

Designing of Selection Framework of Fog Nodes and Device Nodes using the Performance Parameters

Samara Mubeen

Associate Professor, Department of IS&E, JNNCE, Shimoga, Karnataka, India.

Email: samaramubeen@jnnce.ac.in

Abstract: The increase in devices leads to more processing and storage in the cloud. IoT devices failed to get the cloud on time because of overloading of the cloud. The cloud architecture is centralized, getting the processing and storing of the data for real time application results into latency. In this paper a selection module is designed which focus on three key parameters: distance, energy, and resource allocation. Distance impacts latency and network efficiency, requiring optimal placement of fog nodes to minimize communication delays. Energy consumption is a critical factor, as fog devices operate with limited power, necessitating efficient task offloading and scheduling strategies. Effective resource allocation ensures balanced workload distribution, preventing node overloading which improves overall system performance. By analyzing these parameters, an optimized framework called selection module is introduced which enhances service quality while reducing computational costs. The Selection module helps the fog nodes and device nodes match once they meet all the three parameters mentioned above.

Keywords: Distance, Energy, Fog nodes, Resource allocation, Selection module.

I. INTRODUCTION

The demand of the Internet of Things (IoT) and real-time applications has highlighted the limitations of traditional cloud computing, particularly in terms

of latency, energy consumption, and resource management. Cloud computing, while offering high computational power and storage, relies on data centers that are situated far from end devices. This distance introduces significant delays, making it unsuitable for time-sensitive applications such as healthcare, e-commerce and other automation.

Fog computing is a distributed computing model that enhances cloud capabilities by bringing processing power closer to the network edge. This reduces latency, accelerates data processing, and optimizes bandwidth usage, making it ideal for real-time applications. The node which has computation, computing, network connection, and storage can be called a fog node. Switches, servers, surveillance cameras, and routers can be fog nodes. By reducing the distance between computing nodes and end-users, fog computing minimizes latency and improves response times. Managing energy consumption in fog environments remains a challenge, as edge devices operate with limited power resources. Efficient task offloading and workload distribution are essential to balance power usage while maintaining performance.

Additionally, resource allocation plays an important role in optimizing fog computing. Unlike cloud computing, where resources are centrally managed, fog nodes require dynamic and intelligent resource allocation mechanisms to prevent overload, ensure efficient processing, and optimize energy use. Effective scheduling and strategies for matching processing and storage units are necessary to maintain system reliability and enhance overall efficiency.

This research paper explores the interplay of energy efficiency, distance minimization, and resource allocation in fog computing. By analyzing these parameters, a selection module framework is implemented to enhance the connection of fog node and device node of fog-based architectures.

The paper is organized starting with introduction in Section I, Literature Survey in Section II, Architecture of Selection Framework in Section III, Result Analysis in Section IV, Discussion in Section V followed by Conclusion in Section VI.

II. LITERATURE SURVEY

Over the past decade, cloud computing has integrated IoT applications to reduce operational costs and enhance scalability. Traditional cloud models faces with problem like the high-volume, latency-sensitive data streams generated by IoT devices. To address this challenge, fog computing emerged as a solution, bringing computation, communication, and storage resources closer to smart end-devices [1]. This paper introduces AutoFog, a fog computing framework designed to provide a holistic deployment and elasticity solution for IoT big data analytics. AutoFog includes a novel mechanism for managing elasticity in fog-based environments [1]. Additionally, the paper outlines key requirements that an IoT framework on fog computing should support. To validate the proposed framework, an experiment was conducted using a smart home use case, demonstrating its effectiveness in real-world scenarios.

Cloud computing faces some difficulties in responding to a sudden, momentous increase in high number of request for network resources with the considerable growth of mobile computing and IoT devices these days. In 2012 Cisco introduced the idea of bringing computation process to the periphery of the network so that the users can easy offload the main cloud and bandwidth, leading to a faster response time and less cost [2]. Leveraging on this concept, in this paper a novel fog-computing related framework to mitigate the problem of bandwidth bottleneck and packet delay during the Hajj season in Saudi Arabia, where in small area pilgrims are gathered for short period of time. The

proposed framework yields efficient data entry and information retrieval.

