Optimizing Cloud Infrastructure: Analyzing Parameters for Fault Detection and Correction

Anil Rajput

Professor Computer Science & Mathematics,

Govt. C. S. A. P. G. College, Sehore, MP

e-mail:dranilrajput@hotmail.com

Kshmasheel Mishra

Former Reader Computer Science,

Vikram University, Ujjain, MP

e-mail:mishra_ks@hotmail.com

Abstract

As systems have become increasingly complex, the adoption of cloud computing has surged, driven by the need to efficiently and securely manage vast amounts of data. Cloud environments now support a wide range of activities, leading to a significant rise in online operations. This shift marks a transformative era in computing, with cloud technologies expected to further expand across various sectors. However, as cloud computing becomes more widespread, both users and service providers face unique challenges related to the domains of its application. This paper provides a comprehensive analysis of common issues and the essential system conditions required for optimal performance of cloud-based systems. Utilizing a dataset of system parameters, we examine both normal and abnormal operational modes, offering valuable insights into maintaining system integrity and enhancing satisfaction for users and service providers alike.

Keywords: *Cloud Computing, Fault Correction, Data Privacy, Cloud Workloads, Anomaly Detection, Scalability, Data Processing, Statistical Analysis, Cloud Services, Real-World Datasets, Fault Analysis, Cloud Infrastructure, Operational Failures*

1.0 Introduction

Cloud solutions offer scalable resources, enabling businesses and individuals to handle demanding computational tasks without the need for extensive on-premises infrastructure [1]. By leveraging the flexibility and power of cloud computing, organizations can innovate faster, streamline operations, and focus on core activities, leaving the complexities of IT management to specialized providers [2].

Cloud computing workloads are increasing day by day due to the widespread global adoption of cloud computing technologies. Businesses, governments, and individuals are leveraging the cloud for its scalability, flexibility, and cost-efficiency, leading to a surge in data processing and storage demands. Consequently, cloud service providers are continually expanding their capabilities to accommodate the escalating workloads, ensuring reliable and secure cloud environments for users worldwide [3], [4]. Additionally, in the realm of entertainment, cloud computing powers streaming services, online gaming, and content distribution, delivering seamless and high-quality experiences to users worldwide [5].

Beyond these sectors, cloud computing is revolutionizing the way businesses operate across various industries. Manufacturing companies use cloud-based solutions for supply chain management, predictive maintenance, and IoT integration, leading to optimized production processes. Delivery of efficient services by service providers has led to established trust and a sharp rise in cloud computing services. All these factors have further led to the manifold growth of cloud users in recent years [6]. A few Data set are available as such as Google cluster data, Azure Public Dataset, Yahoo cluster data etc

Thepublicly available datasets for specific research purposes like as for failures Computer Failure Data Repository (CFDR): collecting, sharing and analyzing failure data, varying task and machine heterogeneity usage in Heterogeneous Computing Scheduling Problems (HCSP) Instances and Task Execution Time Modeling (TETM).

This paper examines several critical system parameters, emphasizing their importance in designing solutions that ensure fault-free operation and cost-effective service delivery to users.

2.0 Review of Related Work

Here is aoverview of related research worksin the cloud computing fault detection evolution and major contributions over the years by the research community:

A clear taxonomy of fault detection approaches has been established in research work [2]. Fault or failure detection techniques initially emerged from performance log analysis, with classifications based solely on historical datasets. Some researchers used cloud-adaptive anomaly detection methods to ensure high availability of cloud computing resources. Deep learning techniques have also been applied to overcome these challenges [7].

The industry has an abundance of datasets for analysis, generating huge volumes of data, while academia and the research community face a scarcity of useful data for testing new strategies and implementations. The

most widely available dataset for analysis is Google's ClusterData2011. Since its publication in 2011, many researchers have analyzed this dataset. It consists of data from approximately 12,500 machines, including scheduler requests and utilization data over 29 days. Various studies have characterized cluster resource requests, their distribution, and actual resource utilization, revealing that machines are not homogeneous. CPU utilization is less than 60%, and memory utilization is less than 50% of the total cluster size during an average one-hour window [4]. The model creates randomly fluctuating patterns to emulate dynamic environments [8]. Insights from Google cluster data include machine and workload behavior, frequency and patterns of machine maintenance events, job and task-level workload behavior, and overall cluster resource utilization. Researchers used K-means clustering to identify common job groups and conducted correlation analysis between job semantics and behavior for efficient capacity planning and system tuning [9].

