electronics

Article

Energy Efficient Load-Balancing Mechanism in Integrated IoT–Fog–Cloud Environment

Meenu Vijarania [1], Swati Gupta [1], Akshat Agrawal [2,*], Matthew O. Adigun [3], Sunday Adeola Ajagbe [3,4] and Joseph Bamidele Awotunde [5,*]

1 Centre of Excellence, Department of Computer Science & Engineering, School of Engineering and Technology, K R Mangalam University, Gurugram 122103, India; meenuhans.83@gmail.com (M.V.); swattiguptta@gmail.com (S.G.)
2 Department of Computer Science & Engineering, Amity School of Engineering & Technology, Amity University Haryana, Gurugram 122412, India
3 Department of Computer Science, University of Zululand, Kwadlangezwa 3886, South Africa; adigunm@unizulu.ac.za (M.O.A.); 230015266@stu.unizulu.ac.za (S.A.A.)
4 Department of Computer Engineering, First Technical University, Ibadan 200255, Nigeria
5 Department of Computer Science, Faculty of Information and Communication Sciences, University of Ilorin, Ilorin 240003, Nigeria
* Correspondence: akshatag20@gmail.com (A.A.); awotunde.jb@unilorin.edu.ng (J.B.A.)

Abstract: The Internet of Things (IoT) and cloud computing have revolutionized the technological era unabatedly. These technologies have impacted our lives to a great extent. The traditional cloud model faces a variety of complications with the colossal growth of IoT and cloud applications, such as network instability, reduced bandwidth, and high latency. Fog computing is utilized to get around these problems, which brings IoT devices and cloud computing closer. Hence, to enhance system, process, and data performance, fog nodes are planted to disperse the load on cloud servers using fog computing, which helps reduce delay time and network traffic. Firstly, in this article, we highlight the various IoT–fog–cloud models for distributing the load uniformly. Secondly, an efficient solution is provided using fog computing for balancing load among fog devices. A performance evaluation of the proposed mechanism with existing techniques shows that the proposed strategy improves performance, energy consumption, throughput, and resource utilization while reducing response time.

Keywords: cloud computing; fog computing; Internet of Things; load balancing; optimization; energy consumption; wearable sensors and devices

Citation: Vijarania, M.; Gupta, S.; Agrawal, A.; Adigun, M.O.; Ajagbe, S.A.; Awotunde, J.B. Energy Efficient Load-Balancing Mechanism in Integrated IoT–Fog–Cloud Environment. *Electronics* **2023**, *12*, 2543. https://doi.org/10.3390/electronics12112543

Academic Editor: Franco Cicirelli

Received: 21 April 2023
Revised: 27 May 2023
Accepted: 1 June 2023
Published: 5 June 2023

1. Introduction

Numerous smart items and devices, including wearable technology, smartphones, and industrial utility parts, have been equipped with sensors in recent years to monitor and gather real-time physical data from their surroundings. For formulating and managing the sensor nodes, a wireless sensor network provides an efficient framework. Due to the heavy traffic generated by the Internet of Things (IoT) nodes, the network along with servers is overloaded and congested. Thereby, an intermediary layer known as the fog layer is introduced as a bridge amidst the IoT devices and cloud, which aids in achieving reduced response delays and time latency. To support the lower layers' heavy computational workloads, the cloud has introduced the intermediate fog layer to reduce the burden on heavily loaded lower layers.

Numerous complications, which include heterogeneous types of devices, limited bandwidth, restricted computing resources, a wide range of client requirements, operational costs, etc., influence the efficiency of IoT networks [1]. Thereby, an optimal solution is required to address the above-mentioned issues in the network that will significantly enhance the system's performance. Scheduling and allocation of resources are crucial for managing data centers that help maintain load balancing, proper utilization of resources, and cutting of carbon emissions [2]. Applications and sensor data are frequently sent and processed by data center computers. Due to the IoT applications' ever-growing resource requirements, it is very difficult to implement them as there is an exponential increase in energy consumption. Subsequently, the performance of the computing node degrades as a result of the massive data transmission and movement of computing devices [3]. Fog computing shifts IoT application computation from cloud platforms to the network's edge. Various applications of the cloud and IoT, such as healthcare, smart transportation, smart cities, smart homes, smart logistics, environmental monitoring, video surveillance, etc., are implemented to automate daily life processes [4]. Owing to the massive amount of data generated with fast velocity, a fog–cloud environment was needed to handle the computing resources efficiently.

In [5], fog computing is compared with other computing technologies, as represented in Figure 1. Fog computing is a distributed and emerging technology primarily focused on decreasing the latency and data transmission costs to the remote cloud.

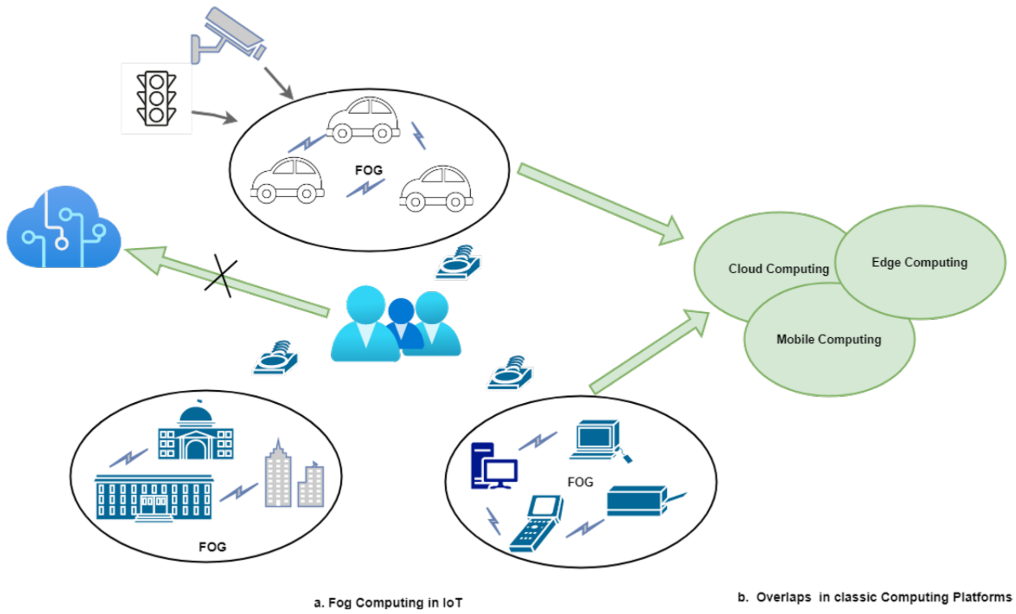

Figure 1. Integration of IoT–fog–cloud model.