The Internet of Things (IoT) era is characterized by billions of connected devices continuously generating vast amounts of data every second. However, the traditional centralized data center model faces challenges due to the long distances between data sources and processing centers, leading to inefficiencies in data handling [3]. To address these limitations, fog computing has emerged as a solution, leveraging advanced network topologies to enable data processing and storage at the network edge. Fog computing architecture and enhancing Quality of Service (QoS) is a crucial area of research. This paper proposes a fog computing architecture aimed at improving QoS for IoT applications [3]. The fog nodes cooperation is proposed system facilitates a specific location, enabling shared data processing. This collaborative approach fosters a distributed computing ecosystem capable of efficiently handling large volumes of service requests.

An alternative IT infrastructure and service model is the cloud computing. Cloud does not handle local issues involving a large number of networked elements and it is responsive enough for application that requires immediate attention of a local controller [4]. To address this issue Fog computing made it way to preserve many benefits of cloud computing and it is also resolves the local and performance issues because its resources and specific services are virtualized and located at the edge of the customer premise [4]. Data security is the challenging as the data movement is between the fog nodes and IoT devices. There is a region based trust aware model for trust among fog node present in the regions, a fog based privacy aware role based access control for access control at fog nodes and developing a mobility management service to handle changes of users and fog devices locations.

Fog computing, a promising architecture, supports on-demand services by leveraging increasingly powerful fog nodes that collaborate for communication and computation. This paper addresses the time reuse problem in fog-enabled cooperative networks [5]. A limited set of time reuse

patterns is determined using the alternate convex search method and the alternating direction method of multipliers. Additionally, a low-complexity semi-distributed algorithm, known as the alternate distributed allocation of time reuse patterns, is proposed to identify the optimal time reuse profile and corresponding association strategies [5]. The semi-distributed nature of this solution makes it well-suited for fog-enabled cooperative D2D networks. Numerical simulations are conducted to validate the proposed design.

Modern IT user's most popular technology is internet of things which is a newer technology. IoT devices are connected with the internet and resources are stored in the cloud. IoT devices are energy constrained it can't directly connect with internet or cloud [6]. Fog computing is used to connect IoT device to internet and cloud; these are extension of cloud computing is fog computing, where the computation is done in edge devices and these are localized data centers or cloudlets. In this paper resource management architectural in fog computing is proposed which contains six components which include resource co-coordinator, scheduler, request handler, resource estimator, allocator and work load analyzer [6]. In future this model can be implemented using real world IoT simulation environments for fog computing.

Fog computing are devoted research efforts for acknowledging the shortcomings of cloud computing. There is rapid increase in the number of devices at the extreme edge of the network that imply the need for timely and local processing [7]. Fog computing offers a promising solution to move computational capabilities closer to the data generated by the devices. In this paper research efforts, application of fog computing how is it beneficial and identify future challenges to be faced by fog computing.

An integral part of our daily lives Internet of Things (IoT) has become. Cloud computing with IoT has proven to be excellent technology for deploying in various fields. Cloud computing is used for processing and storing of the data generated by the IoT devices [8]. There some issues faced by this approach like latency, energy, computation resources, availability,

bandwidth etc. Middle tier computing known as fog computing is used which helps to overcome the obstacles faced by cloud and IoT device [8]. Fog collects and processes the information locally before sending it to the cloud for storing. In this paper a structured review of the current state of fog computing in IoT and its architecture deployed across multiple industries. Multiple challenges faced by fog computing faces in a structured six level approach. This leads to future research in resource management, green computing and security.

The cloud computing paradigm which is centralized faces challenges such as high latency, limited capacity, and network failures, especially with the rapid expansion of Internet of Things (IoT) applications [9]. Fog computing addresses these issues by bringing cloud capabilities closer to IoT devices, enabling local data processing and storage instead of relying solely on cloud servers. By providing faster and higher-quality services, fog computing is considered a superior choice for ensuring efficient and secure IoT operations [9]. This paper explores the benefits and state-of-the-art implementation of fog computing, emphasizing its integration with IoT. It also examines fog computing architecture and its role in enhancing emerging IoT applications. Furthermore, open challenges and future research directions related to fog computing and IoT are discussed.

