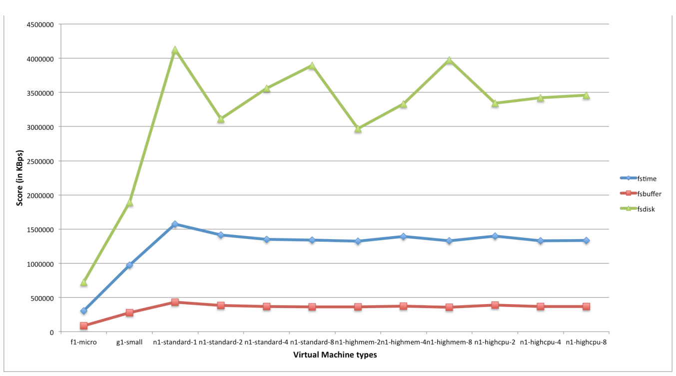

Figure 4: IO Performance results obtained from Unixbench

RESULTS: The IO performance of n1-standard-1 VM type is better than others. The performance of n1-standard-n8 is also good if we consider only mean as it is 48% variable. Generally the CPU volatility is less than the IO volatility and volatility looks worse at the higher performing instances.
Thus overall, n1-standard-1 and n1-highmem-8 have better IO performance.

5.3: COST ANALYSIS:
Different types of Virtual machine are compared on the basis of their hourly and monthly price as shown in Figure 5 and Figure 6, respectively. Price is the monthly cost using hour-hour terms, normalized to 720hrs/month.

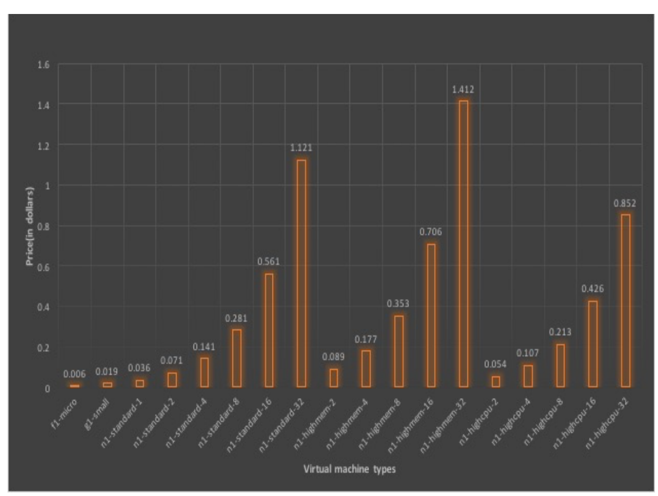

Figure 5: Cost per hour of different VM types

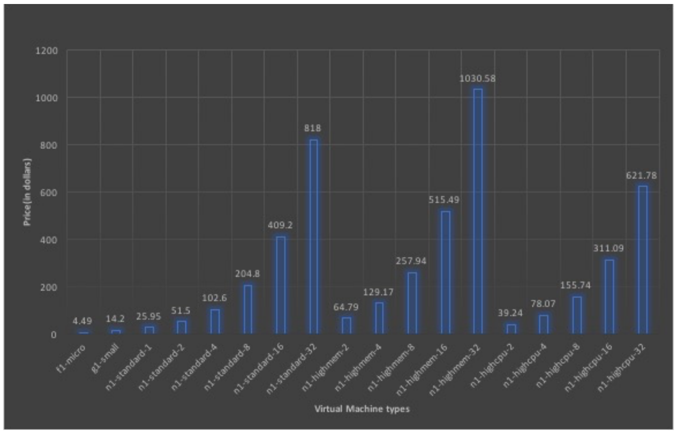

Figure 6: Cost per month of different VM types

RESULTS: Cost analysis is actually a trade-off between the services provided and the price. Here, f1-micro is clearly least expensive out of all the VM types but it only provides 0.6 GB of memory and has only one shared vCPU. Thus, it is not a good tradeoff. Thus, it becomes tricky to find the most cost effective VM as it depends upon the requirements of the customer.
But overall n1-highcpu-2 and n1-highmem-2 are the ones that can fit in ones budget and also provide a decent memory space and CPU performance.

5.4:THROUGHPUT ANALYSIS:
For the Throughput results, Iperf benchmark was used that gives us the values of throughput, execution time and number of bytes transferred by different machine types(in the same zone as well as cross zones). Figure 10 shows throughput timeline for different dates on which the benchmark was run.