Problem Statement

The inefficient management of network resources and load balancing accelerates quality-of-service degradation and energy losses. Due to increased traffic overhead on fog gateways, delays have also increased, which cannot be tolerated in real-time applications. High energy usage and load balancing are major areas of research in the fog–cloud environment. To conquer this issue, the IoT–fog–cloud model is introduced in this article by efficiently dispensing the workload among different devices at different levels. Nevertheless, only a restricted amount of information is forwarded to the cloud for further computation, which leads to enhanced bandwidth and reduced latency in the network.

This article highlights the challenges in emergent IoT, the colossal quantity of data produced through sensors, and the trouble in reporting these challenges with available networking and computing models. Thus, this research article presents the need for a new model and architecture called fog computing. Fog computing aids in filling the technological gaps of cloud and IoT models betweenness, viz., quality of service, time latency, and location awareness. The main objective of this article is to proffer a balancing load mechanism in a fog computing environment for a uniform distribution of incoming traffic among available cloud servers or fog devices, considering underloading or overloading of nodes. This improves performance, energy consumption, throughput, and resource utilization while reducing the response time. Without dependence on the cloud, IoT devices send data to a fog server for processing. Thereby, it can efficiently save cloud storage and network bandwidth for important processes and data [6–8]. A few current challenges of load balancing are efficiency, fault tolerance improvement, overhead, and the consequences of implementation.

The primary objectives are as follows:

1. Investigate the existing literature on the IoT–fog–cloud architecture and models considering energy factors.
2. The three significant issues covered in this article are load balancing, energy consumption, and computation delay or latency.
3. An energy-efficient and load-balanced mechanism for fog environments, considering available CPU capacity.
4. Performance evaluation of the proposed mechanism with existing techniques.

The organization of the rest of the article is as follows: Section 2 inspects the prior studies and analyses their weaknesses and strengths. The methodology of the proposed scheme is illustrated in Section 3. The experimental setup and simulation of this work are described in Section 4. The results and discussion are presented in Section 5. The conclusion of the proposed work is presented in Section 6.

2. Related Work

The authors in [3] proposed resource management and small cell cluster establishment of low complexity for fog computing. This article addressed the load-balancing issues by improving the user's quality of service (QoS) in fog computing. In Ref. [5], Liu, Y. et al. formulated an optimization problem for resource allocation using a genetic algorithm. The authors have also provided the framework and architecture for resource management, latency reduction, fault tolerance, etc. in fog computing. Authors in [9] developed a tree-based algorithm to uniformly disperse the workload among the fog nodes for fog computing, which progress towards the reduction in energy usage of IoT devices in contrast with the cloud model. The authors in [10] devised a scheme that converges fog computing with an energy-efficient hierarchical routing algorithm in wireless networks. Fog computing has the potential to optimize the available power efficiently. This article also used the ant colony optimization mechanism to find the optimal route for efficient data transmission, leading to energy usage reduction and delay reduction, thus extending the network lifetime. In [11], the authors devised a solution for balancing the load on the Fog of Things platform using SDN. It provides the provision of storing and processing the IoT services on edge devices and optimizing the cloud services. Authors in [12] developed an integrated model of the "Cloud of Things" (CoT), which uses fog computing for dealing with delays and energy consumption efficiently. The authors claimed that the proposed energy-aware allocation strategy saves more energy in comparison to the existing methods. In [13], the authors devised an energy-aware load balancing strategy that proficiently distributes the load among fog nodes. Using the simulation, the authors claimed that the power usage of fog and IoT nodes was curtailed significantly. In [14,15], the authors examined the trade-off between transmission latency and power usage in an IoT–fog–cloud environment. The optimal task allocation problem is formulated to minimize energy consumption with limited service delay. The authors proposed the decomposition of the primal problem into subproblems and solved them individually using optimization techniques to achieve the optimal result. In [16], the authors proposed the utilization of queuing models to investigate the transmission delays, power usage, and cost incurred during offloading tasks in the cloud–fog environment. A multiple-objective strategy is developed to minimize the payment cost, execution latency, and energy usage by finding the optimum probability for offloading and transmitting energy for all nodes. In Ref. [17], Q. Fan et al. devised a strategy for heterogeneous networks to improve green energy utilization and flow level throughput.

Similarly, in [18], the authors focused on load balancing for fog environments efficiently and devised a strategy called Dynamic Energy-Efficient Resource Allocation (DEER).

In [19–22], the authors considered the amalgamation of cloud and fog to minimize the resource cost, viz., bandwidth and link level utilization. The authors proposed an optimized strategy that takes into account bandwidth and server resources and provides efficient solutions for the three-layer fog–cloud model. In [23], the authors developed an IoT-oriented router with load metrics and also designed an algorithm for cascade failure for both local and global routing modes. Extensive experimentation and simulation proved that the local routing mode is more reliable in comparison with the global routing mode. The authors in [24] perform an extensive study of cascading failure in edge-assisted IoT. Real-time data packets, congestion status of nodes, and experimental simulation under varied compression ratios provide an optimal solution to prevent cascading failure. In [8,25], authors devised a mechanism called DEELB to deal with the IoT resource allocation problem, which considers the vast issues of energy utilization, load balancing, and the cost of computing. The experimental results indicate that the proposed mechanism provides an efficient and effective solution for allocating resources dynamically. Kaur, M. et al. suggested a model that implements load balancing at the fog layer to enhance resource utilization. When jobs are initialized in the fog nodes, the scheduling of load takes place, and in fog clusters, load balancing is handled by the local controller [26]. The authors in [27] employed simulation to examine the energy efficiency of two particular routing protocols: the Ad hoc On-demand Distance Vector (AODV) and the Destination Sequence Distance Vector (DSDV). It also contrasted how much power was needed in DSDV and AODV to send the data packets to their destinations. Results showed that AODV was more energy efficient than DSDV because its successful routing activities used less energy.

