DOI: 10.24996/ijs.2020.61.7.27

ISSN: 0067-2904

Fog-based Spider Web Algorithm to Overcome Latency in Cloud Computing

Abd Rashid Dar*[1], D Ravindran[1], Shahidul Islam[3]

[1]Department of Computer Science, St Joseph's College (Autonomous),(Affiliated to Bharathidasan University) Tiruchirappalli-620002, Tamil Nadu, India

[2]Department of Computer Science, Baba Ghulam Shah Badshah University Rajouri-185234, Jammu & Kashmir

Received: 25/11/2019 Accepted: 15/3/2020

Abstract

The cloud-users are getting impatient by experiencing the delays in loading the content of the web applications over the internet, which is usually caused by the complex latency while accessing the cloud datacenters distant from the cloud-users. It is becoming a catastrophic situation in availing the services and applications over the cloud-centric network. In cloud, workload is distributed across the multiple layers which also increases the latency. Time-sensitive Internet of Things (IoT) applications and services, usually in a cloud platform, are running over various virtual machines (VM's) and possess high complexities while interacting. They face difficulties in the consolidations of the various applications containing heterogenetic workloads. Fog computing takes the cloud computing services to the edge-network, where computation, communication and storage are within the proximity to the end-user's edge devices. Thus, it utilizes the maximum network bandwidth, enriches the mobility, and lowers the latency. It is a futuristic, convenient and more reliable platform to overcome the cloud computing issues. In this manuscript, we propose a Fog-based Spider Web Algorithm (FSWA), a heuristic approach which reduces the delays time (DT) and enhances the response time (RT) during the workflow among the various edge nodes across the fog network. The main purpose is to trace and locate the nearest f-node for computation and to reduce the latency across the various nodes in a network. Reduction of latency will enhance the quality of service (QoS) parameters, smooth resource distribution, and services availability. Latency can be an important factor for resource optimization issues in distributed computing environments. In comparison to the cloud computing, the latency in fog computing is much improved.

Keywords: Cloud Computing, Delay-time, Latency, VMware, Workloads.

1. Introduction

Cloud computing is a network-based computing paradigm that allows the cloud-users to access the computing resources anytime, anywhere, and on the go. Cloud computing is a distributed platform where the cloud macro data centers and servers are being accessed geographically. Various models (Public, Private, Hybrid & Community) are deployed and different services (Software, Platform, Infrastructure & Network) are offered geographically across the globe [1]. Resource distribution occasionally becomes a quite frustration of a devastating problem over the cloud computing platform as the resources over the distributed cloud servers are highly complex and congested. Hence, the workload distribution is not quite often smooth because the congestion over the core-cloud network increases the complexity in cloud latency from the various participating computing nodes over the network [2]. Cisco

*Email: phd_cs_aamirrashid894@mail.sjctni.edu

introduced the new computing paradigm known as fog computing that is extending the cloud computing capabilities by transforming the computational power from the core network to the edge network into a decentralized distributed computing platform. Fog computing is an ideal platform for services and applications running over the billions of connected edge-devices to the Internet of Things (IoT) environment. It offers the same computational tasks (computing, storage, and networking) as those of the cloud but it provides greater intent and proximity as computing takes place nearer to the user's edge devices [3]. Fog is placed below and near to the cloud, as an intermediate layer between the cloud core data centres and edge devices [4]. Fig.1 shows the hierarchical architectural overview of fog computing. It resides near to the grounded edge devices that makes it the most efficient and elegant platform for the applications and services running in real-time environments. Fog-nodes allow the processing of data at the edge of the network rather than sending it to the cloud macro datacentres [5]. The proposed FSWA approach offers better response time, reduces the complexity of latency, and enhances the searching in proximity as compared to the traditional cloud computing, which is a necessity for the applications and services running in an IoT domain. The workloads are temporally offloaded to the fog nano data centres to reduce the network congestions and bottleneck situations arising in a network during data communication and transformation.

In this manuscript, we highlight some of the issues such as complex latencies and response and searching time that cloud computing end-users are facing. Latency is the most frustrating problem arising while accessing the resources and services in a centralized cloud domain. The main contributions are as follows:

- We put forward the fog-based spider-web algorithmic and consider it to be a suitable approach for real-time services and applications on a distributed platform.
- The main purpose of using fog computing as a middleware utility is to reduce the time delays (TD), the response time (RT), jitter, and congestion of the network to enhance the cloud-user computing experience globally over the cloud network.
- We put forward a descriptive literature review, where we define and highlight the various factors causing the high latencies in the cloud environment.
- In the proposed FSWA, the nearest available resource searching will be minimum as in fog-based spider web search approaches. The fog-nodes can be located in proximity to the edge-users.
- The reductions in hops count can improve the lagging process.

The remaining parts of this manuscript are organized as follows. In Section II, we conduct a brief review of related articles to our problem of interest. In section III, we define some of the recent trends in the cloud domain. In Section IV, we come with the statement of the cloud-latency problem and the various parameters in the cloud domain causing complex latencies. In Section V, we propose a methodology to overcome the latency-oriented issues with our proposed FSWA. Finally, in Section VI, we conclude the manuscript and provide future directions and scope of the related work.

