Energy Efficient and Fault Tolerance in Cloud Computing – A Literature Survey

[1] Khairunnisa

[1]Assistant Professor, Department of Computer Science
[1]Jamal Mohamed College (Autonomous), Tiruchirappalli 620 020, Tamil Nadu, India.

Abstract : Cloud computing provides various services over the Internet. It uses data centers which contain hardware and software support for storage, servers, and networks. The main reason for the popularity of Cloud computing is because it is reliable, less expensive, rapidly scalable, has broad network access and much more. A major cause for concern in cloud computing is the faults that occur either in the software or the hardware. Several techniques are available to overcome the faults. When a system is able to perform all its operations despite the occurrence of different faults, it is said to be fault tolerant. As the popularity of cloud computing increases, many organizations and institutions are turning to cloud computing. It has to be ensured that energy consumption is kept at satisfactory levels along with fault tolerance. This paper examines the various literature survey on the Fault Tolerance and Energy Efficiency in the Cloud Computing Environment using various techniques like Scheduling, Optimization and Data Mining.

IndexTerms **- Cloud Computing, Energy Efficiency, Fault Tolerant, Optimization Techniques.**

I. INTRODUCTION

As stated in the National Institute of Standards and Technology, USA. (NIST), “The term ‘Cloud Computing’ is perfect for allowing the on-demand, appropriate network permission to a mutual collection of configurable computing assets (e.g., application, networks, storage, servers, and services) that can be promptly provisioned and unconstrained by slightest administration effort or service provider communication”. The benefits of cloud computing comprise of scalability, reliability, elasticity, low cost, and great availability to the end users. The esteemed organizations like Google, Microsoft, IBM, Salesforce, and Amazon are consuming cloud to distribute their services. Among that Google has a cloistered cloud for transporting dissimilar facilities comprising of statistics, text translations, analytics and more; created on big data analytics [1]. There are thousands of servers internally associated with each other in contemporary data centers. The various applications are accommodated on these cloud servers. In addition, there are numerous computing resources accessible to the end users over the internet in the system of configurable Virtual Machines (VMs) [2]. Most of the huge data centers are cybernetic, not real. Figure 1 exhibits the virtualization system method in cloud computing.

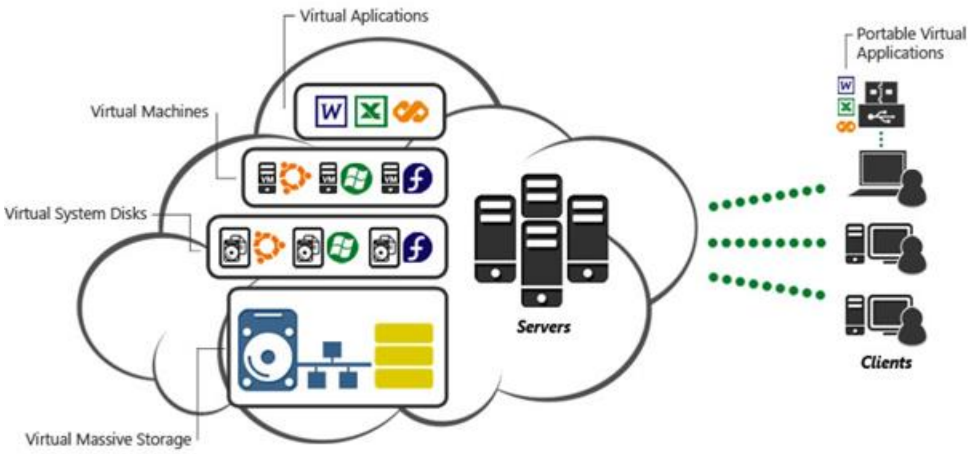

Figure 1: Virtualization in Cloud Computing [3]

Cloud computing is an emerging commercial set-up of networked and cost-effective paradigm where information can be retrieved from an online browser by customers as per their requirements. It's a paradigm for deploying on-demand availability of assets in an easily accessible manner, wherein the support is accustomed to represent set-up, structures, application, facilities, or storage. Cloud computing is a group of computing resources for clients which are shared and used by the clients via internet. Moreover, it provides essential applications for customers. It has a feasible setup which helps to assign or reassign the computing resources/services and allows the users/clients to take possession of these resources/services till required.

Currently the cloud data centers which are distributed over vast geographical locations consume lots of electricity for carrying out its multifarious functions. Cloud computing is often categorized into private, public, community and hybrid cloud. The cloud computing services are classified as Infrastructure as a carrier (IaaS), Platform as a service (PaaS), and software as a service (SaaS), and so on.

II. FAULT TOLERANCE IN CLOUD COMPUTING

Fault tolerance is building-up of a computer system which can perform by overcoming the faults. Faults are the result of errors which lead to the failure of a system. In order to minimize the failure impact on the system and application execution, failure should be anticipated and proactively handled. Various fault-tolerant algorithms and techniques can be used initially to predict the type of failure and then what immediate action should be taken? Failure can be transient, intermittent, or permanent. Here, we'll deal with the software failure and the most difficult task in designing the fault tolerant software system is to verify that it will meet its reliability requirements. Formation of such a system requires a number of models and testing in the fault environment. Overloading on VM can be detected, if the CPU utilization of a host falls below the lower threshold, all VMs have to be migrated from this host and the host has to be switched to the sleep mode in order to eliminate the idle power consumption and If the utilization exceeds the upper threshold, some VMs have to be migrated from the host to reduce the utilization in order to prevent a potential SLA violation. One of the overloading detections is Median absolute Deviation which is quite robust in nature and much sensitive [4]. Fault tolerance can be achieved by applying a set of analysis, implementing algorithms and improvement in design technique which can improve dependability. Dependability of any software system comes with reliability (continuity of the service), availability (readiness for usage), safety (avoidance of catastrophic consequences on the environment) and maintainability (how easily a failed system can be repaired).