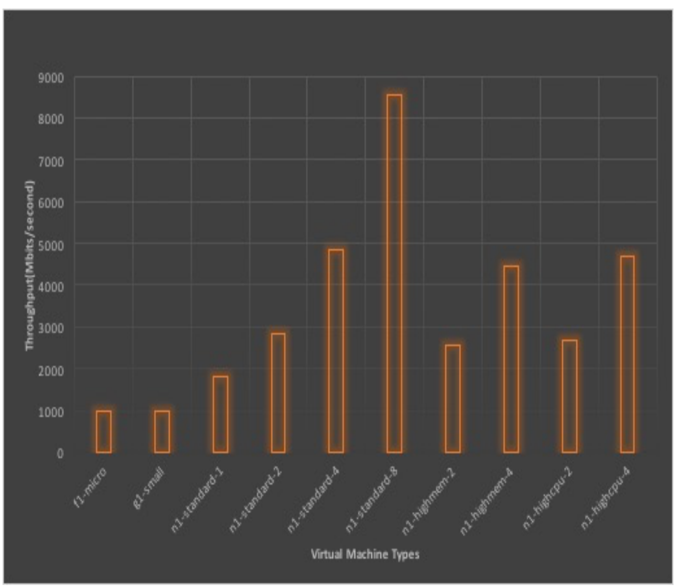

Figure 7: Throughput results obtained from Iperf benchmark

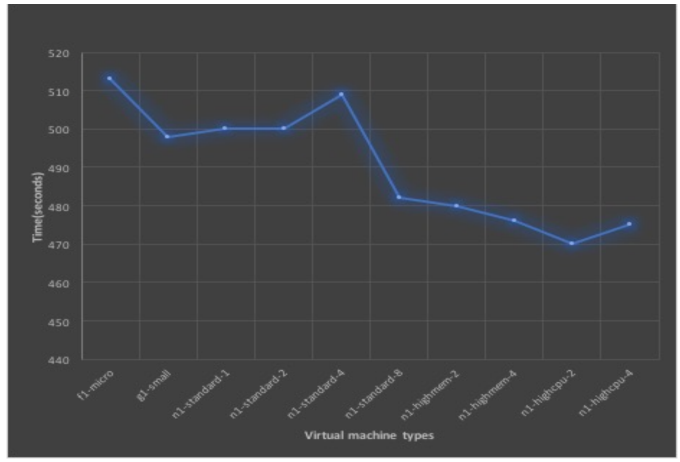

Figure 8: Execution time obtained from Iperf benchmark

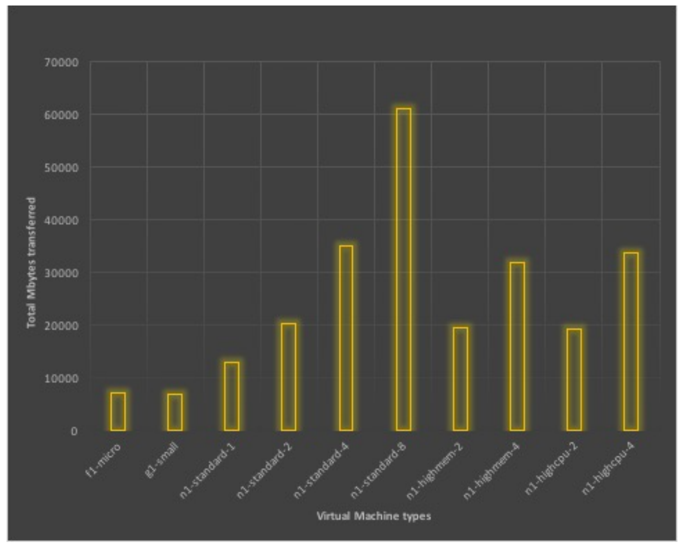

Figure 9: Bandwidth obtained from Iperf benchmark

RESULTS: It is clear from Figure 7 that n1-standard 8 has the highest throughput value but it 48% variable(shown in pie chart in Appendix B), therefore not a better choice to go forward with. Thus n1-standard-4 has the second highest throughput value with 14% variability but its execution time is higher.
Hence, the better choice would be n1-highcpu-4 and n1-higmem-4 as their execution times are lower and they have 15% and 17% variability.

