International Journal of Computer Sciences and Engineering
Vol.**11**, Issue **3**, pp.**39-43**, March **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Paper

Dynamic Core Allocation: Enhancing Fault Tolerance and Energy Efficiency in Cloud Computing

Vikas Mongia[1]

[1]Dept. of Computer Science, Guru Nanak College, Moga, India

Author's Mail Id: vikasmongia@gmail.com

Received: 17/Feb/2023; **Accepted**: 20/Mar/2023; **Published**: 31/Mar/2023. **DOI**: https://doi.org/10.26438/ijcse/v11i3.3943

Abstract: As the prevalence of cloud computing continues to surge, cloud computing entities face the formidable challenge of meeting coordinated Service Level Agreement (SLA) understandings, particularly in terms of stability and operational efficiency, all while achieving cost and energy efficiency. This paper introduces Shadow Replication, a novel adaptation to internal failure mechanisms for cloud computing that seamlessly addresses faults at scale, concurrently limiting energy consumption and reducing its impact on costs. Energy conservation is realized by establishing dynamic cores as opposed to static cores, achieved through the deployment of cloudlets. Essentially, equivalent cores are created, with core failure metrics considering memory capacity, energy, and power consumption. If any of these parameters exceed the threshold value, the core is flagged, and progress is maintained within a shadow, assigned one for each host. The workload of a failed core is transferred to the next core within another virtual machine (VM). In the event of all cores within a VM failing, VM migration is executed. Results obtained through the proposed system exhibit improvements in indexed energy consumption, latency, cost, and fault rate.

Keywords: Shadow Replication; fault tolerance; Energy Conservation

1. Introduction

Cloud Computing has evolved into an attractive platform for an increasingly diverse array of process and data-intensive applications, offering advantages such as low-entry costs, on-demand resource provisioning and distribution, and reduced maintenance costs of internal IT infrastructure [1]. The growth trajectory of cloud computing is expected to persist, drawing attention from business and public market segments. Recent studies project an annual growth rate of 17.7 percent by 2016, positioning cloud computing as the fastest-growing segment in the software industry [2].

In its fundamental form, a cloud computing infrastructure constitutes a vast network of interconnected back-end servers hosted in a datacenter. This infrastructure is provisioned to deliver on-demand, "pay-as-you-go" services and computing resources to users through a front-end interface [3]. As the demand for cloud computing accelerates, cloud service providers (CSPs) will face the imperative to expand their foundational infrastructure to ensure expected levels of performance, reliability, and cost-effectiveness. This expansion will result in a manifold increase in the number of computing, storage, and communication components in their datacenters.

The immediate consequences of extensive datacenters include increased administrative complexity in the computing infrastructure, elevated levels of energy consumption, and a susceptibility to failures. The advantages of adopting green cloud computing are evident. With datacenters rapidly becoming a significant contributor to global energy consumption, potential savings related to energy use, CO_2 emissions, and e-waste are considerable. However, realizing these savings necessitates novel algorithmic models designed to reduce energy and power consumption and promote environmentally friendly cloud computing performance environments.

Another challenge for cloud computing, especially at scale, arises from its inherent susceptibility to failure. While the likelihood of an individual server failure is small, the sheer number of computing, storage, and communication components that could fail is substantial. At such an extensive scale, failure becomes the norm rather than an exception [4].

As the number of users entrusting their computing tasks to Cloud Service Providers (CSPs) increases, Service Level Agreements (SLAs) become a critical aspect of a sustainable cloud computing business model. In its fundamental form, an SLA is a contractual agreement between CSPs and consumers, specifying the terms and conditions under which the service is to be provided, including anticipated response time and reliability. Failure to deliver the service as stipulated in the SLA subjects the CSP to pay a penalty, resulting in lost revenue.

In addition to penalties arising from the failure to meet SLA requirements, CSPs grapple with the escalating energy costs of their large-scale data centers. Reports indicate that energy costs alone could constitute 23%–50% of the overall expenses, amounting to $30 billion worldwide [5]. This raises the question of how fault tolerance may impact power consumption and ultimately affect the environment.