III. ENERGY EFFICIENT IN FAULT TOLERANCE

The major objective of Energy management is to handle the current disputes and to enhance the ingestion of power is also vital owing to both ecological and financial restraints. As of date, the greatest power-consuming modules are processors, it is estimated that in forthcoming distributed systems, the power used to accomplish I/O transfers and power supply will frame a much better segment of the complete consumption of energy. Both in terms of latency and consuming energy, the comparative rate of communication is predicted to escalate intensely. Accordingly, the communication and computation loads of classic applications employed in a cloud environment and HPC will direct the large consumption and heat dissipation of power. In the future, the energy management approaches are required to exploit the energy professionally, with the intention of saving energy significantly. The profit margins of the service providers will be augmented by supporting the finest use of conserving device and underutilized resources.

The amount of computing, communication and storage components that can fail is comparatively great; therefore the prospect of failure also rises. Therefore tolerant approaches are required to manage the failures effectively whereas it ensures scalability, reliability and continuous availability of services.

To deliver the preferred Quality of Service (QoS), liability tolerance and energy competence are the major aspects to be measured in cloud computing [5].

IV. RELATED WORKS

Bala, Anju, and Inderveer Chana [6] suggested hybrid heuristic to plan the scientific workflows efficiently. Fault tolerant method has been employed through virtual machine migration technique that transfers the virtual machine spontaneously in case of task failure events owing to the overutilization of resources.

Bagui, Sikha, and Loi Tang Nguyen [7] offered a design and application of a dispersed database structure sharding to offer fault-tolerance, high availability, and scalability of large databanks in the cloud. Sharding, or horizontal partitioning, is employed to scatter the statistics between the data nodes positioned on production servers for effective management of big data on the cloud.

Amoon, Mohammed [8] recommended a structure so as to achieve a dependable platform of cloud applications. The process for picking the most appropriate fault tolerance method is accessible. A new algorithm for choosing the most consistent virtual machines for executing client's demands is obtainable.

Liu, Dong [9] projected an innovative fault-tolerant design for Real-Time Online Interactive Application (ROIA) in the cloud, which is created on the cell intersecting method. This novel design affords the redundancy to augment the sturdiness and the scalability of ROIA. It highlighted on examining three qualities of the new architecture: dynamic load balance, seamless migration across zones and server crash protection.

Chen, Gang, et al [10] offered a lightweight software fault-tolerance scheme, termed SHelp, which can meritoriously recuperate databases from numerous kinds of software bugs in the cloud structure. Through fault virtualization methods, it is suggested that 'weighted' release points the methods to efficiently subsist software failures over bypassing the damaged path.

Park, JiSu, et al [11] categorized mobile strategies into clusters along with the obtainability and the flexibility so as to accomplish dependable portable resource. As the data of mobile devices are frequently shifting, clustering should reflect the lively environment. The researchers offered a lively group-based mobile cloud computing that smears fault tolerance methods by barriers or duplication in every group.

Lu, Kuan, et al [12] expanded a service architecture that augments the resource distribution along with business strategies, in addition to a tool for optimization in SLA arbitration. The recommended work parts the contract's fault-tolerance disputes and techniques into numerous independent layers that can be hierarchically pooled into a spontaneous, parallelized, operative and competent management structure.

Wadhwa, Anu, and Anju Bala [13] elucidated Fault tolerance is approximately operative of resources deprived of any influence of liabilities happening in them. The researchers employed to decide and manage disputes like consistency and obtainability. There are various types of faults and fault tolerance methods in cloud computing have been argued.

Singh, Sukhpal, et al [14] suggested an energy proficient autonomic cloud system [Self-Optimization of Cloud Computing Energy-efficient Resources (SOCCER)] for energy efficient programming of cloud resources in data centers. The recommended work contemplates energy as a Quality of Service (QoS) parameter and mechanically augments the proficiency of cloud resources by decreasing the energy consumption.

He, Jin, et al [15] presented a novel network security architecture for cloud computing (NetSecCC) allowing for features of cloud computing. In specific, 1) offers protection to both external and internal traffics in cloud computing, 2) achieves flexible scalability pertaining to virtual middlebox load, and 3) attains fault-tolerant among virtual middlebox failure.

Sampaio, Altino M., and Jorge G. Barbosa [16] exhibited two advanced fault-tolerance procedures are associated by obtainability of services to clients and energy rates to the Cloud providers.