5.5:COMPUTATION ANALYSIS:
High-Performance Linpack Benchmark(HPCC) is used to calculate different attributes of performance of high performance computer systems.
RESULTS: Figure 11 shows the results of HPCC benchmark. The results again depend upon the requirements of the customer and thus we cannot pick out one choice and say it has best HPCC performance attributes.

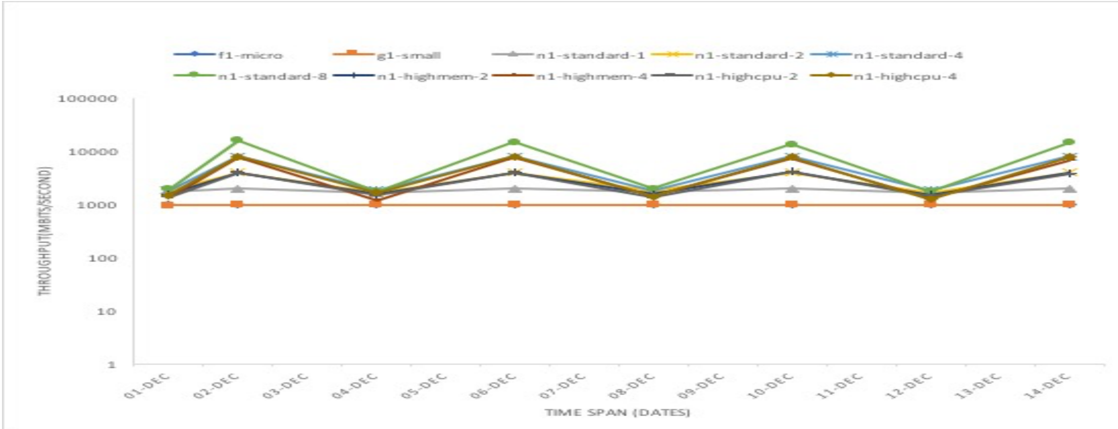

Figure 10: Time for throughput results of different VMs normalized to Logarithmic scale

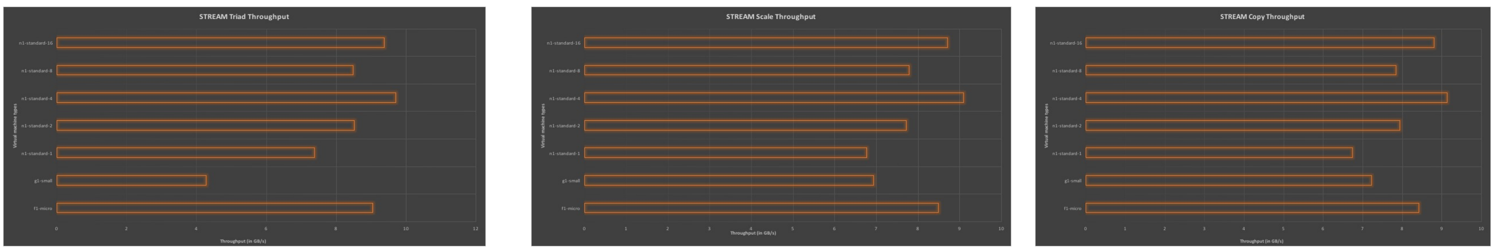

Figure 11.1, 11.2, 11.3: Stream Triad Throughput, Stream Scale Throughput and Stream Copy Throughput

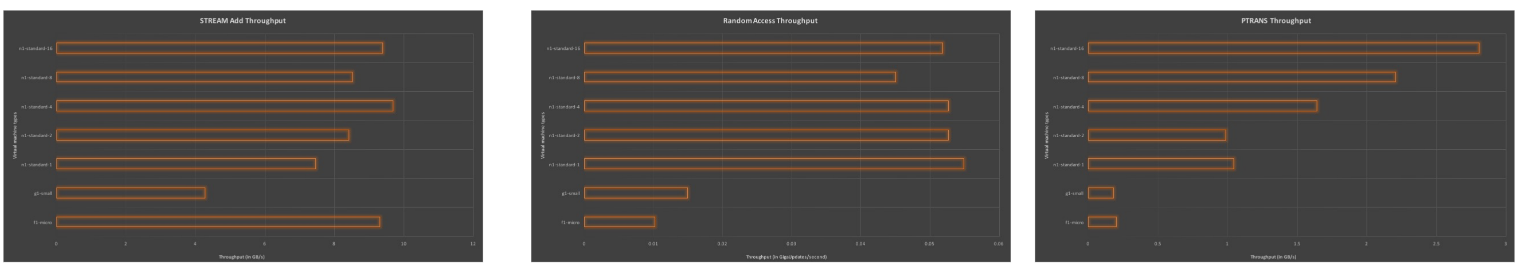

Figure 11.4, 11.5, 11.6: Stream ADD Throughput, Random Access throughput and Ptrans throughput