Various state-of-the-art algorithms mentioned in Table 1 deal with a variety of parameters in cloud and fog computing. As IoT devices are generating colossal amounts of data, it becomes very cumbersome to handle that data at nodes on cloud and fog levels. This article presents a novel approach to resolving this issue by integrating IoT, fog, and cloud in a three-tier structure.

Table 1. Summary of load-balancing techniques used in fog environment.

References	Methodology Adopted	Features/Parameters	Limitation
[3]	Proposed low-complexity techniques for load balancing and cluster formation for fog environment	Quality of experience, network performance, latency	The proposed algorithm is not efficient and works well with large clusters
[5]	A framework for optimized resource allocation is proposed and formulated the optimization problem using a genetic algorithm	Latency, optimized resource allocation, privacy, and fault tolerance	Results are computed using a lower number of nodes
[6]	Presented an algorithm for allocating tasks using an energy-aware policy	Energy consumption, round-trip delay	Dynamic conditions of the network were not taken into account
[9]	Developed TBFC, i.e., tree-based fog computing model	Execution time and total electric energy consumption	Proposed techniques do not consider the imbalanced tree of fog, child nodes, and edge nodes
[10]	Fog computing-based energy-efficient hierarchical routing strategies for WSN are proposed	Network lifetime, packet loss, and energy consumption	Lacks optimal resource allocation, and fog node is assumed to be balanced
[13]	In fog computing, an energy-aware and load-balanced algorithm is suggested to assign tasks to fog nodes.	Energy consumed, network cost, execution time, and load scheduling	Limited fog devices, and performance was not evaluated under different dynamic scenarios

Table 1. *Cont.*

References	Methodology Adopted	Features/Parameters	Limitation
[16]	Proposed the utilization of queuing models to investigate the transmission delays, energy usage, and cost incurred for offloading tasks in the cloud–fog scenarios	Power consumption, latency, and cost	A simulation was performed in the restricted environment
[17]	Devised a strategy for a heterogeneous network to improvise green energy utilization and flow level throughput	Throughput and power consumption	Limited to the extent of architecture
[18]	Focused on efficient load balancing for fog environment and devised the Dynamic Energy-Efficient Resource Allocation strategy	Computational cost, load balancing, and energy consumption	Lack of fault tolerance
[21]	An efficient framework is presented that considers the trade-off of power usage and delay in the integrated cloud–fog computing structure	Power consumption and delay	Optimization is not achieved in a distributed environment
[23]	A dynamic energy-efficient load balancing mechanism is proposed to deal with IoT resource allocation dynamically	Energy utilization, load balancing, and cost of computing	Service migration and security concerns
[25]	A potential mechanism for cascading failure in edge-assisted IoT is proposed	Real-time data packets and link congestion	Varied network configurations with different topologies are not considered

3. Methodology

With the tremendous growth in the field of cloud technologies, load balancing and efficient energy consumption have become very difficult tasks. Many cloud providers are available in the market, which provide various services such as pay-as-you-go with a flexible infrastructure. Two key issues arise in the IoT–fog–cloud model: firstly, the declining performance of fog gateways with continuously increasing fog devices; and secondly, the increased congestion in fog devices and the fog gateway transmission link data that are overly delayed. These issues are the repercussions of the dynamic behavior of the IoT–fog–cloud environment. Therefore, it is important to ponder the dynamic nature of the IoT–fog–cloud structure and develop an efficient algorithm to cater to the abovementioned issues. The following principles are followed while designing and implementing the algorithm for the dynamic IoT–fog–cloud environment:

A. Fog nodes are prioritized overcloud nodes. When data flows are imbalanced at fog devices or gateways, data will be forwarded to the cloud to utilize cloud nodes.
B. Cloud nodes will be utilized only when fog nodes are congested or drained of power.
C. Computational delay and latency, load balancing, and energy usage are computed in the IoT–fog–cloud network.
D. A few fog nodes or gateways can be deactivated when the computing demand is low to conserve energy.

System Model

Cisco introduced the idea of fog computing in 2014, and it provides the distributed infrastructure for IoT to expand capabilities and computing power to the network's edge [1]. In the fog model, data is managed and controlled by the end users, such as local clouds. As an example, as represented in Figure 2, there are fog nodes, N IoT nodes, and a central cloud node. Each IoT node generates lots of service requests. The traffic models are as follows: at the IoT node level, M/M/1 [28] queue; at the fog node, M/M/c [29,30] queue; at the cloud node, M/M/∞. Each IoT device forwards the request to the fog node using a wireless medium. If the maximal acceptance rate of the fog node is higher than the request rate, then the fog node will process all the requests. Otherwise, the fog device will send the traffic to the cloud centrally for execution.