Fog computing has emerged as a cutting-edge technology to address computing and networking bottlenecks in large-scale IoT deployments. As a complementary paradigm to cloud computing, it brings computational, networking, storage, and acceleration resources closer to the edge and network layers [10]. This paper includes a comprehensive taxonomy covering the architectural, algorithmic, and technological aspects of fog computing. It also reviews the architectural distinctions between various computing paradigms, including cloud, edge, mobile edge, and fog computing [10]. Additionally, a proposed maturity model is utilized to assess base architectures in terms of application, software, and security, computing resource management, and networking.

III. ARCHITECTURE OF SELECTION FRAMEWORK

In this the architectural framework of selection module is designed. The Fig. 1 shows the architecture of selection of devices by the fog nodes. Selection module is placed between the fog nodes and the

devices. Selection module helps the device to select respective fog nodes based on the different parameters, like distance between the device and fog nodes, energy of the fog nodes as it's required for processing the information and resource allocated to the fog nodes.

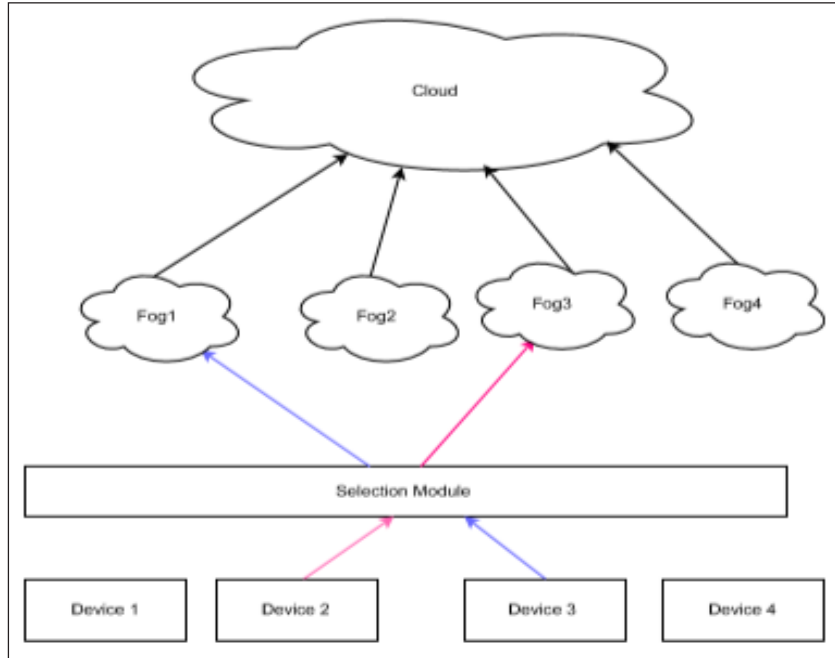


Fig. 1: Architecture of Selection Framework

Fog nodes handle multiple devices simultaneously, processing their information based on resource availability. Since the resources of a fog node may not always be sufficient to satisfy the demand, a resource allocation parameter assess the availability of resources within each fog node.

The diagram illustrates how the selection module assigns devices to fog nodes. Specifically, Device 3 selects Fog Node 1 for processing, while Device 2 selects Fog Node 3. This allocation ensures efficient utilization of available fog resources.