Even if dataset traces are not available from the IT industry, analytical methods can extract valuable system design insights [10], [11], [12] and [13]. The low average resource utilization across clusters indicates significant scope for resource consolidation and energy-aware job scheduling to reduce power consumption. Anomaly detection can be enhanced through peer comparison [14].

3.0 Methodology

To analyze faults and the affected parameters, we downloaded a cloud workload dataset from the Kaggle platform. The dataset includes thirteen attributes and approximately 2880 values, representing both faulty and non-faulty modes during system operation. We pre-processed the dataset, cleaning it by removing redundant attributes and selecting the most valuable ones. Initially, we segregated the data based on the provided labels, 0 and 1. We found 2727 entries with label 0, representing normal system operation, while the remaining 152 entries with label 1 indicated faulty or abnormal system operation. To estimate the probabilities of different system parameters for faulty and non-faulty systems, we used the standard normal distribution, which is best suited for this context.

4.0 Experiments and Results

We studied the parameters and observed that the range of maximum and minimum values during system faults is significantly larger compared to when the system is operating normally, as shown in the graphs. By examining comparative graphs of various statistics, such as CPU Usage (MHZ), Disk Write Throughput

(KB/s), Network Received Throughput (KB/s), Memory Usage (GB), Disk Read Throughput (KB/s), and Network Transmitted Throughput (KB/s), these behaviors are clearly depicted.

The respective behaviors are summarized in Table 1 and illustrated in Figure 1 to Figure 6.

	CPU Usage [MHZ]		Disk Write Throughput [KB/s]		Network Received throughput [KB/s]		Memory Usage [GB]		Disk Read Throughput [KB/s]		Network Transmitted Throughput [KB/s]	
	NON FAULTY	FAULTY	NON FAULTY	FAULTY	NON FAULTY	FAULTY	NON FAULTY	FAULTY	NON FAULTY	FAULTY	NON FAULTY	FAULTY
Min	58.52	83.88	0.29	0.47	0.00	0.00	0.00	0.00	0.00	0.00	0.93	0.93
Mean	72.23	7546.43	1.06	7940.53	0.03	282.64	0.03	6.49	0.52	228.48	1.01	38.73
Sd	7.48	4465.90	3.67	7135.27	1.38	667.42	0.15	5.17	6.85	910.85	0.07	139.50
mid point MP	87.78	5606.22	85.94	13314.17	35.90	1781.20	2.08	10.92	80.77	4272.37	2.60	507.50
Max	117.04	11128.55	171.60	26627.87	71.80	3562.40	4.16	21.83	161.53	8544.73	4.27	1014.07
P (MP > x <MAX)	0.0188	0.4568	0.0000	0.2213	0.0000	0.0124	0.0000	0.1948	0.0000	0.0000	0.0000	0.0004
max-min	58.5	11044.7	171.3	26627.4	71.8	3562.4	4.2	21.8	161.5	8544.7	3.3	1013.1