Ding, Yongsheng, Guangshun Yao, and Kuangrong Hao [17] recommended an offline fault-tolerant elastic scheduling algorithm for workflow in Cloud systems (FTESW). After examining the restrains of primary-backup scheduling in Cloud systems produced by the reliance among responsibilities in the succumbed workflow, a flexible tool in the framework of fault tolerance is aimed lively to regulate the resource provisioning based on the resource demand by implementing the tools of resource migration. The FTESW is suggested to accomplish both high resource and fault tolerance utilization for workflow in Cloud systems. To confirm the efficiency of the suggested FTESW, sequences of virtual reality experiments are directed through arbitrarily produced workflows and real-world workflows.

Jhawar, Ravi, and Vincenzo Piuri [18] the objective of this research is to enhance an appreciative of the environment, statistics, and type of faults that perform in typical cloud computing set-ups, by what means these faults disturb user applications, and how faults can be managed in a competent and economical way.

Poola, Deepak, et al [19] directed to classify and organize the various fault-tolerant methods and deliver a comprehensive vision of fault-tolerance in the workflow domain for disseminated settings. The authors planned various scheduling methodologies and changed workflow management systems. It also gives a comprehensive vision into different fault-tolerant approaches and their modifications.

Mohammed, Bashir, et al [20] suggested a more healthy and dependable design using optimal checkpointing approach to confirm high system obtainability and condensed system job service to finish in time. By employing the pass rates and virtualized mechanisms, the recommended smart failover strategy (SFS) scheme deploys modules like a cloud load balancer, cloud controller, cloud fault manager, and a selection mechanism, delivering fault tolerance via optimized selection, severance, and checkpointing.

Arabnejad, Hamid, et al [21] offered a fuzzy job distributor (load balancer) for fault tolerance management to moderate the stages of management intricacy for the clients. The recommended method wishes to lessen the prospect of fault rates in the structure by fair dissemination of client job requests among accessible resources. In our self-adaptive method, the system accomplishes strange circumstances that influence leads to the catastrophe by dispensing the incoming job demand based on the consistency of handling nodes, i.e., virtual machines (VMs).

Marahatta, Avinab, et al [22] established a vibrant task project and arrangement scheme, namely the energy-aware fault-tolerant dynamic scheduling scheme (EFDTS), to co-ordinately augment resource utilization and energy consumption with a fault tolerant mechanism. In the task assignment system, a task arrangement process is established to divide the imminent responsibilities into different periods and then distribute them to the most appropriate virtual machines based on their modules to moderate the malicious response time while allowing for energy consumption.

Kaur, Kamaljit, Navdeep Kaur, and Kuljit Kaur [23] intended to exhibit a setting and load alert procedure for competent task scheduling by altered genetic algorithm branded as a family genetic algorithm. Based on investigations of the user features, user requests are rewarded by the accurate form of resource. Such an organization supports to attain competent scheduling and amended load balancing and will substantiate the value for the forthcoming of the cloud.

Oma, Ryuji, et al [24] recommended a fault-tolerant TBFC (FTBFC) model. Here, we recommended non-replication and replication FTBFC simulations to make fog nodes fault-tolerant. In the non-replication FTBFC model, another operational fog node takes over a faulty fog node.

Latiff, Muhammad Shafie Abd, Syed Hamid Hussain Madni, and Mohammed Abdullahi [25] recommended a dynamic clustering league championship algorithm (DCLCA) scheduling method for fault tolerance responsiveness to state the cloud task implementation which would replicate on the present obtainable resources and lessen the untimely failure of independent tasks.

Xie, Guoqi, et al [26] intended to augment energy-efficient fault-tolerant scheduling for a consistent parallel application on assorted distributed embedded systems, where the parallel application is pronounced by a directed acyclic graph (DAG). Energy-Efficient scheduling with a reliability goal (ESRG) algorithm is exhibited to lessen the energy consumption while sustaining the dependability goal for the parallel application. The recommended energy-efficient fault-tolerant scheduling with a reliability goal (EFSRG) algorithm to lessen the energy consumption while sustaining the consistency goal based on an active duplication structure.

Kushwah, Virendra Singh, Sandip Kumar Goyal, and Avinash Sharma [27] highlighted an investigation on meta-heuristic techniques for fault tolerance under a cloud computing environment. The indication behind using meta-heuristics is to upsurge the effectiveness and lessen the calculation time to attain the job done and, in our case,, meta-heuristics are to be measured the robust solution of outcome the accurate combinations of resources and tasks to reduce the cut costs, computational expenses and offer quality services for clients. Fault tolerance plays a vital role in safeguarding high serviceability and solid quality in the cloud.

Paulraj, Getzi Jeba Leelipushpam, et al [28] proposed a fault-tolerant system. The fault-tolerant system examines the strength of each host using fault tree-based investigation. Virtual machines are transferred from the corrupt host. The suggested approach has been investigated with numerous failure cases, and its amount is evidenced to be the best associated with the state-of-the-art approaches in works of literature.

Wu, Hao, et al [29] offered a novel fault tolerant framework named Fault Tolerance Algorithm using Selective Mirrored Tasks Method (FAUSIT) for the fault tolerance of successively a big data application on cloud. First, it offers wide-ranging theoretical investigations on how to advance the performance of fault tolerance for running a single task on a processor. Second, considering the balance between the parallelism and the topology of an application, it exhibits a discriminating mirrored task technique. In conclusion, by retaining the selective mirrored task method, the FAUSIT is architecture to augment the fault tolerance for DAG based application and integrates two vital objects: reducing the makespan and the calculation cost.