Existing fault tolerance strategies rely on either time or hardware redundancy to withstand failures. The first approach, utilizing time redundancy, necessitates the re-execution of the failed task once the failure is detected [6]. Although this can be further optimized through the use of checkpointing and rollback recovery, such an approach can result in a significant delay increase, exposing CSPs to penalties for violating SLA terms and incurring high energy costs due to the re-execution of failing tasks.

The second approach exploits hardware redundancy and executes multiple instances of the same task in parallel to overcome failure, ensuring that at least one task completes successfully. This approach, extensively used to address failure in critical applications, is currently employed in cloud computing to provide fault tolerance while concealing the delay of re-execution [7]. However, this solution increases the energy consumption for a given service, which may outweigh the benefit gained from providing the service. The trade-off between profit and fault tolerance calls for new strategies to address both SLA requirements and energy considerations in dealing with failures.

2. Literature Review

In the study by [8], a measured usage model of the PSTR plot is presented, demonstrating its compatibility with most business continuous operating systems. This modular implementation model lends itself well to a comprehensive examination of recovery time limits, a metric of significant importance in complex systems. Another noteworthy work, [9], integrates the PSTR scheme with a network surveillance (NS) plot, resulting in a substantial improvement in fault coverage and achieved recovery time bounds. The NS plot employed is a recently developed scheme effective in a broad range of point-to-point networks, known as the supervisor-based NS (SNS) scheme.

In the investigation conducted by [10], the focus is on assessing the dynamic load balancing plate scheduling algorithm in conjunction with tied de-clustering as an alternative to traditional sizing plans. The research explores how this combination can dynamically respond to workload variations and disk failures.

In the study by [11], CMP memory systems are developed for server consolidation, with a focus on enhancing sharing within Virtual Machines (VMs). The memory systems presented aim to optimize shared memory accesses within VMs, minimize interference among distinct VMs, facilitate dynamic VM reassignment to processors and memory, and support content-based page sharing among VMs. The initial approach involves a tiled architecture, with each of the 64 tiles comprising a processor, private L1 caches, and an L2 bank.

Research conducted by [12] highlights the underutilization of servers in data-intensive compute clusters, suggesting opportunities for improved workload consolidation in the Hot Zone. Examination of traces from a Yahoo! Hadoop cluster revealed significant heterogeneity in the access patterns of data, providing insights for guiding energy-aware data placement strategies.Additionally, [2] introduces a simulation environment for energy-aware cloud computing data centers. This simulator, designed to capture workload distribution details, aims to represent the intricacies of energy consumption by datacenter components (servers, switches, and interfaces) as well as packet-level communication patterns in realistic configurations.

In the work by [13], the significance of communication patterns in datacenter energy consumption is emphasized, introducing a scheduling approach named DENS. The DENS method strives to harmonize the energy usage of a datacenter, individual job performance, and traffic demands, aligning energy efficiency with network awareness. Similarly, [14] introduces the Energy-Efficient Adaptive File Replication System (EAFR), incorporating three elements. This system adapts to time-varying file popularities, striking a balance between data availability and efficiency. Increased popularity of a file leads to more replications and vice versa.

Additionally, [15] provides an overview of resource provisioning and utilization patterns in data centers, proposing a macro resource management layer to coordinate between digital and physical resources. The review encompasses existing work and solutions in the field, elucidating their limitations. [16] addresses the integration of both versatility and power considerations, a departure from much of the debated work, which tends to address these aspects independently. Employing a systematic model that accounts for both power consumption and failures, the study investigates the performance of checkpoint and replication-based methods on current and future systems. It utilizes power measurements from existing systems to validate the findings.