Table 1 Analysis of cloud parameters for fault detection

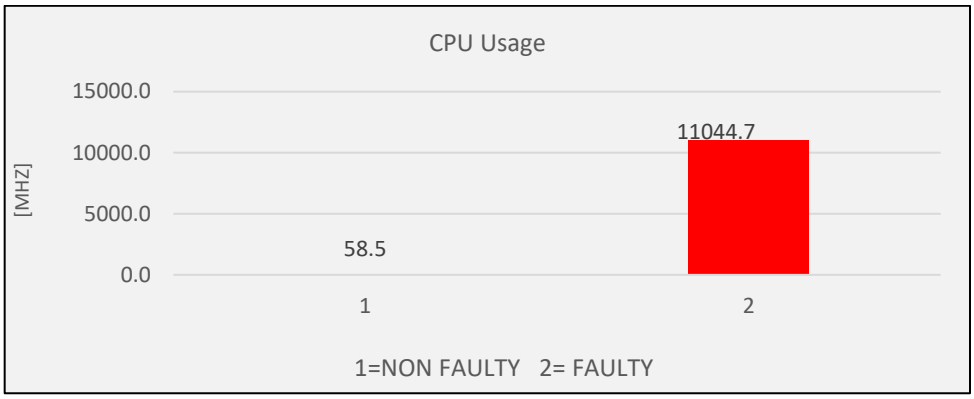

Figure 1:Graph (a), showing CPU Usage in MHZ

In graph (a), showing CPU Usage in MHZ, the stable non-faulty min and max value variation is only 58.5 MHZ, while during faults, the variation reaches 11,044 MHZ.

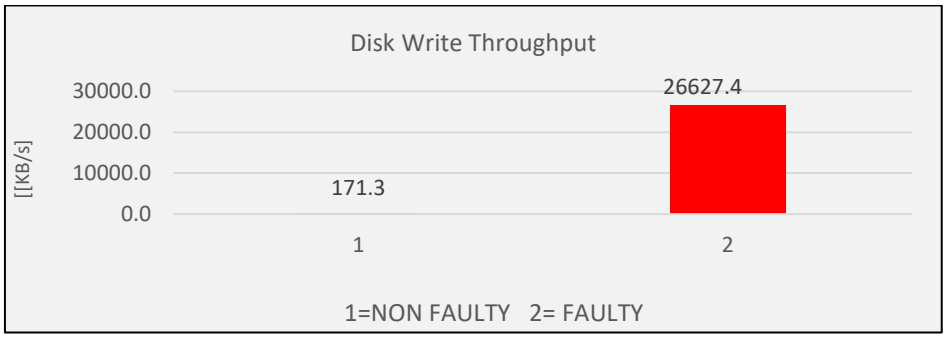

Figure 2: Graph (b), showing Disk Write Throughput in KB/s,

In graph (b), displaying Disk Write Throughput in KB/s, the non-faulty variation is 171.3 KB/s, whereas during faults, the variation soars to 26,627.4 KB/s.

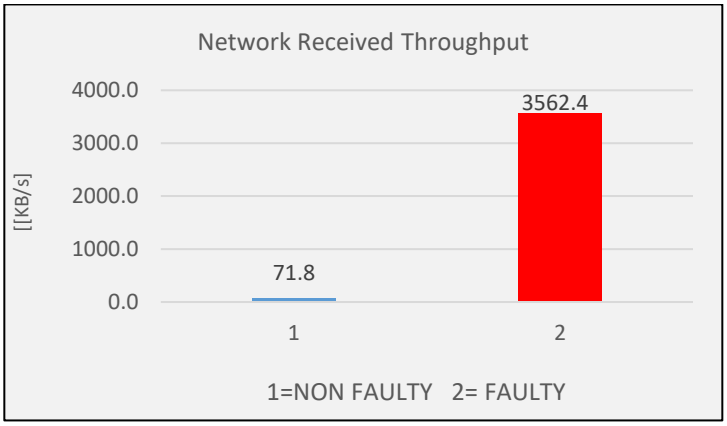

Figure 3: Graph (c) showing Network Received Throughput in KB/s

Graph (c) shows Network Received Throughput in KB/s, with a non-faulty variation of 71.8 KB/s, but during faults, the variation increases to 3,562.4 KB/s.