Eischer, Michael, and Tobias Distler [30] presented a Byzantine fault tolerant (BFT) scheme architecture that is capable of profit from further hardware resources. To attain these possessions while still provided that strong consistency, Omada first parallelizes agreement into multiple groups and then performs the requests managed by different groups in a deterministic order.

Yan, Hui, et al [31] introduced the indecision to our task runtime approximation method and suggested a fault-tolerant task allocation mechanism that intentionally uses two fault tolerant task scheduling models while the indecision is measured. In addition, the researchers employ the intersecting mechanism to advance the source utilization of cloud. Based on the two devices, researchers recommended a novel Dynamic Fault-Tolerant Elastic scheduling algorithm-DEFT for scheduling real-time tasks in the cloud where the scheme presentation volatility should be measured.

Sivagami, V. M., and K. S. Easwarakumar [32] recommended a new Dynamic Fault-Tolerant VM Migration (DFTM) for the enforcement of cloud data center organization consistency through progressive recovery mechanism of VN (Virtual Network) loads. Integer Linear Programming model is realized for examining the path traffic which reflects all associated statistical issues to select the best VM via the optimal path. Formerly, Alternative Switch Identification Algorithm regulates the new VM migration and routing is proficient.

Wu, Binghong, et al [33] suggested a system that pools an original intrinsic-plasticity-inspired rescheduling execution model (IPIREM) with an enhanced NSGA-III with knee points. The IPIREM delivers a comparatively steady assessment deprived of an approximation of the failure distribution when temporary failures happen. Based on the assessment result of IPIREM, the multi-objective optimization problem is explained by the procedure of an amended NSGA-III, which quickens merging and stimulates the variety.

D'Angelo, Gabriele, Stefano Ferretti, and Moreno Marzolla [34] offered FT-GAIA, a software-based fault-tolerant parallel and dispersed simulation middleware. FT-GAIA has being considered to reliably manage Parallel And Distributed Simulation (PADS) representations, which are required to appropriately pretend and analyze intricate systems ascending in any form of scientific or engineering field. PADS proceed the benefit of numerous implementation units tracks in multicore processors, collection of workstations or HPC systems.

Boru, Dejene, et al [35] planned data replication in cloud computing data centers. In contrast, the other methods available, reflect both energy competence and bandwidth depletion of the system. In addition, enhanced quality of service QoS attained as a result of the abridged communication interruptions.

Dong, Ziqian, Ning Liu, and Roberto Rojas-Cessa [36] announced an ideal of task scheduling for a cloud-computing data center to examine energy-efficient task scheduling. The researchers articulate the tasks of tasks to servers as an integer-programming problem with the disinterested of decreasing the energy disbursed by the servers of the data center. The authors evidenced that the practice of a greedy task scheduler limits the control service time whilst lessening the number of active servers.

Hosseinimotlagh, Seyedmehdi, Farshad Khunjush, and Rashidaldin Samadzadeh [37] announced an optimal utilization level of a host to perform a definite amount of commands to lessen the energy consumption of the host. In addition, the research recommended a virtual machine (VM) scheduling algorithm based on the incomparable utilization level to come up with the best energy consumption while meeting a given QoS.

Saab, Salwa Adriana, et al [38] offered an accurate model that signifies this energy consumption optimization problem. The authors proposed an algorithm to vigorously solve the issue while captivating the safety measures into account. The authors also proposed the free sequence protocol (FSP) that permits for the vibrant performance of apps along with their call graph.

Tchana, Alain, et al [39] expressed a solution familiarising dynamic software consolidation. Software consolidation makes it conceivable to vigorously collocate numerous software applications on the same VM to decrease the number of VMs used. This method can be pooled with VM consolidation which collocates several VMs on a condensed quantity of physical machines. Software consolidation can be practiced in a private cloud to decrease power consumption, or by a client of a public cloud to lessen the quantity of VMs used, accordingly in reducing the costs.

Li, Hongjian, et al [40] established a dynamic energy-efficient virtual machine (VM) migration and association process based on a multi-resource energy efficient process. It can decrease the energy consumption with Quality of Service guarantee. In our algorithm, it has calculated with a method of twin threshold with multi-resource application to prompt the migration of VMs. The Improved Particle Swarm Optimization method is introduced into the association of VMs to evade sinking into local optima which are a corporate flaw in traditional heuristic algorithms.

Jiang, Dingde, et al [41] projected a robust routing algorithm to influence the sophisticated network energy competence, which is based on optimization problem. To achieve the extremely energy-efficient direction-finding in energy-efficient systems for cloud computing, the bond of low utilization is revolved into the napping state to protect the net energy. Simultaneously, the low link traffic is combined to the link with great consumption to augment the link utilization and to nap the associations as many as probable. The authors also offered an elevated link sleeping technique to make the best use of the quantity of the napping links.

Wang, Xiaoli, Yuping Wang, and Yue Cui [42] proposed a new energy-aware multi-job scheduling model based on MapReduce in this paper. In the recommended model, first, the discrepancy of energy ingestion with the presentation of servers is taken into account; second, meanwhile network bandwidth is a comparatively restricted source in cloud computing, 100%data vicinities is guaranteed; last but not least, in view of that task-scheduling approaches be subject to unswervingly on data placement policies, It express the problem as a digit bi-level programming method.