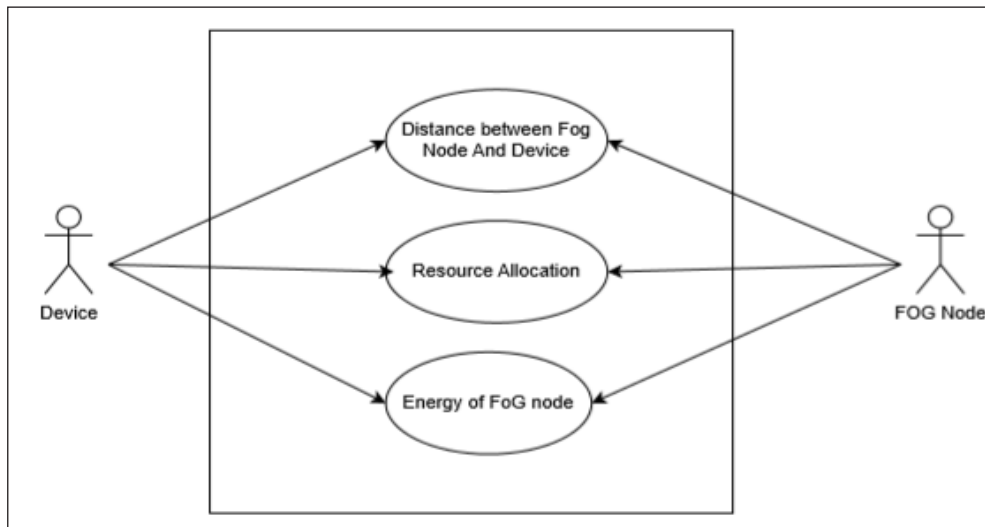


Fig. 2: Use Case of Selection Module

Fig. 2 presents the use case model of the selection module. The use case diagram features two primary actors: the device and the fog node. The selection module comprises three key use cases. First one is distance calculation to determine the distance between the fog node and the device, second is the resource allocation to identify the availability of resources in the fog node and finally Energy Assessment which checks the energy levels of the fog node.

The device node interacts with these use cases to select the most suitable fog node, as matched by the selection module.

Implementation of Selection Module

Fig. 3 illustrates the flowchart for implementing the selection module. The process begins by identifying the available fog nodes and determining the positions between each fog node and the respective device. Additionally, the energy levels and their allocated resources are recorded.

Threshold values are set for key parameters, including distance (th), energy (th1), and resources (th2). A comparison is then made between these thresholds and the actual values: the distance between the fog node and the device, the energy available in the fog node, and the allocated resources. If all threshold conditions are met, the respective device is assigned to the appropriate fog node for processing.

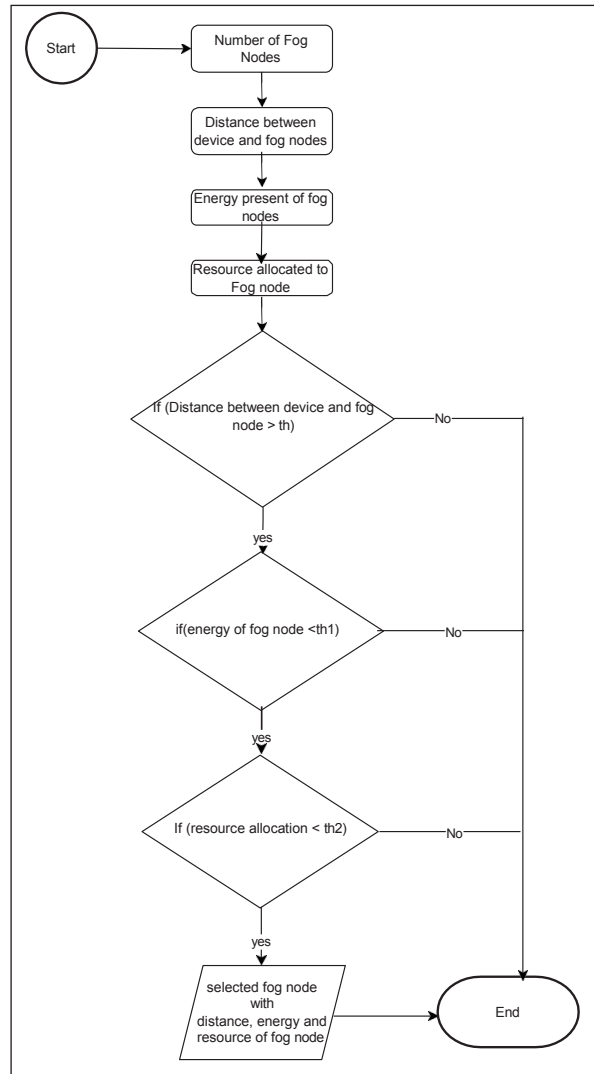


Fig. 3: Flowchart of Implementation of Selection Module

IV. RESULT ANALYSIS

This section presents the analysis of results based on the output generated from running the Python code. Threshold values are set for key performance parameters, including the energy levels of fog nodes,

the distance between the fog node and the device, and the resources allocated to each fog node. Comparing these parameters against their respective thresholds to assess the effectiveness of the selection module in optimizing resource allocation and device-to-fog node assignments.