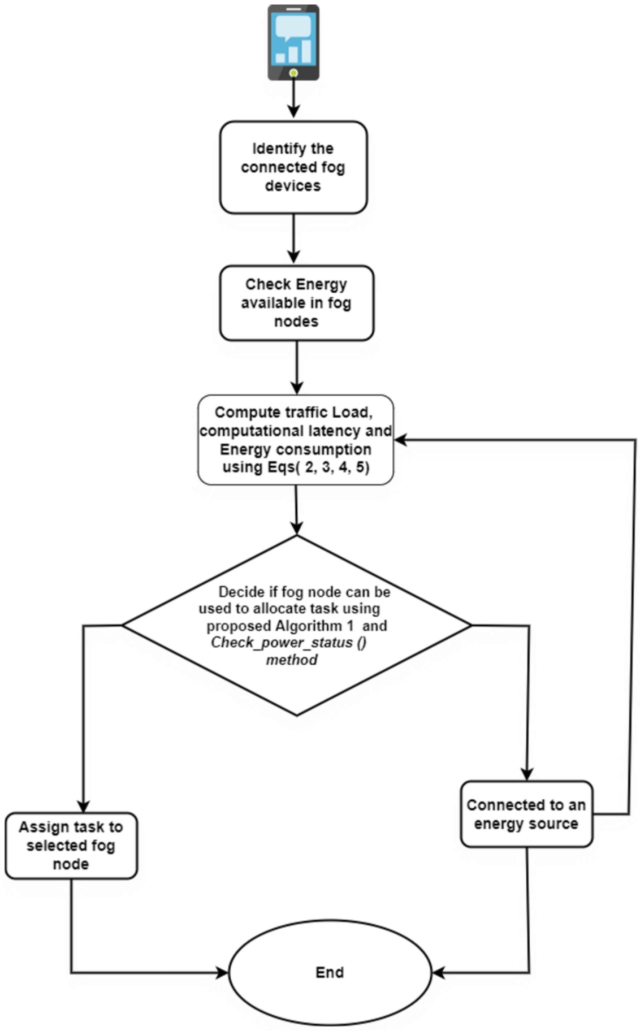

Figure 2. Flow chart of the methodology.

i. Transmission Model

Data transmission rate from IoT to fog nodes as per Shannon's equation [31] can be described as:

$$C_{i,j}(t) = Blog_2\left(1 + \frac{P_i^c(t)G_{i,j}(t)}{WN_0}\right) \tag{1}$$

where $G_{i,j}(t)$ is the channel gain, P_i^c represents the transmission power, B is the bandwidth, N0 represents the noise, and $C_{i,j}(t)$ represents the data transmission at time t from I to j. Using the M/M/1 [18], the uplink latency (transmission time and queuing) and average delay can be computed using Equation (2) [32]

$$\tau_D = \frac{1}{\frac{C_{i,j}(t)}{\omega_i(t)P_i(t)} - \omega_i(t)\varphi_i(t)} \tag{2}$$

where τ_D represents the transmission delay, ω_i denotes the offloading ratio, and P_i signifies the packet magnitude of ith task. Note that the fixed-size data are transmitted and utilized to find the transmission rate. Consequently, it is not essential to take into account each packet's length since it does not affect the optimal decision and subsequently the optimal solution.

ii. Fog device computational delay

Considering the kth fog device with the rate of traffic t_k and s_i service rate [33], the computation latency which includes service time and waiting time CD_k^{fog} is

$$CD_k^{fog} = \frac{1}{s_i - t_k} \tag{3}$$

iii. IoT–Fog System Energy Consumption Model

Overall energy usage [34] in IoT–fog can be expressed as

$$Q_{iot-fog} = \sum_{j=1}^{K} Q_j(t) + \sum_{i=1}^{N} Q_{i,j}^{c}\,(t) + \sum_{i=1}^{N} (1 - \omega_i(t)) Q_i\,(t) \tag{4}$$

where $Q_j(t)$ represents the energy used by the jth fog node that can be computed as $Q_j(t) = CD_k^{fog}\ v\ F_i(t))^3$; $Q_{i,j}^{c}\,(t)$ is the energy required to transfer data from node I to j; K represents the number of IoT nodes; and N represents the set of fog nodes.

iv. Modeling for Load Balancing in IoT–Fog–Cloud Environment

This segment presents various mechanisms for load balancing in the fog–cloud model. The prime objective of this technique is to improve the usage of present resources by uniformly distributing the tasks among computing nodes, avoiding the underloading and overloading of one node. It also takes care of idle nodes. Using fog computing, a group of fog nodes is responsible for workload distribution among different computing nodes in the cloud model, thereby achieving more reliability in incoming jobs and increasing the capability of present clients.

Balancing uniform load is important in the cloud environment owing to the dynamic number of requests coming from the client for network resources. To minimize the lag, dynamic load balancing is essential to avoid lagging in execution considering the long queue of tasks at the computing node. Architecture balancing load at various layers is shown in Figure 3 for the IoT–fog–cloud model.

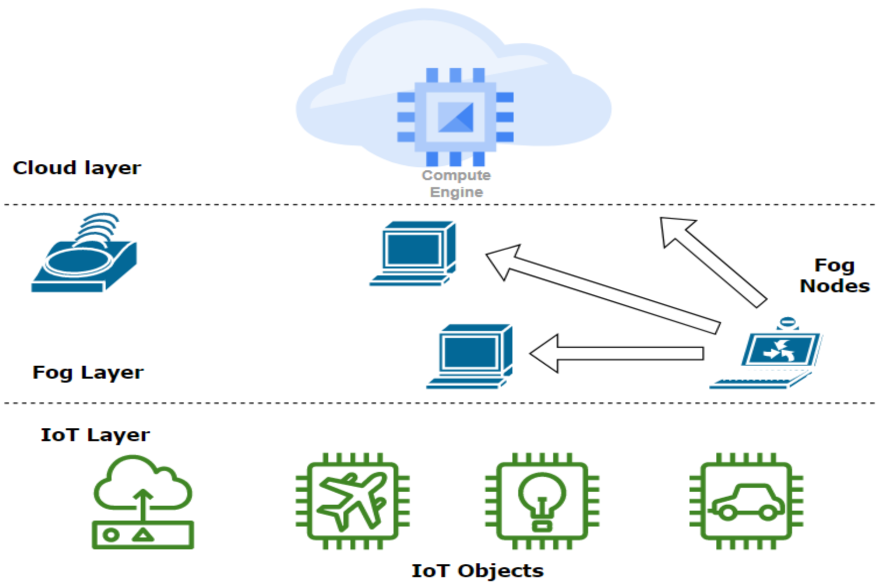

Figure 3. IoT–fog–cloud model architecture.