Salimian, Leili, Faramarz Safi Esfahani, and Mohammad-Hossein Nadimi-Shahraki [43] proposed an adaptive fuzzy threshold-based algorithm to identify overloaded and under-loaded hosts. The recommended algorithm creates procedures vigorously and apprises affiliation tasks to acclimatize the modifications in workload. It is authenticated with real-world PlanetLab workload.

Singh, Sukhpal, and Inderveer Chana [44] presented fuzzy logic based energy-aware autonomic resource scheduling structure for cloud for energy-efficient scheduling of cloud calculating resources in data centers.

Dashti, Seyed Ebrahim, and Amir Masoud Rahmani [45] suggested a hieratical design to fulfill both providers' and consumers' requirements in these tools. The authors also design a new service in the PaaS layer for forecast consumer tasks. In the providers' viewpoint, inconsistency between the requirement of a physical machine and user demands in cloud hints to complications like energy-performance trade-off and large power consumption with the aim of profits are reduced. The authors proposed to adapt Particle Swarm Optimisation to reallocate migrated virtual machines in the overloaded host.

Teng, Fei, et al [46] proposed a DVFS-based heuristic TRP-FS to combine virtual groups on physical servers to protect energy whereas guarantee job SLAs. To substantiate the most competent frequency that reduces the energy consumption, and the upper bound of energy saving through DVFS methods.

Fiandrino, Claudio, et al [47] proposed a structure of new-fangled metrics capable of evaluating the execution and energy efficiency of cloud computing communication systems, procedures, and rules. The recommended metrics have been assessed for the most common data center designs comprising fat tree three-tier, DCell, BCube, and Hypercube.

Duan, Hancong, et al [48] suggested a novel scheduling method named PreAntPolicy that comprises of a forecast model based on fractal mathematics and a scheduler on the source of an amended ant colony algorithm. The forecast method regulates whether to generate the performance of the scheduler by virtue of load trend forecast, and the scheduler is accountable for source forecast while reducing energy consumption below the principle of certifying the Quality-of-Service (QoS).

You, Changsheng, et al [49] recommended a sub-optimal resource-allocation algorithm is projected to decrease the calculation intricacy for calculating the threshold. Next, The authors contemplate the orthogonal frequency-division multiple access (OFDMA) Mobile-edge computation offloading (MECO) system, for which the optimal resource distribution is expressed as a mixed-integer problem. To explain the puzzling problem and describe its scheme construction, a low-complexity sub-optimal algorithm is suggested by transmuting the OFDMA issue to its time-division multiple access (TDMA) counterpart.

Sharkh, Mohamed Abu, and Abdallah Shami [50] presented a novel measured optimization model to explain the problem of energy competence in a cloud data center. Then, the scholars recommend an answer based on VM migration that tackles this problem and minimizes energy efficiency in comparison to other common solutions. This solution includes a novel proposed technique to be integrated into any consolidation-based energy efficiency solution. This technique depends on dynamic idleness prediction (DIP) using machine learning classifiers.

Gai, Keke, Meikang Qiu, and Hui Zhao [51] dedicated on the energy-saving problem and reflect the energy wastes when jobs are allocated to distant cloud servers or heterogeneous core processors. The results aim to condense the total energy cost of the mobile heterogeneous embedded systems by a novel task to heterogeneous cores and mobile clouds. The suggested model is named as Energy-Aware Heterogeneous Cloud Management (EA-HCM) model and the major algorithm is Heterogeneous Task Assignment Algorithm (HTA2).

Vafamehr, Ali, and Mohammad E. Khodayar [52] offered the criteria, assets, and simulations for energy-aware cloud computing practices and foresee a market organization that discourses the influence of the eminence and value of energy supply on the value and cost of cloud computing services. Energy management practices for cloud providers at the macro and micro stages to progress the rate and consistency of cloud services are highlighted.

Oma, Ryuji, et al [53] suggested a tree-based fog computing approach to organizing procedures and numbers to fog nodes to facilitate the total electric energy consumption of nodes can be condensed in the IoT.

Juarez, Fredy, Jorge Ejarque, and Rosa M. Badia [54] recommended a virtual dynamic scheduling system to perform professionally task-based applications on disseminated computing stages with the intention of decrease energy consumption. The anticipated algorithm reduces a multi-objective function which pools the energy-consumption and implementation time along with the energy-performance importance factor delivered by the resource provider or user, also taking into account sequence dependent setup times between tasks, setup times and down times for virtual machines (VM) and energy profiles for various architectures.

Oma, Ryuji, et al [55] propose a tree-based fog computing model to distribute processes and data to servers and fog nodes so that the total electric energy In the evaluation, we show the total electric energy consumption of nodes in the tree-based model is slighter than the cloud computing model. consumption of nodes can be reduced in the IoT.

Reddy, V. Dinesh, G. R. Gangadharan, and G. Subrahmanya VRK Rao [56] proposed an improved distinct particle swarm optimization based on the distinctive particle swarm optimization for the original location of virtual machines and a novel virtual machine selection algorithm for enhancing the present distribution based on bandwidth utilization, memory utilization, and scope of the virtual machine.