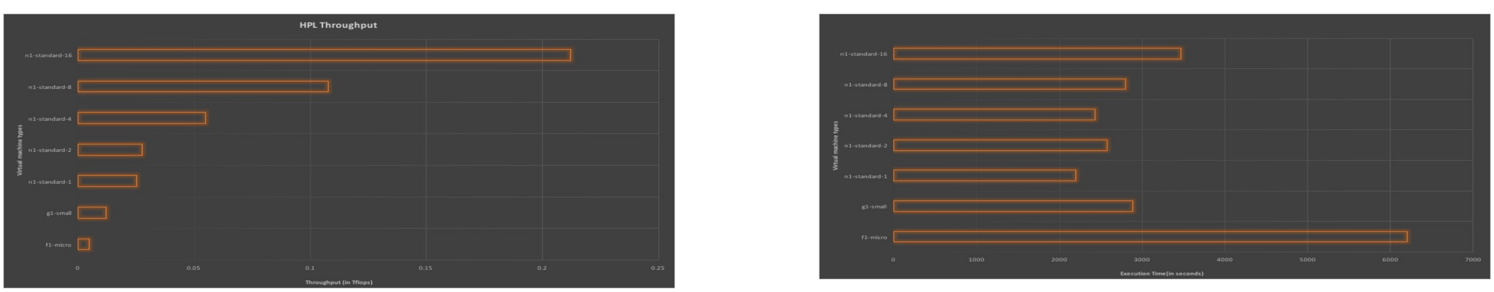

Figure 11.7, 11.8: HPL throughput and Execution time

Figure 11: Results of HPCC benchmark showing different attributes of performance

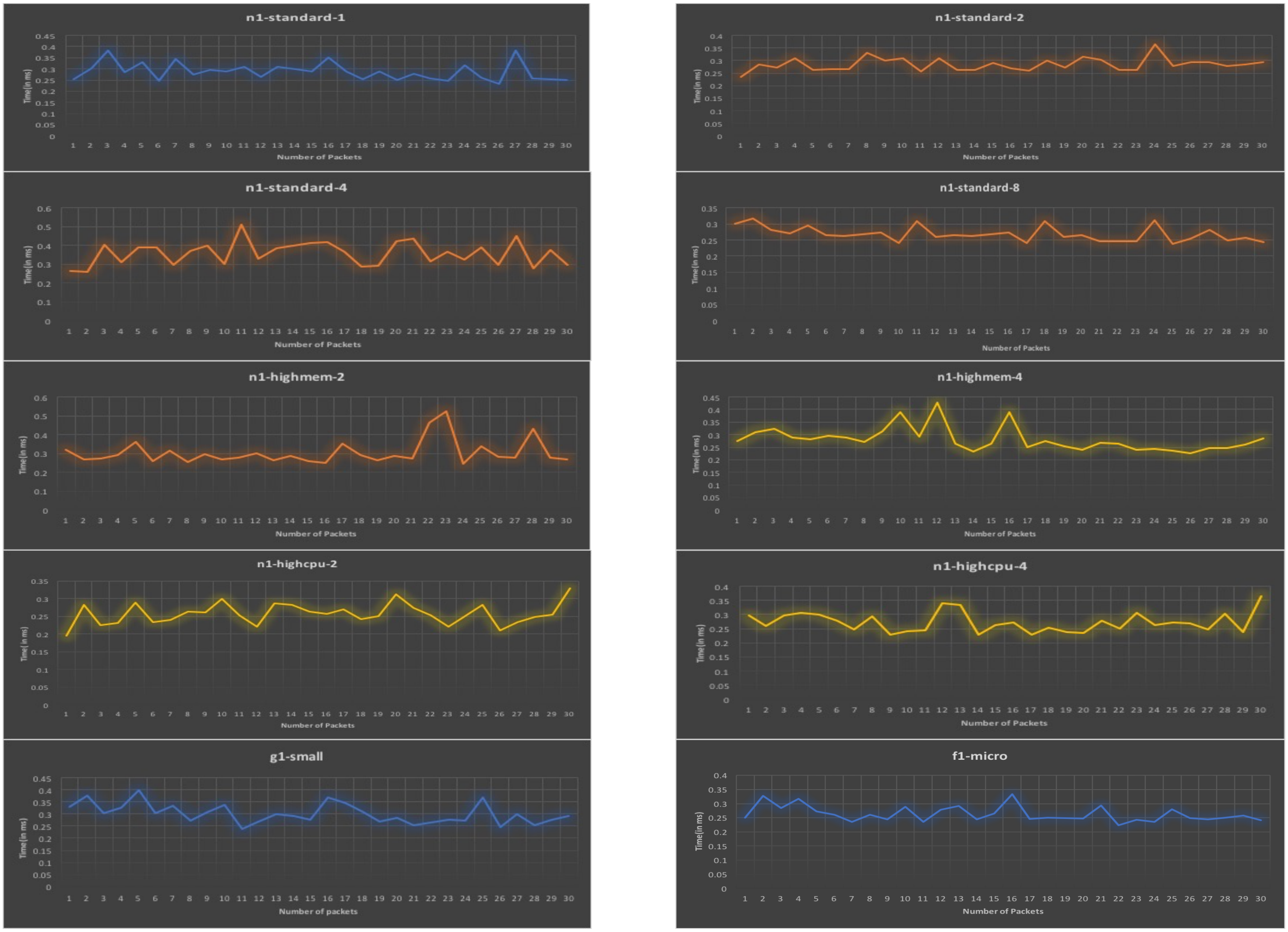

Figure 12: Latency results of individual packets from 1 to 30 for different Virtual machine types

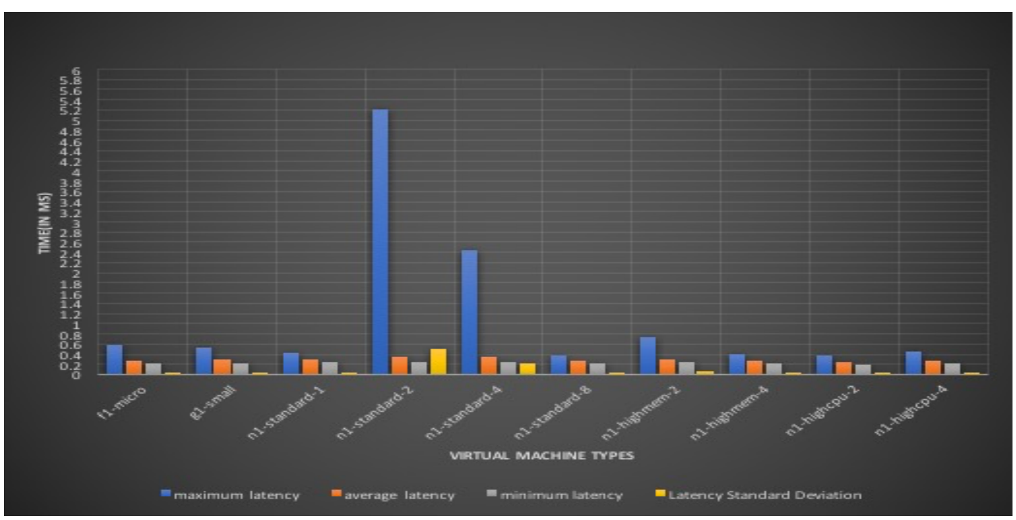

Figure 13: Average latency, Minimum latency, maximum latency and latency standard deviation obtained from ping benchmark

5.6 LATENCY ANALYSIS:

For this analysis, ping benchmark is used which gives the results for minimum, maximum, average latency and also provides latency standard deviation values for different machine types.

RESULTS: Figure 12 and 13 shows the latency values for different VM's. From Figure 13, it is clear that n1-highcpu-2 and n1-highcpu-4 has lower latency and less standard deviation values, therefore provide better performance as compared to other VM types.

5.7 AVAILABILITY OF VMs:

Availability of Virtual Machines at different regions proves to be a better metric to figure out which VM is better than the other. The availability is given in Table C in Appendix C. The availability is checked in 5 regions to see whether particular VM instance types are present is those regions or not.