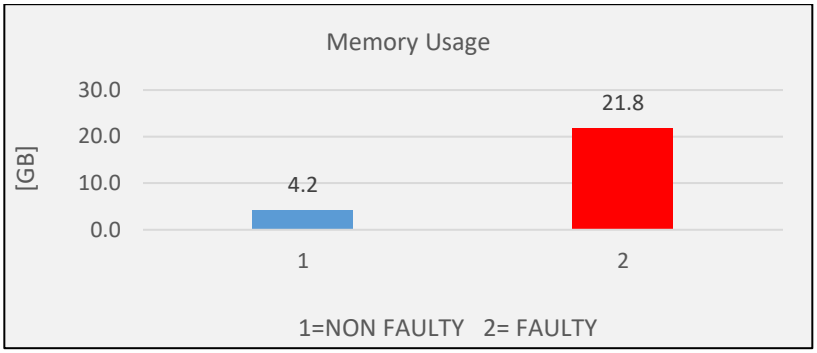

Figure 4: Graph (d) showing Memory Usage in GB

Graph (d) illustrates Memory Usage in GB, where the non-faulty variation is 4.2 GB, and during faults, it reaches 21.8 GB.

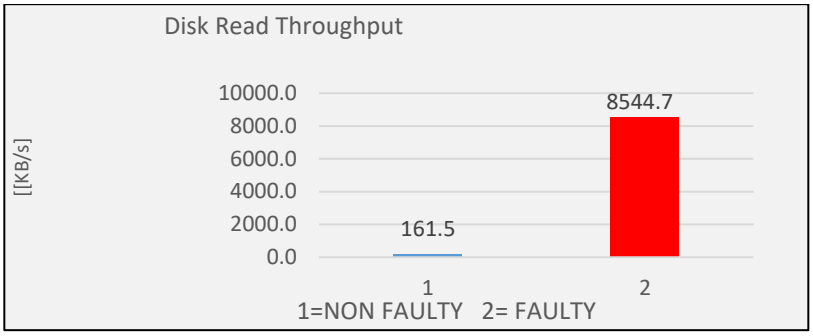

Figure 5: Graph (e) showing Disk Read Throughput in KB/s

Graph (e) presents Disk Read Throughput in KB/s, with a non-faulty variation of 161.5 KB/s, while during faults, it rises to 8,544.7 KB/s.

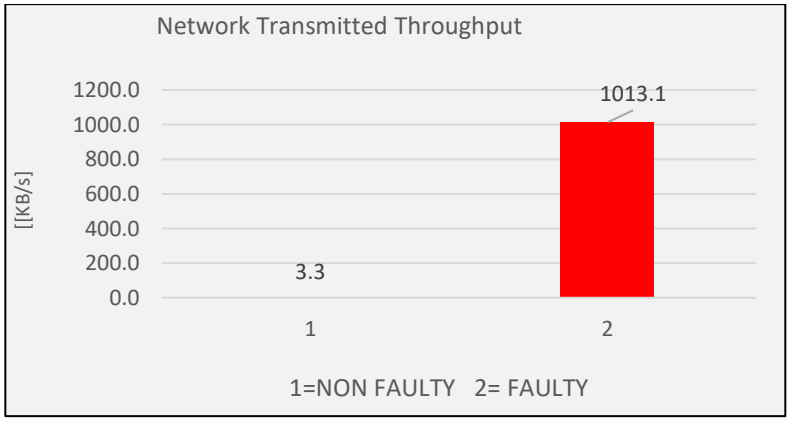

Figure 6: Graph (f) showing Network Transmitted Throughput in KB/s

Lastly, graph (f) shows Network Transmitted Throughput in KB/s, with a non-faulty variation of 3.3 KB/s, but during faults, the variation climbs to 1,013.1 KB/s.

These observations reveal that during faults, all system parameters peak abruptly to high levels, whereas, under normal operation, the parameters remain within a normal range.

We analyzed the parameters and discovered that their probability distribution is continuous. We modeled the probability of usage and impact of these attributes under normal and faulty conditions. The probability distributions reveal that the likelihood of usage parameters falling between the statistical mean and the maximum value in the non-faulty dataset is very low: 0.018 for CPU utilization and absolute zero for Disk Write Throughput, Network Received Throughput, Memory Usage, Disk Read Throughput, and Network Transmitted Throughput. Conversely, in the faulty dataset, the probability usage statistics are significantly higher (0.4568, 0.2213, 0.0124, 0.1948, 0.0000, and 0.0004, respectively).