Furthermore, [17] challenges the prevalent perception, revealing that data networks are used less intensively compared to the telephone network. Even the backbone of the Internet operates at lower capacities (10% to 15%) than the switched voice network (which averages over 30% capacity). Private line networks are even less intensively utilized (at 3% to 5%). Moreover, this scenario is likely to persist. [18] investigates the comparative performance of three high-availability data replication techniques—tied de-clustering, mirrored disks, and interleaved de-clustering—in a shared-nothing database machine environment. The study delves into several aspects, including (1) the relative performance of different techniques under normal conditions, (2) the impact of a single-node failure on system throughput and response time, (3) the influence of varying CPU speed and disk page

size on various replication strategies, and (4) the trade-off between the benefits of intra-query parallelism and the overhead associated with initiating and coordinating additional operator processes.

Moreover, [11] explores the interplay between energy management, load balancing, and replication strategies in data-intensive cluster computing. Notably, the study reveals that Chained Declustering—a replication method proposed over 20 years ago—can support highly flexible energy management plans. [19] introduces Lazy Shadowing, a versatile and power-conscious algorithm designed for achieving high levels of reliability in large-scale, failure-prone computing environments through forward progress. Lazy Shadowing assigns a "shadow" process to each primary process, executing at a reduced rate, and intelligently advances each shadow to catch up with its leading process during failure recovery. Furthermore, [20] presents two pivotal features: (1) a context-aware transmission approach that efficiently controls data transmission based on data priority, battery level, and network data rate, and (2) an architecture supporting the model of mobile interactive applications where clients and servers can autonomously interact with each other.

3. Problem Definition

In the current methodology, the absence of shadow replication poses a significant challenge. If faults occur during the execution of tasks by the existing virtual machine (VM), the progress made by the VM is lost. When a new VM takes over, it initiates work from the beginning, resulting in the loss of the entire work progress and a decrease in energy efficiency. Energy conservation aims to utilize more resources with minimal energy consumption, distinguishing it from efficient energy use, which involves using less energy for a constant service. The introduction of Shadow Replication addresses this need.

Shadow Replication efficiently preserves the work progress of the last VM. In the event of a fault, the new VM starts performing from the point where the last VM left off. This approach enhances energy efficiency and reduces time consumption. The new VM does not need to initiate work from the beginning, preventing the wastage of energy and time invested by the last VM. Fault tolerance capabilities are thereby improved, and energy consumption and time utilization are optimized.

However, the existing system faces several challenges:

- Limited fault tolerance capabilities.
- Inherent limitations in energy efficiency.
- Higher time consumption during the execution of cloudlets.
- The static cores of the existing system contribute to increased energy consumption and diminished fault tolerance capabilities.

4. Objectives of Study

The current approach to fault tolerance relies on either time redundancy or hardware redundancy to address faults. Time redundancy involves the re-execution of the failed computation after the fault is detected, and although this can be optimized using checkpoints, these solutions still introduce significant delays. In many mission-critical systems, hardware redundancy has traditionally manifested as process replication to ensure fault tolerance, avoiding delays and meeting tight deadlines. Both approaches have drawbacks; re-execution demands additional time, and replication requires additional resources, particularly energy. This compels system engineers to choose between time and hardware redundancy, with cloud computing environments largely opting for replication due to the critical importance of response time. In this paper, we introduce a novel computational model called shadow computing, which offers objective-based adaptive flexibility through dynamic execution.

- The proposed system aims to improve fault tolerance capabilities.
- The study focuses on enhancing energy efficiency as its primary objective.
- The goal is to minimize time consumption or latency in the proposed system.
- A comparative analysis between static and dynamic core systems is conducted to determine the superior approach.

ALGORITHM

Input: Process or tasks (Cloudlets)
Output: Delay, Execution time, Power consumed, Cost

1. Initialize Cloud with every datacenter, dividing each into VMs, and partitioning VMs into cores.
2. Assign cost to each virtual machine.
3. Associate energy dissipation with each VM; energy is consumed during job execution.
4. Order virtual machines based on energy consumption.
5. Partition VM into dynamic cores based on load.
6. Execute the task or process on the VM with the minimum energy consumption.
7. Calculate Dissipation_Energy using the formula:
   *Dissipation_Energy = (CPU_Per) / (P_T1 + P_T2 * CPU_Per) * (P_T1 + P_T2)*
8. In case of failure, search for another optimal VM for allocation.
9. If *Consumed_Energy > threshold_Energy*, the core has failed, and VM migration is required.
10. Copy progress to another optimal virtual machine.
11. Output delay in execution and consumed cost.