Sundararaj, Vinu [57] propose a hybrid Queue Ant Colony-Artificial Bee Colony Optimization (Ant-Bee) algorithm for the optimal task of tasks in MCC environment. The anticipated algorithm functions on a two-way MCC model with the discharging system, that reflects both the ‘cloudlets’ and the public ‘cloud’. The ‘cloud’ and the ‘cloudlets’ are calculated on the source of the queue method for the approximation of customers waiting time in the restriction of resources.

Li, X. X., F. Z. He and W. D. Li [58] emphasized an innovative cloud-terminal-based cyber-physical system (CTCPS) architecture to aid energy efficient machining process optimization. The CTCPS comprises of four stages: control level, machine level, decision support level, and data level. The machine level and control level are poised of all brands of workstations linked to the machining process and generally accountable for observing machines and controlling machines to implement the optimal solutions.

Sun, Gang, et al [59] pronounced a low-complexity heuristic algorithm named energy-efficient online SFC request orchestration across multiple domains (EE-SFCO-MD) for near-optimally answering the stated problem. Lastly, the scholars performed simulation experiments to assess the execution of our algorithm.

Kumar, Sunil, and Mala Kalra [60] dedicated an emerging eco-friendly and energy-efficient scheduling algorithm. Various techniques for energy efficient task scheduling have already been recommended, however, there is numerous room for enhancements. The researchers exhibited a mixture of Genetic Algorithm and Artificial Bee Colony-based approach along with DVFS to complete energy-efficient task scheduling.

IV. RESULTS AND DISCUSSION

The foremost trials regarding fault tolerant and energy efficiency of Cloud applications is linked to its improvement process, explicitly going from optimized implementation, lightweight design, and smart distribution by choosing targeted hardware. Even if, central attention of developers is on a great presentation, some of the energy efficiency objectives are moderately reliable with execution goals, as scalability and short execution time characteristically linking to low energy utilisation. On the other hand, the great challenge is the runtime, where designers tend to freight the whole thing in memory and retain the system up by 24/7 with the aim of attaining the speed of machines that aid high claim peaks. Such policies hardly submit with energy efficiency goals, in cloud environment. However, competence can still be enhanced over close incorporation of a working system and assert by eradicating redundant structures. In addition, it also comprises of fine-tuning the size of virtual machines or offering sandboxes or Linux containers for applications with the aim to decrease the overhead. In conclusion, the running of Cloud appliances in an energy-efficient and fault tolerant way needs communication between applications and other areas.

V. CONCLUSION

At present, Cloud computing plays a major role in offering computing services to various sizes of businesses. In cloud computing, Fault tolerance is one of the main challenges and critical issue. It is concerned with all the techniques which are highly necessary to allow a system to tolerate faults occurred in hardware or software remaining in the system after its expansion. Similarly, in recent years, energy efficiency has evolved as one of the most critical research issues in the field of cloud computing. Because of the rising need of cloud computing, the energy consumption of data centers becomes a major problem. Cloud computing consumes a huge amount of electrical power which results in high operational costs and emission of carbon dioxide**.** Fault tolerance and energy efficiency are the main parameters to be considered by cloud service providers in order to provide the expected Quality of Service (QoS) to clients while gaining profit at the same time. This paper, work done by many many researchers in the field of fault-tolerance and energy efficiency for cloud applications.

REFERENCES

[1] Fegaras, Leonidas. "Compile-Time Query Optimization for Big Data Analytics." *Open Journal of Big Data (OJBD)* 5.1 (2019): 35-61.

[2] Naik, Ketaki, G. Meera Gandhi, and S. H. Patil. "Multiobjective virtual machine selection for task scheduling in cloud computing." *Computational Intelligence: Theories, Applications and Future Directions-Volume I*. Springer, Singapore, 2019. 319-331.