TABLE I: SELECTION OF FOG NODE AND DEVICE BASED ON PERFORMANCE PARAMETERS

Iteration	Energy of Fog Node > 60	Distance between the Fog Node and Device < 100	Resource Allocated to Fog Node < 10	Fog Node Selected	Device Node Selected
1	63	29	7	4	3
2	NA	NA	NA	NA	NA
3	87	45	9	2	1
4	61	33	6	1	2
5	86	92	9	3	4
6	70	53	7	4	1

Table I presents various parameters used to determine the selection of fog nodes and device nodes based on threshold values. The thresholds are set as follows:

- Energy of the fog node is set to 60.
- Distance between the fog node and the device is measured as 100 meters.
- Resource allocation for the fog node is allocated as 10.

Each iteration is evaluated against these conditions. If all thresholds are met, the corresponding fog node and device node are selected and displayed.

In Iteration 1, all performance parameters energy, distance, and resource allocation—satisfy their respective threshold conditions, resulting in the selection of the appropriate fog node and device node. However, in Iteration 2, the performance parameters fail to meet the threshold requirements, preventing the selection of any fog or device nodes.

V. DISCUSSION

The proposed selection module efficiently identifies the most suitable fog node and device by considering three critical parameters: distance, energy, and resource availability. The results summarize that the module significantly selects the free available fog

node and devices; only thing is they should satisfy the entire three thresholds.

VI. CONCLUSION

Fog computing is extending version of cloud computing in which the nodes are localized near the devices by where data is processed and storage is very fast compared to cloud computing. IoT devices fail to get the fog nodes, as the due longer distance, scarcity of resource and less energy present in the fog nodes. To overcome this selection module is designed and implemented which helps to establish connection between the fog node and devices based on the threshold. Threshold is set for all the three parameters. If all the parameters thresholds are met, then the selection module helps to select respective fog node and device by which the data processing and storage of information can be made speed. The selection module which is designed and implemented using only three parameters, beside this other related features would have been considered. In the selection module the fog node and devices are selected only if all the condition are satisfied, out of 100 iteration we may get selection for 70%, the selection can upgraded in future work.

REFERENCES

- [1] L. M. Pham, T.-T. Nguyen, and T.-Q. Hoang, "Towards an elastic fog-computing framework for IoT big data analytics applications," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1-16, 2021, Art. no. 3833644, doi: <https://doi.org/10.1155/2021/3833644>.
- [2] S. Alraddady, A. S. Li, B. Soh, and Mohd. Alzain, "Deployment of fog computing during Hajj season: A proposed framework," in *5th Information Systems International Conference*, in *Procedia Computer Science*, vol. 161, pp. 1072-1079, 2019.
- [3] Mohd. Al-Khafaji, T. Baker, H. A. Al-Libawy, A. Waraich, C. Chalmers, and O. Alfandi, "Fog computing framework for internet of things applications," in *11th International Conference on Developments in eSystems Engineering (DeSE)*, 2018, pp. 71-77.
- [4] T. D. Dang, and D. Hoang, "A data protection model for fog computing," in *2nd International Conference on Fog and Mobile Edge Computing (FMEEC)*, 2017, pp. 8-11.
- [5] S. Jin, Z. Zhu, Y. Yang, M.-T. Zhou, and X. Luo, "Alternate distributed allocation of time reuse patterns in fog-enabled cooperative D2D networks," *IEEE Fog World Congress (FWC)*, 2017, pp. 1-6.
- [6] B. J. Hubert Shanthan, and L. Arockiam, "Architectural framework for resource management in fog computing," *International Journal of Engineering Research & Technology (Special Issue)*, vol. 5, no. 13, pp. 1-4, 2017.
- [7] J. Gedeon, J. Heuschkel, L. Wang, and M. Mühlhäuser, "Fog computing: Current research and future challenges," *Fog Computing*. Darmstadt, Germany, 2018.
- [8] A. A. Bakar Sajak, R. Qureshi, M. F. Azril Zuhairi, and Z. M. P. Ahmad Baidowid, "A review of fog computing: Concept, architecture, application, parameters, and challenges," *International Journal on Informatics Visualization*, vol. 8, no. 2, pp. 564-575, May 2024.
- [9] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," *Big Data Cognitive Computing*, vol. 2, no. 10, 2018, doi: <https://doi.org/10.3390/bdcc2020010>.
- [10] P. Habibi, Mohd. Farhoudi, S. Kazemian, S. Khorsandi, and A. Leon-Garcia, "Fog computing: A comprehensive architectural survey," *IEEE Access*, vol. 8, pp. 69105-69133, 2020.