Even though most performance and value scores are marginally, but noticeably, the difference in availability of services is significant, n1-standard-16 is not available in the US and the n1-higmem-16 instance type is not available in Asia and Europe

5.8: MEMORY ANALYSIS:

This analysis shows how much memory is provided by different instance types. And from Figure 14, we can see that n1-highmem-32 provides most memory of 208GB followed by n1-standard-32 which provides 120 GB. N1-highmem-16 provides 104 GB of memory.

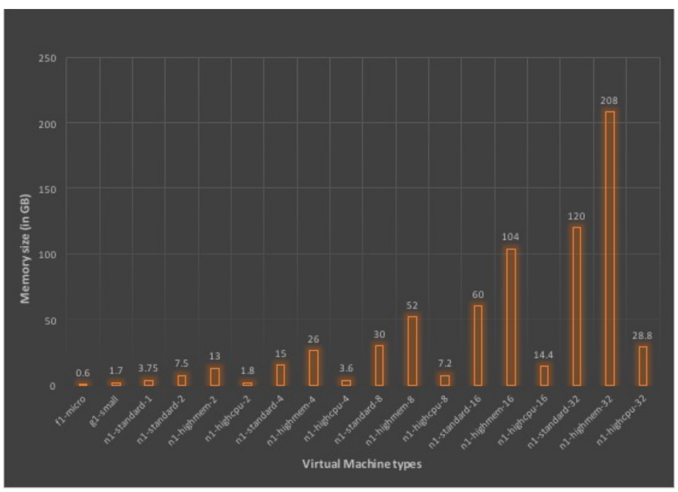

Figure 14: Memory measurements of different VM types

6. THREATS TO VALIDITY

The accuracy and validity of results that are obtained in the Experiments above can be affected by the following scenerios:

1. Improper selection of test workloads (test workload is not representative of real workloads expected at a system).
2. Not considering important system and workload parameters.
3. Not considering a proper range of levels for factors.
4. Not collecting enough monitoring data.
5.Collecting incorrect monitoring data.
6. The data sample size might affect the results as too less or too large data sample can negatively influence the accuracy of the results.
7. We have to make sure as how many runs need to be performed to get accurate results.

7. CONCLUSION

This paper presents a comparative study between different VMs to check which VM is better than the other in terms of performance, throughput, execution time, memory, availability etc.

- Top 2 VMs in terms of CPU Performance are n-standard-4 and n1-highcpu-4
- Top 2 VMs in terms of IO Performance are n1-standard-1 and n1-highmem-8
- Top 2 VMs in terms of Throughput performance are n1-highcpu-4 and n1-higmem-4
- Top 2 VMs in terms of Memory are n1-highmem-32 and n1-standard 32
- Price performance is actually a trade off between more price paid and better performance delivered
- The results show that n1-highcpu-2 and n1-highmem-2 are most cost effective with better performance
- The same instance performance can fluctuate by 50% over time
- Not all locations have equal availability and performance of instance types

8. FUTURE WORK

This paper defined the scope for measuring the performance of cloud infrastructure service and comparing the performance characteristics of different VM instance types. In the future, we can look into other parameters of the performance of VMs and then we can compare the performance of VMs cross platform like comparing VMs of Microsoft Azure and GCP or Amazon AWS. Moreover, this paper only refers to a limited amount of VMs which were a provided as a part of the trial account in Google Cloud Platform. So, we can upgrade the account and look into other VMs provided by GCP and calculate performance. We can compare performances of VMs in different zones too. Thus, we might analyze the trends of VM usage by clients as to which VM is most popular among GCP clients. The rate of change in pricing, performance over time, instance types and availability of services by location confirms that the traditional way of benchmarking a small set of instance types in a unique event is not sufficient anymore in today's world of cloud computing

9. REFERENCES

[1] Suakanto, Sinung, Suhono H. Supangkat, and Roberd Saragih. "Performance Measurement of Cloud Computing Services." *arXiv preprint arXiv:1205.1622* (2012).

[2] Mirzaei, N. "cloud computing, Institute Report, Community Grids Lab, Indiana." (2008).

[3] Mei, Yiduo, et al. "Performance measurements and analysis of network i/o applications in virtualized cloud." *2010 IEEE 3rd International Conference on Cloud Computing*. IEEE, 2010.