[3] Koumaras, Harilaos, et al. "Virtualization evolution: from IT infrastructure abstraction of cloud computing to virtualization of network functions." Web Services: Concepts, Methodologies, Tools, and Applications. IGI Global, 2019. 1762-1789.
[4] V.M.Sivagami and Dr.K.S.EaswaraKumar, "Survey on Fault Tolerance Techniques in Cloud Computing Environment," International Journal of Scientific Engineering and Applied Science (IJSEAS) - Volume-1, Issue-9, December 2015.
[5] Ambica Sood and Kamaljit Kaur, "A Coordinated Approach for Energy-Aware Fault Tolerance in Cloud Computing", International Journal of Advanced Research in Computer Science (IJARCS), Volume 8, No.5, June 2017.
[6] Bala, Anju, and Inderveer Chana. "Autonomic fault tolerant scheduling approach for scientific workflows in Cloud computing." *Concurrent Engineering* 23.1 (2015): 27-39.
[7] Bagui, Sikha, and Loi Tang Nguyen. "Database sharding: to provide fault tolerance and scalability of big data on the cloud." International Journal of Cloud Applications and Computing (IJCAC) 5.2 (2015): 36-52.
[8] Amoon, Mohammed. "A framework for providing a hybrid fault tolerance in cloud computing." *2015 Science and Information Conference (SAI)*. IEEE, 2015.
[9] Liu, Dong. "A fault-tolerant architecture for ROIA in cloud." *Journal of Ambient Intelligence and Humanized Computing*6.5 (2015): 587-595.
[10] Chen, Gang, et al. "A lightweight software fault-tolerance system in the cloud environment." *Concurrency and Computation: Practice and Experience* 27.12 (2015): 2982-2998.
[11] Park, JiSu, et al. "Dynamic group-based fault tolerance technique for reliable resource management in mobile cloud computing." *Concurrency and Computation: Practice and Experience* 28.10 (2016): 2756-2769.
[12] Lu, Kuan, et al. "Fault-tolerant service level agreement lifecycle management in clouds using actor system." *Future Generation Computer Systems* 54 (2016): 247-259.
[13] Wadhwa, Anu, and Anju Bala. "Preventing faults: Fault monitoring and proactive fault tolerance in cloud computing." *Proceedings of International Conference on ICT for Sustainable Development*. Springer, Singapore, 2016.
[14] Singh, Sukhpal, et al. "SOCCER: self-optimization of energy-efficient cloud resources." *Cluster Computing* 19.4 (2016): 1787-1800.
[15] He, Jin, et al. "NetSecCC: A scalable and fault-tolerant architecture for cloud computing security." *Peer-to-Peer Networking and Applications* 9.1 (2016): 67-81.
[16] Sampaio, Altino M., and Jorge G. Barbosa. "A comparative cost study of fault-tolerant techniques for availability on the cloud." *International Symposium on Ambient Intelligence*. Springer, Cham, 2017.
[17] Ding, Yongsheng, Guangshun Yao, and Kuangrong Hao. "Fault-tolerant elastic scheduling algorithm for workflow in cloud systems." *Information Sciences* 393 (2017): 47-65.
[18] Jhawar, Ravi, and Vincenzo Piuri. "Fault tolerance and resilience in cloud computing environments." *Computer and information security handbook*. Morgan Kaufmann, 2017. 165-181.
[19] Poola, Deepak, et al. "A taxonomy and survey of fault-tolerant workflow management systems in cloud and distributed computing environments." *Software Architecture for Big Data and the Cloud*. Morgan Kaufmann, 2017. 285-320.
[20] Mohammed, Bashir, et al. "Failover strategy for fault tolerance in cloud computing environment." *Software: Practice and Experience* 47.9 (2017): 1243-1274.
[21] Arabnejad, Hamid, et al. "A fuzzy load balancer for adaptive fault tolerance management in cloud platforms." *European Conference on Service-Oriented and Cloud Computing*. Springer, Cham, 2017.
[22] Marahatta, Avinab, et al. "Energy-aware fault-tolerant dynamic task scheduling scheme for virtualized cloud data centers." *Mobile Networks and Applications* (2018): 1-15.
[23] Kaur, Kamaljit, Navdeep Kaur, and Kuljit Kaur. "A novel context and load-aware family genetic algorithm based task scheduling in cloud computing." *Data Engineering and Intelligent Computing*. Springer, Singapore, 2018. 521-531.
[24] Oma, Ryuji, et al. "Fault-tolerant fog computing models in the IoT." *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. Springer, Cham, 2018.
[25] Latiff, Muhammad Shafie Abd, Syed Hamid Hussain Madni, and Mohammed Abdullahi. "Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm." *Neural Computing and Applications* 29.1 (2018): 279-293.
[26] Xie, Guoqi, et al. "Energy-efficient fault-tolerant scheduling of reliable parallel applications on heterogeneous distributed embedded systems." *IEEE Transactions on Sustainable Computing* 3.3 (2018): 167-181.
[27] Kushwah, Virendra Singh, Sandip Kumar Goyal, and Avinash Sharma. "Meta-Heuristic Techniques Study for Fault Tolerance in Cloud Computing Environment: A Survey Work." *Soft Computing: Theories and Applications*. Springer, Singapore, 2019. 1-11.
[28] Paulraj, Getzi Jeba Leelipushpam, et al. "Fault-Tolerant Cloud System Based on Fault Tree Analysis." *Advances in Big Data and Cloud Computing*. Springer, Singapore, 2019. 1-9.
[29] Wu, Hao, et al. "A Selective Mirrored Task Based Fault Tolerance Mechanism for Big Data Application Using Cloud." *Wireless Communications and Mobile Computing* 2019 (2019).
[30] Eischer, Michael, and Tobias Distler. "Scalable Byzantine fault-tolerant state-machine replication on heterogeneous servers." *Computing* 101.2 (2019): 97-118.
[31] Yan, Hui, et al. "DEFT: Dynamic Fault-Tolerant Elastic scheduling for tasks with uncertain runtime in cloud." *Information Sciences* 477 (2019): 30-46.
[32] Sivagami, V. M., and K. S. Easwarakumar. "An Improved Dynamic Fault Tolerant Management Algorithm during VM migration in Cloud Data Center." *Future Generation Computer Systems* 98 (2019): 35-43.
[33] Wu, Binghong, et al. "An integrated algorithm for multi-agent fault-tolerant scheduling based on MOEA." *Future Generation Computer Systems* 94 (2019): 51-61.
[34] D'Angelo, Gabriele, Stefano Ferretti, and Moreno Marzolla. "Fault tolerant adaptive parallel and distributed simulation through functional replication." *Simulation Modelling Practice and Theory* 93 (2019): 192-207.
[35] Boru, Dejene, et al. "Energy-efficient data replication in cloud computing datacenters." *Cluster computing* 18.1 (2015): 385-402.