5. Results and Performance Analysis

A comparative analysis was conducted to assess the performance of static and dynamic cores. The dynamic cores demonstrated notably superior results compared to their static counterparts.

Cloudlet Size: 95000

Table 1: Comparison of Parameters for Cloudlet Size 95000

Parameter	Existing	Proposed
Energy Consumed	43.938152	38.412603
Delay	14196	12755
Expense	345	267
Rate of Fault Tolerance	68	51

The table illustrates the clear advantages of the proposed system, showing reduced energy consumption, lower delay, decreased expenses, and improved fault tolerance rate when compared to the existing system. These outcomes highlight the effectiveness of dynamic cores in enhancing overall performance metrics.

The performance evaluation was extended to cloudlet size 80000, with a comparison between the existing system and the proposed solution. The results, tabulated below, demonstrate the significant improvements achieved by the proposed system.
Cloudlet Size: 85000

Table 2: Comparison of Parameters for Cloudlet Size 85000

Parameter	Existing	Proposed
Average Energy Consumed	54.405746	14.436229
Latency	13079	10503
Cost Encountered	898	159
Fault Tolerant Rate	49	82

The table emphasizes the superior performance of the proposed system across various parameters. Notably, the proposed system exhibits significantly reduced energy consumption, lower latency, decreased costs, and an enhanced fault-tolerant rate, affirming its effectiveness in optimizing cloudlet processing at the specified size.

The performance evaluation was extended to cloudlet size 80000, with a comparison between the existing system and the proposed solution. The results, tabulated below, highlight the substantial improvements achieved by the proposed system.
Cloudlet Size: 80000

Table 3: Comparison of Parameters for Cloudlet Size 80000

Parameter	Existing	Proposed
Average Energy Consumed	51.703804	14.504681
Latency	11532	9038
Cost Encountered	793	276
Fault Tolerant Rate	49	83

The table illustrates the superior performance of the proposed system across various parameters. Significantly reduced energy consumption, lower latency, decreased costs, and an enhanced fault-tolerant rate affirm the effectiveness of the proposed system in optimizing cloudlet processing at the specified size.

6. Conclusion and Future Scope

In conclusion, this study highlights that static cores within virtual machines (VMs) contribute to increased energy consumption and costs, primarily due to core failures that result in an augmented load on alternative cores within the same VM. The adoption of dynamic cores, determined by cloudlet size, effectively addresses this challenge. Dynamic cores not only improve energy efficiency by keeping additional cores in sleep mode until load allocation is necessary but also elevate the fault tolerance rate.

Looking into the future, the research scope involves exploring the integration of a fault-tolerant strategy into the cloud resource allocation framework. Additionally, the incorporation of deep learning methodologies into the dynamic core allocation process holds promise for optimizing energy efficiency and enhancing fault tolerance mechanisms in cloud computing systems. The introduction of deep learning techniques can contribute to more adaptive and intelligent resource allocation strategies, ensuring improved performance and sustainability in dynamic cloud environments.

References

[1] B. Meroufel and G. Belalem, "Adaptive time-based coordinated checkpointing for cloud computing workflows," Scalable Comput., Vol.**15**, No.**2**, pp.**153–168, 2014.**

[2] D. Kliazovich, P. Bouvry, and S. U. Khan, "GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers," J. Supercomput., Vol.**62**, No.**3**, pp.**1263–1283, 2012.**

[3] B. Alami Milani and N. Jafari Navimipour, "A comprehensive review of the data replication techniques in the cloud environments: Major trends and future directions," J. Netw. Comput. Appl., Vol.**64**, pp.**229–238, 2016.**