[4] Rimal, Bhaskar Prasad, Eunmi Choi, and Ian Lumb. "A taxonomy, survey, and issues of cloud computing ecosystems." *Cloud Computing*. Springer London, 2010. 21-46.

[5] Buyya, Rajkumar, Rajiv Ranjan, and Rodrigo N. Calheiros. "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities." *High Performance Computing & Simulation, 2009. HPCS'09. International Conference on*. IEEE, 2009.

[6] Bautista, Luis, Alain Abran, and Alain April. "Design of a performance measurement framework for cloud computing." (2012).

[7] ISO, ISO, and I. E. C. Std. "25010 Systems and software engineering--Systems and software Quality Requirements and Evaluation (SQuaRE)--System and software quality models." *ISO (International Organization for Standardization)* (2010).

[8] Singh, Brij Raj, and Rajesh Bharati. "A Review on Performance Analysis of Cloud Computing Services for Scientific Computing."

[9] Jeon, Hangoo, and Kwang-Kyu Seo. "A Framework of Cloud Service Quality Evaluation System for Activating Cloud Service Ecosystem." *Advanced Science and Technology Letters* 35 (2013): 97-100.

[10] Jeon, Hangoo, Young-Gi Min, and Kwang-Kyu Seo. "A Framework of Performance Measurement Monitoring of Cloud Service Infrastructure System for Service Activation." *International Journal of Software Engineering and Its Applications* 8.5 (2014): 127-138.

[11] Singh, Sheetal, Vipin Kumar Rathi, and Bhawna Chaudhary. "Big Data & Cloud Computing: Challenges and Opportunities." *International Journal of Innovations in Engineering and Technology (IJIET), Delhi, India* (2015): 117-125.

[12] Ostermann, Simon, et al. "A performance analysis of EC2 cloud computing services for scientific computing." *International Conference on Cloud Computing*. Springer Berlin Heidelberg, 2009.

[13] Pavlo, Andrew, et al. "A comparison of approaches to large-scale data analysis." *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*. ACM, 2009.

[14] Zaharia, Matei, et al. "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing." *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association, 2012.

[15] Engle, Cliff, et al. "Shark: fast data analysis using coarse-grained distributed memory." *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*. ACM, 2012.

[16] Li, Min, et al. "Sparkbench: a comprehensive benchmarking suite for in memory data analytic platform spark." *Proceedings of the 12th ACM International Conference on Computing Frontiers*. ACM, 2015.

[17] http://hadoop.apache.org/mapreduce/docs/current/gridmix.html.

[18] Cooper, Brian F., et al. "Benchmarking cloud serving systems with YCSB." *Proceedings of the 1st ACM symposium on Cloud computing*. ACM, 2010.

[19] Armstrong, Timothy G., et al. "LinkBench: a database benchmark based on the Facebook social graph." *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*. ACM, 2013.

[20] Wang, L., Zhan, J., Luo, C., Zhu, Y., Yang, Q., He, Y., ... & Zheng, C. (2014, February). Bigdatabench: A big data benchmark suite from internet services. In *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)* (pp. 488-499). IEEE.

[21] Sort benchmark home page. http://sortbenchmark.org/

[22] Ghazal, Ahmad, et al. "BigBench: towards an industry standard benchmark for big data analytics." *Proceedings of the 2013 ACM SIGMOD international conference on Management of data*. ACM, 2013.

[23] Ferdman, Michael, et al. Clearing the clouds: A study of emerging workloads on modern hardware. No. EPFL-REPORT-168849. 2011.

[24] Jia, Zhen, et al. "Characterizing data analysis workloads in data centers." *Workload Characterization (IISWC), 2013 IEEE International Symposium on*. IEEE, 2013.