Based on these findings, it is evident that the usage probability of CPU, memory, disk, and network parameters is much higher during fault conditions compared to normal conditions. This probabilistic estimate can serve as an indicator for detecting faults.

5.0 Conclusion

The probability of usage of important system parameters like CPU usage, Memory Usage,Disk Read/Writes, Network packets Transmitted and Received is high for faulty systems starting from the mid point towards the maximum value of the range in comparison to the non faulty systems. Thus our research support the concept of detecting faulty systems in best possible manner. This analysis facts can be directly used in design of fault or anomaly detection algorithms design

References

[1] R. Buyya, C. S. Y. S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation computer systems* , vol. 25, no. 6, pp. 599-616, 2009.

[2] D. Hilley, "Cloud computing: A taxonomy of platform and infrastructure-level offerings," in *Georgia Institute of Technology, Tech. Rep (2009): 44-45.*, 2009.

[3] G. M. Lee and N. Crespi, "Shaping future service environments with the cloud and internet of things: networking challenges and service evolution," in *Leveraging Applications of Formal Methods, Verification, and Validation: 4th International Symposium on Leveraging Applications, ISoLA 2010, Proceedings, Part I 4, pp. 399-410. Springer Berlin Heidelberg*, Heraklion, Crete, Greece, October 18-21, 2010.

[4] S. K. Barker and P. Shenoy, "Empirical evaluation of latency-sensitive application performance in the cloud," in *Proceedings of the first annual ACM SIGMM conference on Multimedia systems, pp. 35-46. 2010.*, Singapore, 2010.

[5] M. Wu, T.-J. Lu, F.-Y. Ling, J. Sun and H.-Y. Du., "Research on the architecture of Internet of Things," in *3rd international conference on advanced computer theory and engineering (ICACTE), vol. 5, pp. V5-484. IEEE, 2010.*, Honkong, 2010.

[6] L. M. Vaquero, L. Rodero-Merino and R. Buyya, "Dynamically scaling applications in the cloud," *Computer and Communication Review*, vol. 41, no. 1, p. 45–52, 2011.

[7] Baddar, A. Merlo and M. Migliardi, "Anomaly detection in computer networks: A state-of-the-art review," *Ubiquitous Computing Dependable Applications*, vol. 5, no. 4, p. 29–64, 2010.

[8] L. I. Bo-Hu, Z. Lin, W. Shi-Long, T. Fei, C. A. O. Jun-Wei, J. Xiao-dan, S. Xiao and C. Xu-dong, "Cloud manufacturing: a new service-oriented networked manufacturing model," *Computer integrated manufacturing system* , vol. 16, no. no. 01 (2010), 2010.

[9] K. A. G. G. a. J. R. H. Church, "On Delivering Embarrassingly Distributed Cloud Services," *HotNets,*, vol. 3, no. 6, pp. pp. 55-60., 2008.

[10] B. Gibson. and G. Schroeder, "The Computer Failure Data Repository (CFDR): Collecting, Sharing and Analyzing Failure Data," in *Proceedings of the 2006 ACM/IEEE Conference on Supercomputing*, 2006.

[11] Q. Guan, Z. Zhang and S. Fu, "Ensemble of Bayesian predictors and decision trees for proactive failure management in cloud computing systems," *Journal of Computing Communication*, vol. 7, no. 1, p. 52–61, 2010.

[12] H. Niedermayer, R. Holz, M.-O. Pahl and G. Carle, "On using home networks and cloud computing for a future internet of things," in *Future Internet-FIS 2009: Second Future Internet Symposium*, FIS 2009, Berlin, Germany, September 1-3, 2009.

[13] M. Reitblatt, N. Foster, J. Rexford and D. Walker, "Consistent updates for software-defined networks: Change you can believe in!," in *Proceedings of the 10th ACM workshop on hot topics in networks, pp. 1-6. 2011.*, 2011.

[14] Z. Wan, "Cloud Computing infrastructure for latency sensitive applications.," *IEEE 12th International Conference on Communication Technology, ,* pp. pp. 1399-1402, 2010.