[4] R. Balamanigandan, "Analyzing massive machine data maintaining in a cloud computing," Vol.**23**, No.**10**, pp.**78–81, 2013.**

[5] D. Singh, J. Singh, and A. Chhabra, "High availability of clouds: Failover strategies for cloud computing using integrated checkpointing algorithms," Proc. - Int. Conf. Commun. Syst. Netw. Technol. CSNT 2012, pp.**698–703, 2012.**

[6] Y. Zhang, Z. Zheng, and M. R. Lyu, "BFTCloud: A Byzantine Fault Tolerance framework for voluntary-resource cloud computing," Proc. - 2011 IEEE 4th Int. Conf. Cloud Comput. CLOUD 2011, no. July 2011, pp.**444–451, 2011.**

[7] P. K. Szwed, D. Marques, R. M. Buels, S. A. McKee, and M. Schulz, "SimSnap: Fast-forwarding via native execution and application-level checkpointing," Proc. - Eighth Work. Interact. between Compil. Comput. Archit. INTERACT-8 2004, pp.**65–74, 2004.**

[8] K. H. Kim and C. Subbaraman, "A modular implementation model of the Primary-Shadow TMO replication scheme and a testing approach using a real-time environment simulator," Softw. Reliab. Eng. 1998. Proceedings. Ninth Int. Symp., pp.**247–256, 1998.**

[9] K. H. Kim and C. Subbaraman, "An Integration of the Primary-Shadow TMO Replication (PSTR) Scheme with a Supervisor-based Network Surveillance Scheme and its Recovery Time Bound Analysis," Proc. SRDS '98 (IEEE CS 17th Symp. Reliab. Distrib. Syst. 1998, pp.**168–176, 1998.**

[10] Hsiao, Hui-I., and David J. DeWitt. "Chained declustering: A new availability strategy for multiprocessor database machines." University of Wisconsin-Madison Department of Computer Sciences, **1989.**

[11] M. R. Marty and M. D. Hill, "Virtual hierarchies to support server consolidation," ACM SIGARCH Comput. Archit. News, Vol.**35**, no.**2**, pp.**46, 2007.**

[12] R. T. Kaushik, "GreenHDFS: Towards An Energy-Conserving , Storage-Efficient , Hybrid Hadoop Compute Cluster," HotPower, pp.**1–9, 2010.**

[13] D. Kliazovich, P. Bouvry, and S. U. Khan, "DENS: Data center energy-efficient network-aware scheduling," Cluster Comput., Vol.**16**, No.**1**, pp.**65–75, 2013.**

[14] Y. Lin and H. Shen, "EAFR: An Energy-Efficient Adaptive File Replication System in Data-Intensive Clusters," IEEE Trans. Parallel Distrib. Syst., vol.**28**, no.4, pp.**1017–1030, 2017.**

[15] J. Liu, F. Zhao, X. Liu, and W. He, "Challenges Towards Elastic Power Management in Internet Data Centers," 2009 29th IEEE Int. Conf. Distrib. Comput. Syst. Work., pp.**65–72, 2009.**

[16] B. Mills, T. Znati, R. Melhem, K. B. Ferreira, and R. E. Grant, "Energy consumption of resilience mechanisms in large scale

systems," Proc. - 2014 22nd Euromicro Int. Conf. Parallel, Distrib. Network-Based Process. PDP 2014, pp.**528–535, 2014.**

[17] A. Odlyzko, "Data Networks are Lightly Utilized, and will Stay that Way," Rev. Netw. Econ., Vol.**2**, no.**3**, pp.**210–237, 2003.**

[18] H.-I. Hsiao and D. J. DeWitt, "A performance study of three high availability data replication strategies," [1991] Proc. First Int. Conf. Parallel Distrib. Inf. Syst., pp.**18–28, 1991.**

[20] C. S. Shih and T. K. Trieu, "Shadow phone: Context aware device replication for disaster management," Proc. - 2012 5th IEEE Int. Conf. Serv. Comput. Appl. SOCA 2012, **2012.**