[25] https://en.wikipedia.org/wiki/PerfKitBenchmarker

[26] https://github.com/GoogleCloudPlatform/PerfKitBenchmarker

[27] www.perfkitbenchmarker.org/

APPENDIX

APPENDIX A

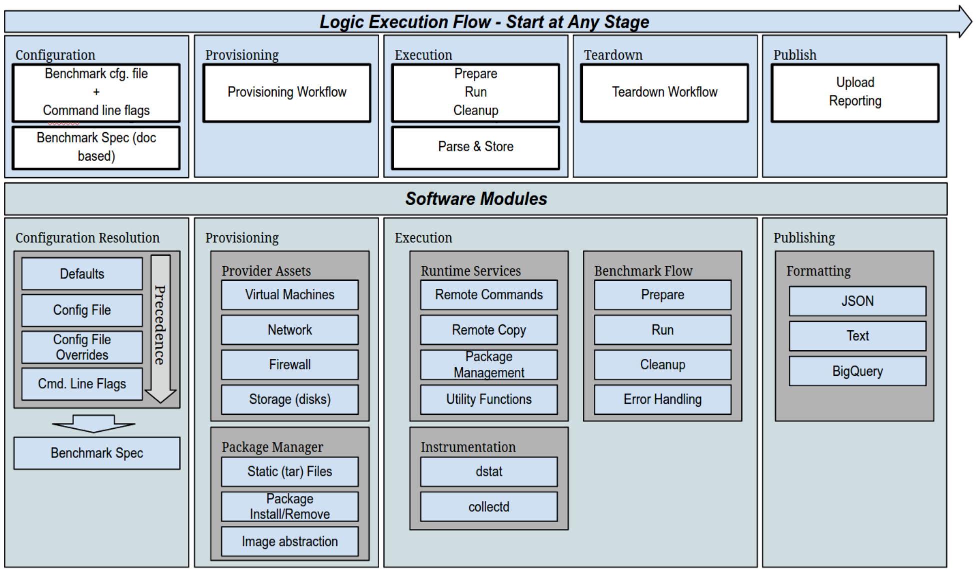

Figure A: PerfKit Benchmarker Architecture

APPENDIX B

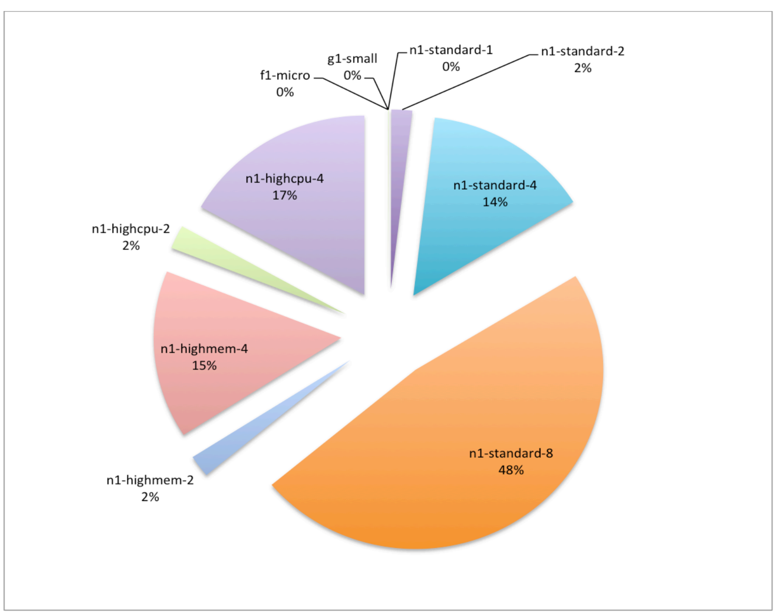

Figure D: Variance of throughput results after 4 runs

APPENDIX C

	Asia east	Asia west	Asia north	Asia south	US east	US west	US central	US north	US south	Europe east	Europe west	Europe north	Europe south
f1-micro	3	-	3	-	3	2	4	-	-	-	3	-	-
g1-small	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-standard-1	3	-	3	-	3	2	4	-	-	-	3	-	-
N1-standard-2	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-standard-4	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-standard-8	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-standard-16	3	-	3	-	1	-	1	-	-	-	3	-	-
n1-highmem-2	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highmem-4	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highmem-8	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highmem-16	-	-	2	-	3	2	4	-	-	-	-	-	-
n1-highcpu-2	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highcpu-4	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highcpu-8	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highcpu-16	3	-	3	-	3	2	4	-	-	-	3	-	-
n1-highcpu-32	Data NA	-	-	-	-	-	-	-	-	-	-	-	-
n1-highmem-32	Data NA	-	-	-	-	-	-	-	-	-	-	-	-
n1-standard-32	Data NA	-	-	-	-	-	-	-	-	-	-	-	-

Table C: Avalability of VMs in different zones (out of 5 regions)