The current *Load* status of devices is represented in Equation

$$Load(D_j) = \frac{\sum_{n=1}^{N} P(t)}{t} \tag{5}$$

where $P(t)$ signifies the job size, and the simulation time is "t". A total number of tasks N processed at node D_j. The average load at each device is calculated as follows:

$$A_{Load^i} = \frac{\sum_{j=1}^{M} \left[\, RT_j(N) + TET\ (N) \right]}{M} \tag{6}$$

where RT_j denotes the remaining time of the node. Fog nodes M presently used at time "i". Execution time is denoted by $TET(N)$ of all jobs. At the output layer, the balanced workload is predicted for all M fog devices.

Algorithm 1 show the proposed algorithm for load balancing.

Algorithm 1: Proposed Algorithm for Load Balancing

Step 1. PowerStatus = Check_power_status (node$_s$)
Step 2. For fog node from i = 1 to n do
Step 3. Calculate the existing load of each fog node using Equation (5)
Step 4. Calculate the Power status of each fog node
Step 5. The transmission delay and Computational delay of each node are computed using Equations (2) and (3)
Step 6. Calculate the Energy consumption of each fog and IoT device
Step 7. If (PowerStatus of ith node == Full || intermediate && load_status == LOW) then
Step 8. If energy_consumption $\leq$ Threshold
Selected_Node = N$_i$
Else
Step 9. Find another Node with minimum energy consumption
Else
Step 10. Find another node
End
End

Function Check_power_status ()
Data Input: fog nodes, IoT nodes, Cloud nodes
Output: node lifetime, Power_Status;
Node_lifetime = $\frac{\text{Capacity\_of\_node}}{\text{Drain Rate}}$

1. For all nodes in the network do
   a. If (node_lifetime > 80%) then
      i Return Power_status = Full
   b. Else if (node_lifetime > 40% && node_lifetime < 80%) then
      i Return Power_status = Intermediate
   c. Else
      i Return Power_status = Low
   d. End if
2. End for

4. Experimental Setup for Simulations

The experiments were carried out with Intel and 16 GB of RAM using Ubuntu 16.04 LTS. The simulation Core i5-4210 processor time was set to 1000 s to provide variations in workload to identify the inactive FOG gateways and saturation situation. During simulation, 10 nodes are considered, which are randomly generating the data at 120 tasks per second and a data transmission rate of 2 MB/s.

The simulation was performed with varying parameters for 1200 individual runs to find a stable and feasible result. Various parameters for simulation are detailed in Table 2.

Table 2. Parameter values used for simulation.

Parameter	Values Used for Simulation
Total end devices	50 to 100
Total fog devices (F)	10 to 100
CPU frequency in cloud servers	100 × 109 [cyc/sec]
Total gateway devices (G)	2
The average number of incoming tasks (λi)	50 [tasks/sec]
Maximum channel bandwidth	30 MHz
Total cloud servers (S)	5
CPU frequency in end devices	500 × 106 [cyc/sec]
CPU frequency in fog devices	50 × 109 [cyc/sec]
Processing energy usage	0.5 Joules
The transmission power of end devices	mW

Performance Evaluation

The performance is evaluated to ascertain the efficiency of the proposed technique using the simulation toolkit fogs. Analysis and comparisons are carried out with the existing algorithm EEHFC [10] and LBS [30], considering the following qualitative performance metrics: the usage of energy and response time within the IoT–fog–cloud environment. To evaluate the performance of the proposed technique, the following metrics are calculated and utilized:

i. Network lifetime: It represents the lifespan of devices in the network. When an average number of devices are not functional, network partitioning occurs [13]. Network lifetime can be computed as follows:

$$Node\ Lifetime = \frac{Node's\ Remaining\ Energy}{Drain\ Rate} \tag{7}$$

ii. Total energy consumption: It represents the overall power consumed in the IoT–fog–cloud environment. The overall energy [35] usage of all devices can be computed using Equation (4). Due to the increased energy of fog nodes, the suggested technique saves energy and increases network longevity, as shown in Figure 3.

iii. Average Delay: It consists of both transmission and Computational delay in IoT–fog–cloud operation [36]. It can be calculated using Equations (2) and (3).

$$Average\ Delay = \frac{\tau_D + CD_k^{fog}}{Total\ No.\ of\ devices} \tag{8}$$

iv. Response time: It represents the response time of all tasks (T) up to the present interval [37]. The average response time can be computed as follows:

$$ResTime\ (T_I) = \frac{\sum_{d_t^j \in T} ResponseTime\left(d_t^j\right)}{ResponseTime(d_j^{S_e})} \tag{9}$$

where T is a set of tasks, and d_t is the host device at time t.

5. Results and Discussion

The rigorous performance evaluation of the above-suggested algorithm is carried out based on the above-mentioned parameters, as shown in Table 2, and compared with existing EEHFC and LBS techniques. The network lifetime of all nodes and all links used in the network is computed for the proposed technique and compared with other schemes, as shown in Figure 4. It is observed that the network lifetime of the proposed scheme is enhanced because it considers efficient energy consumption and load balancing among various devices at different levels in the network.

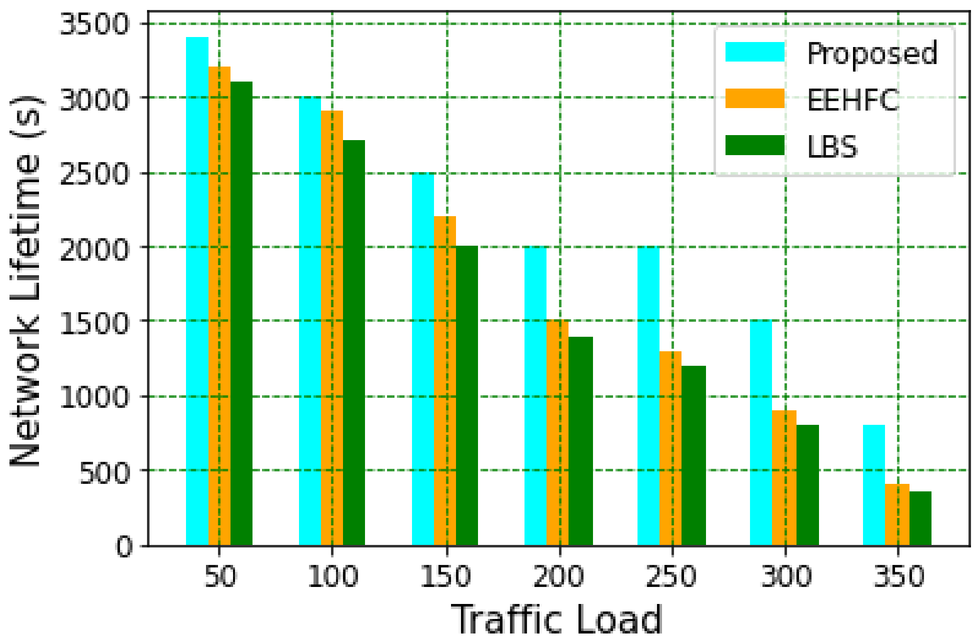

Figure 4. Network Lifetime vs. Traffic Load.