[36] Dong, Ziqian, Ning Liu, and Roberto Rojas-Cessa. "Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers." *Journal of Cloud Computing*4.1 (2015): 5.

[37] Hosseinimotlagh, Seyedmehdi, Farshad Khunjush, and Rashidaldin Samadzadeh. "SEATS: smart energy-aware task scheduling in real-time cloud computing." *The Journal of Supercomputing* 71.1 (2015): 45-66.

[38] Saab, Salwa Adriana, et al. "Partial mobile application offloading to the cloud for energy-efficiency with security measures." *Sustainable Computing: Informatics and Systems*8 (2015): 38-46.

[39] Tchana, Alain, et al. "Software consolidation as an efficient energy and cost saving solution for a SaaS/PaaS cloud model." *European Conference on Parallel Processing*. Springer, Berlin, Heidelberg, 2015.

[40] Li, Hongjian, et al. "Energy-efficient migration and consolidation algorithm of virtual machines in data centers for cloud computing." *Computing* 98.3 (2016): 303-317.

[41] Jiang, Dingde, et al. "An optimization-based robust routing algorithm to energy-efficient networks for cloud computing." *Telecommunication Systems* 63.1 (2016): 89-98.

[42] Wang, Xiaoli, Yuping Wang, and Yue Cui. "An energy-aware bi-level optimization model for multi-job scheduling problems under cloud computing." *Soft Computing* 20.1 (2016): 303-317.

[43] Salimian, Leili, Faramarz Safi Esfahani, and Mohammad-Hossein Nadimi-Shahraki. "An adaptive fuzzy threshold-based approach for energy and performance efficient consolidation of virtual machines." *Computing* 98.6 (2016): 641-660.

[44] Singh, Sukhpal, and Inderveer Chana. "EARTH: Energy-aware autonomic resource scheduling in cloud computing." *Journal of Intelligent & Fuzzy Systems* 30.3 (2016): 1581-1600.

[45] Dashti, Seyed Ebrahim, and Amir Masoud Rahmani. "Dynamic VMs placement for energy efficiency by PSO in cloud computing." *Journal of Experimental & Theoretical Artificial Intelligence* 28.1-2 (2016): 97-112.

[46] Teng, Fei, et al. "Energy efficiency of VM consolidation in IaaS clouds." *The Journal of Supercomputing* 73.2 (2017): 782-809.

[47] Fiandrino, Claudio, et al. "Performance and energy efficiency metrics for communication systems of cloud computing data centers." *IEEE Transactions on Cloud Computing* 5.4 (2017): 738-750.

[48] Duan, Hancong, et al. "Energy-aware scheduling of virtual machines in heterogeneous cloud computing systems." *Future Generation Computer Systems* 74 (2017): 142-150.

[49] You, Changsheng, et al. "Energy-efficient resource allocation for mobile-edge computation offloading." *IEEE Transactions on Wireless Communications* 16.3 (2017): 1397-1411.

[50] Sharkh, Mohamed Abu, and Abdallah Shami. "An evergreen cloud: Optimizing energy efficiency in heterogeneous cloud computing architectures." *Vehicular Communications* 9 (2017): 199-210.

[51] Gai, Keke, Meikang Qiu, and Hui Zhao. "Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing." *Journal of Parallel and Distributed Computing* 111 (2018): 126-135.

[52] Vafamehr, Ali, and Mohammad E. Khodayar. "Energy-aware cloud computing." *The Electricity Journal* 31.2 (2018): 40-49.

[53] Oma, Ryuji, et al. "A tree-based model of energy-efficient fog computing systems in IoT." *Conference on Complex, Intelligent, and Software Intensive Systems*. Springer, Cham, 2018.

[54] Juarez, Fredy, Jorge Ejarque, and Rosa M. Badia. "Dynamic energy-aware scheduling for parallel task-based application in cloud computing." *Future Generation Computer Systems* 78 (2018): 257-271.

[55] Oma, Ryuji, et al. "Evaluation of an energy-efficient tree-based model of fog computing." *International Conference on Network-Based Information Systems*. Springer, Cham, 2018.

[56] Reddy, V. Dinesh, G. R. Gangadharan, and G. Subrahmanya VRK Rao. "Energy-aware virtual machine allocation and selection in cloud data centers." *Soft Computing* 23.6 (2019): 1917-1932.

[57] Sundararaj, Vinu. "Optimal task assignment in mobile cloud computing by queue-based Ant-Bee algorithm." *Wireless Personal Communications* 104.1 (2019): 173-197.

[58] Li, X. X., F. Z. He, and W. D. Li. "A cloud-terminal-based cyber-physical system architecture for energy efficient machining process optimization." *Journal of Ambient Intelligence and Humanized Computing* 10.3 (2019): 1049-1064.

[59] Sun, Gang, et al. "Energy-efficient and traffic-aware service function chaining orchestration in multi-domain networks." *Future Generation Computer Systems* 91 (2019): 347-360.

[60] Kumar, Sunil, and Mala Kalra. "A Hybrid Approach for Energy-Efficient Task Scheduling in Cloud." *Proceedings of 2nd International Conference on Communication, Computing and Networking*. Springer, Singapore, 2019..