Energy consumption has been optimized for the proposed scheme, as shown in Figure 5, as compared with other existing schemes. This is because loads are balanced across different levels. If fog nodes are processing the requests of IoT devices, then the data traffic is not forwarded to the cloud. Hence, cloud nodes are efficiently utilized. The energy consumption of the proposed EEHFC and LBS techniques is represented in Figure 5. Energy usage is computed with a varying number of devices, from 50 to 500. The energy of the nodes is drained out with an increase in data traffic.

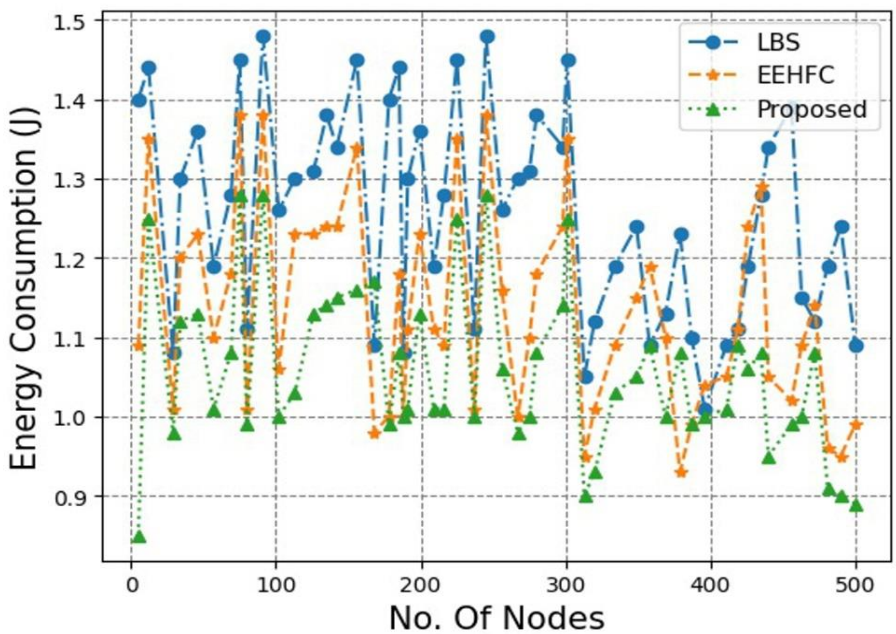

Figure 5. Energy Consumption vs. No. of nodes.

The average delay of various techniques is shown in Figure 6. Delay comprises transmission delay and computational delay at both the fog layer and the cloud layer. The proposed algorithm is focused on achieving load balancing between fog–cloud tiers. Delay is reduced as tasks are offloaded and uploaded from the fog to the cloud layer. It is visible in Figure 6 that the delay is increasing as more and more nodes and data flow through the network.

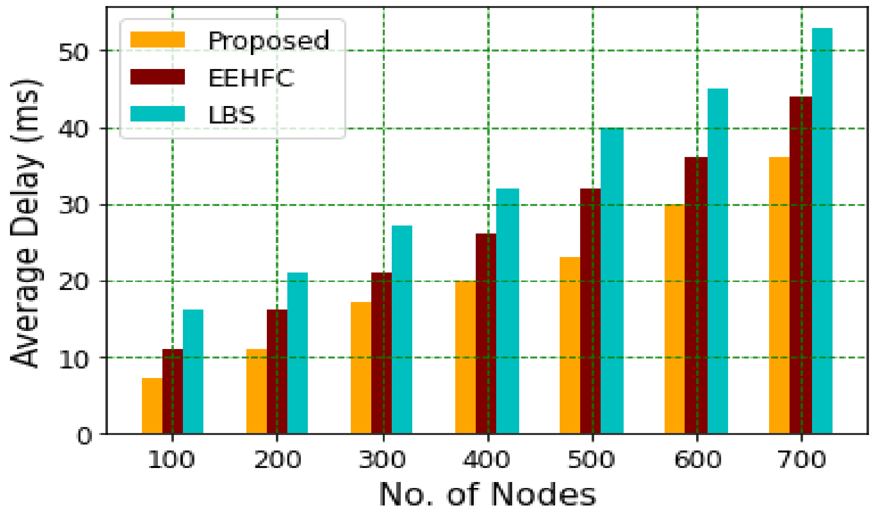

Figure 6. Average Delay vs. No. of nodes.

The information shown in Figure 7 indicates that response time is reduced with the number of nodes as a result of load balancing. This is due to the deployment of fog devices in the fog layer, which reduces the response time by distributing the load among fog devices and forwarding traffic load to the cloud layer. The response time of the fog node altered as the load on the various fog nodes increased or decreased. The response time variation in many settings demonstrates the performance of the suggested algorithm, and the findings demonstrate that the proposed technique beats competing algorithms in different scenarios.

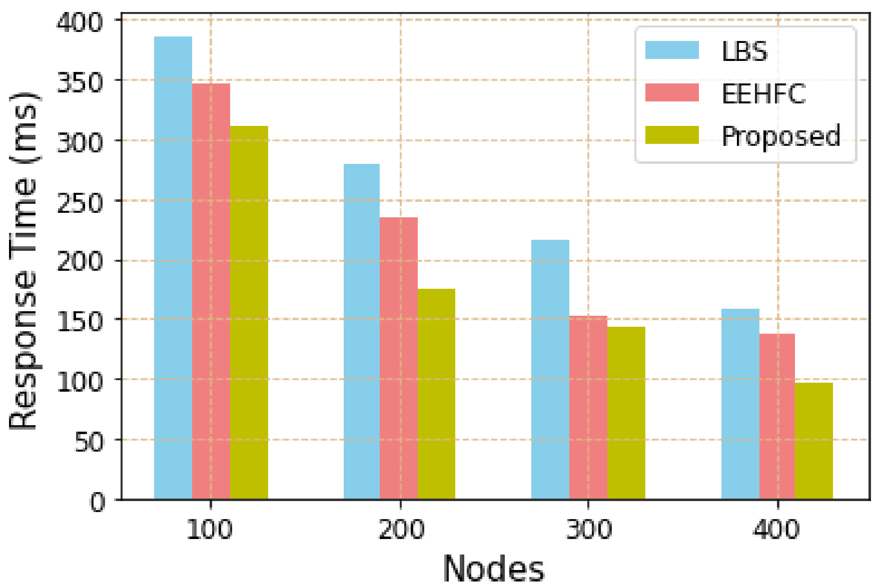

Figure 7. Response Time vs. No. of nodes.

6. Conclusions

With the colossal increase in IoT device usage, there is a need for an efficient system that can deal with the issues of increased load, energy, and delay as a result of more data traffic and processing. In this article, an integrated model is proposed for IoT–fog–cloud to efficiently use the available resources. A three-tier architecture having IoT devices, fog nodes, fog gateways, and cloud devices is used to achieve uniform load balancing among fog and cloud tiers. A novel algorithm is developed which considers energy usage of nodes, delay, network lifetime, and response time. Concretely, a systematic framework is developed to investigate the energy consumption, load balancing, and delay in an IoT–fog–cloud environment. Using the iFogSim simulator, the proposed algorithm is tested for load balancing in an IoT–fog–cloud environment. The results are presented that show the proposed model is outperforming the existing mechanisms under different scenarios. The proposed algorithm works well for a moderate number of devices., but if the IoT data continues increasing exponentially, it will be difficult to handle. Therefore, in future work, we intend to investigate the IoT–fog–cloud environment with more QoS parameters and with more data traffic at each layer.

Author Contributions: Conceptualization, project administration, and validation, M.V.; conceptualization and methodology, S.G.; software, writing—original draft, A.A.; resources, editing and review, M.O.A.; methodology, resources, writing—original draft, S.A.A.; visualization, supervision, J.B.A. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by the Centre of Excellence, University of Zululand, Kwadlangezwa 3886, South Africa.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, K.; Bagula, A.; Nyirenda, C.; Ajayi, O. An IoT-based fog computing model. *Sensors* **2019**, *19*, 2783. [CrossRef] [PubMed]
2. Sulimani, H.; Alghamdi, W.Y.; Jan, T.; Bharathy, G.; Prasad, M. Sustainability of Load Balancing Techniques in Fog Computing Environment. *Procedia Comput. Sci.* **2021**, *191*, 93–101. [CrossRef]
3. Oueis, J.; Strinati, E.C.; Barbarossa, S. The fog balancing: Load distribution for small cell cloud computing. In Proceedings of the 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), Glasgow, UK, 11–14 May 2015; pp. 1–6.
4. Chiang, M.; Zhang, T. Fog and IoT: An overview of research opportunities. *IEEE Internet Things J.* **2016**, *3*, 854–864. [CrossRef]
5. Liu, Y.; Fieldsend, J.E.; Min, G. A framework of fog computing: Architecture, challenges, and optimization. *IEEE Access* **2017**, *5*, 25445–25454. [CrossRef]
6. Mahmoud, M.M.; Rodrigues, J.J.; Saleem, K.; Al-Muhtadi, J.; Kumar, N.; Korotaev, V. Towards energy-aware fog-enabled cloud of things for healthcare. *Comput. Electr. Eng.* **2018**, *67*, 58–69. [CrossRef]
7. Díaz, M.; Martín, C.; Rubio, B. State-of-the-art, challenges, and open issues in the integration of the Internet of things and cloud computing. *J. Netw. Comput. Appl.* **2016**, *67*, 99–117. [CrossRef]
8. Zukerman, M. Introduction to Queueing Theory and Stochastic Teletraffic Models. *arXiv* **2019**, arXiv:1307.2968.
9. Oma, R.; Nakamura, S.; Duolikun, D.; Enokido, T.; Takizawa, M. An energy-efficient model for fog computing in the internet of things (IoT). *Internet Things* **2018**, *1*, 14–26. [CrossRef]
10. Abidoye, A.P.; Kabaso, B. Energy-efficient hierarchical routing in wireless sensor networks based on fog computing. *EURASIP J. Wirel. Commun. Netw.* **2021**, *2021*, 8. [CrossRef]
11. Batista, E.; Figueiredo, G.; Prazeres, C. Load balancing between fog and cloud in fog of things based platforms through software-defined networking. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 7111–7125. [CrossRef]
12. Ogungbe, A. Energy Optimization in Fog Computing to Improve Quality of Service. Ph.D. Dissertation, National College of Ireland, Dublin, Ireland, 2020.
13. Awotunde, J.B.; Folorunso, S.O.; Imoize, A.L.; Odunuga, J.O.; Lee, C.C.; Li, C.T.; Do, D.T. An Ensemble Tree-Based Model for Intrusion Detection in Industrial Internet of Things Networks. *Appl. Sci.* **2023**, *13*, 2479. [CrossRef]
14. Botta, A.; De Donato, W.; Persico, V.; Pescapé, A. Integration of cloud computing and internet of things: A survey. *Future Gener. Comput. Syst.* **2016**, *56*, 684–700. [CrossRef]
15. Liu, L.; Chang, Z.; Guo, X.; Mao, S.; Ristaniemi, T. Multiobjective optimization for computation offloading in fog computing. *IEEE Internet Things J.* **2017**, *5*, 283–294. [CrossRef]
16. Fan, Q.; Ansari, N. Throughput aware and green energy aware user association in heterogeneous networks. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6.
17. Rehman, A.U.; Ahmad, Z.; Jehangiri, A.I.; Ala'Anzy, M.A.; Othman, M.; Umar, A.I.; Ahmad, J. Dynamic Energy Efficient Resource Allocation Strategy for Load Balancing in Fog Environment. *IEEE Access* **2020**, *8*, 199829–199839. [CrossRef]
18. Suganuma, T.; Oide, T.; Kitagami, S.; Sugawara, K.; Shiratori, N. Multiagent-based flexible edge computing architecture for IoT. *IEEE Netw.* **2018**, *32*, 16–23. [CrossRef]
19. Kumar, A.; Sharma, S.; Goyal, N.; Gupta, S.K.; Kumari, S.; Kumar, S. Energy-efficient fog computing in Internet of Things based on Routing Protocol for Low-Power and Lossy Network with Contiki. *Int. J. Commun. Syst.* **2022**, *35*, e5049. [CrossRef]

20. Deng, R.; Lu, R.; Lai, C.; Luan, T.H.; Liang, H. Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption. *IEEE Internet Things J.* **2016**, *3*, 1171–1181. [CrossRef]
21. Awotunde, J.B.; Ayo, F.E.; Panigrahi, R.; Garg, A.; Bhoi, A.K.; Barsocchi, P. A Multi-level Random Forest Model-Based Intrusion Detection Using Fuzzy Inference System for Internet of Things Networks. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 31. [CrossRef]
22. Fu, X.; Pace, P.; Aloi, G.; Li, W.; Fortino, G. Cascade failures analysis of Internet of Things under global/local routing mode. *IEEE Sens. J.* **2021**, *22*, 1705–1719. [CrossRef]
23. Fu, X.; Wang, Y.; Yang, Y.; Postolache, O. Analysis on cascading reliability of edge-assisted Internet of Things. *Reliab. Eng. Syst. Saf.* **2022**, *223*, 108463. [CrossRef]
24. Shuaib, M.; Bhatia, S.; Alam, S.; Masih, R.K.; Alqahtani, N.; Basheer, S.; Alam, M.S. An Optimized, Dynamic, and Efficient Load-Balancing Framework for Resource Management in the Internet of Things (IoT) Environment. *Electronics* **2023**, *12*, 1104. [CrossRef]
25. Kaur, M.; Aron, R. Focalb: Fog computing architecture of load balancing for scientific workflow applications. *J. Grid Comput.* **2021**, *19*, 40. [CrossRef]
26. Jia, M.; Cao, J.; Liang, W. Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks. *IEEE Trans. Cloud Comput.* **2015**, *54*, 725–737. [CrossRef]
27. Ajagbe, S.A.; Ayegboyin, M.O.; Idowu, I.R.; Adeleke, T.A.; Thanh, D.N.H. Investigating Energy Efficiency of Mobile Ad-hoc Network (MANET) Routing Protocols. *Informatica* **2022**, *46*, 269–275. [CrossRef]
28. Wang, Y.; Lin, X.; Pedram, M. A nested two stage game-based optimization framework in mobile cloud computing system. In Proceedings of the 2013 IEEE Seventh International Symposium on Service-Oriented System Engineering, San Francisco, CA, USA, 25–28 March 2013; pp. 494–502.
29. Maswood, M.M.S.; Rahman, M.R.; Alharbi, A.G.; Medhi, D. A Novel Strategy to Achieve Bandwidth Cost Reduction and Load Balancing in a Cooperative Three-Layer Fog-Cloud Computing Environment. *IEEE Access* **2020**, *8*, 113737–113750. [CrossRef]
30. Asghar, A.; Abbas, A.; Khattak, H.A.; Khan, S.U. Fog Based Architecture and Load Balancing Methodology for Health Monitoring Systems. *IEEE Access* **2021**, *9*, 96189–96200. [CrossRef]
31. Singh, J.; Singh, P.; Amhoud, E.M.; Hedabou, M. Energy-Efficient and Secure Load Balancing Technique for SDN-Enabled Fog Computing. *Sustainability* **2022**, *14*, 12951. [CrossRef]
32. Sanjay, A.; Vijarania, M.; Jaglan, V. Security surveillance and home automation system using IoT. *EAI Endorsed Trans. Smart Cities* **2020**, *5*, 1–6. [CrossRef]
33. Vijarania, M.; Agrawal, A.; Sharma, M.M. Task scheduling and load balancing techniques using genetic algorithm in cloud computing. In *Soft Computing: Theories and Applications: Proceedings of SoCTA 2020*; Springer: Singapore, 2021; Volume 2, pp. 97–105.
34. Awotunde, J.B.; Arowolo, M.O.; Imoize, A.L.; Farhaoui, Y.; Adeniyi, A.E. A Machine Learning-Based Model for Energy Efficiency Classification of an Unmanned Aerial Vehicle. In *Artificial Intelligence and Smart Environment: ICAISE'2022*; Springer International Publishing: Cham, Switzerland, 2023; pp. 54–63.
35. Vijarania, M.; Dahiya, N.; Dalal, S.; Jaglan, V. WSN based efficient multi-metric routing for IoT networks. In *Green Internet of Things for Smart Cities*; CRC Press: Boca Raton, FL, USA, 2021; pp. 249–262.
36. Hassija, S.; Arora, Y.; Tripathi, K.; Vijarania, M. A Review on Congestion Control Using Routing Technique and Machine Learning. In Proceedings of the Third International Conference on Information Management and Machine Intelligence: ICIMMI 2021, Jaipur, India, 23–24 December 2021; pp. 431–443.
37. Vijarania, M.; Jaglan, V.; Mishra, B.K. The modelling of an energy efficient algorithm considering the temperature effect on the lifetime of a node in a wireless network. *Int. J. Grid High Perform. Comput.* **2020**, *12*, 87–101. [CrossRef]