1. RELATED WORK

In an earlier work [6], the authors proposed a delay minimizing analytical model reducing the service delays in the IoT environment. The reductions in the delays automatically improved the QoS for the time-sensitive IoT applications. In another study [10], the authors proposed the novel swarm intelligence-based metaheuristic Social Spider Cloud Web Algorithm, reducing the workloads distribution issues in the centralized cloud computing platform. The foraging behavioural mechanism was used to achieve the maximum number of positions of the prey within the nearest local geographical locations. In another work [11], the authors proposed and formulated the convex optimized distributed algorithm based on the proximal algorithm to improve the joint resource allocation of minimizing the carbon footprints for video streaming over fog computing. Their methodology was designed for solving the large-scale problems into smaller subproblems. In another report [12], the authors examined the properties of completing the latencies in fog systems on six heterogenic points conducting on benchmarking-based task completion latencies in fog computing. In the conducted work, their results reflected the optimal latencies. However, the validations of their suggested algorithms were stated as their futuristic work. In another study [13], the authors proposed a novel profiling-based consolidation methodology and the main aim was to improve the tail latency by reducing the number of physical machines. Experimental results show that the proposed method can greatly reduce the tail latency compared with the traditional consolidation method. In another article [14], the authors presented the smart gateways for the fog computing platform. Various delay parameters such as synchronization, uploading, and network jitters were addressed. Smart gateway-based communication was shown to reduce the workload burden and enhance cloud resource provisioning, resulting in efficient utilization. In another study [15], the authors presented the cost-based task scheduling algorithm for better recovery time in failure circumstances and improved the reliability and availability of cloud services. In a later work [16], the authors put forward the framework for distributed applications and proposed a delay-minimizing approach with an analytical model to reduce the IoT-enabled, environment-based service delays and offloading policies for devices running over the fog-platform. In another work [17], the authors considered the edge-fog computing to overcome the MCPS's instability and the long delays across the health devices and cloud data centres. They also considered the base station association, task distribution, and virtual machine placement in fog platform-supported MCPS's by lowering the resource utilization costs and satisfying the QoS. In another report [18], the authors proposed a fog server deployment technique to reduce the movement of data path in a fog computing. They applied a technique to utilize the resources of devices by fully using a vector bin packing algorithm. The results showed that their proposed algorithm reduced the distance of the data movement and minimized the utilization of the fog computing resources.

2. Recent Trends in Centralized Computing

3.1 Fog Computing

Fog computing is an extended version of cloud computing that was introduced by Cisco in 2012. It transfers the power of computation from the core-network to edge-network. Fog is considered to be the best suitable platform for the Internet of Things. Some of fog computing characteristics are important to be highlighted in this paper to understand how it is going to extend the cloud capabilities. Fog might overcome various inherent drawbacks of cloud computing since it acts as an intimidate layer between the End-user/IoT-devices and Cloud data centres. Provisioning of various computing resources at the edge-network will reduce the hops-count and decrease the complexity in cloud latency. Preferably, the features of fog described below might reduce some of the cloud limitations while accessing the various time-sensitive services and applications. It is a convenient platform for autonomy and service experience efficiency [7]. Some of the features are as below;

- **Latency:** It supports low latency-oriented applications like gaming and video streaming consoles.
- **Real-time Analytics:** Real-time data analysis at the fog layer improves the latencies, which is beneficial for applications running in an IoT environment.
- **Federation:** Cloud federation offers the back end support and reduces the bottlenecks and network congestions, thus providing smooth communication across the heterogenic network.
- **Local-awareness:** the various nodes are near to the cloud, which reduces the response round time delays.
- **Hops count:** It reduces the distance from the source to destination by minimizing the hop count among the fog devices wherein the actual data travels.
- **Proliferation:** Now and then, the new breed of services and applications are developing and fog is more convenient, feasible and more reliable platform IoT-enabled edge computing.
- **Smart Utility:** Fog is predominant in connected vehicles, smart grids, wireless actuators, sensors and smart cities.
- **Collaboration**: By the integration between the two interdependent platforms, their positive consequences will lead to the better Principle of Fog's localization and Cloud's globalization.

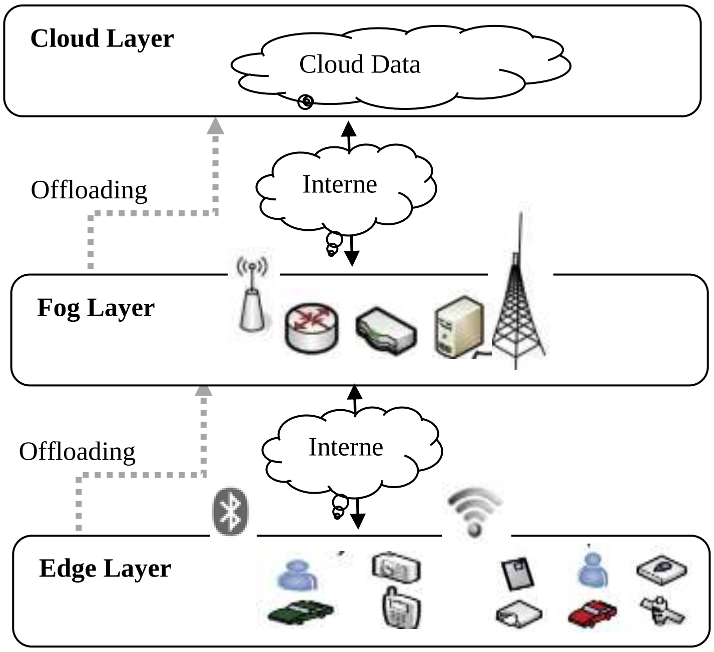

Figure 1-Layered Architectural Overview of Fog Computing

2.2 Edge Computing

Edge devices produce the voluminous data at the edge-network. Edge computing is the closest to the source of the data generation and inherits the capabilities and functionalities to process the data directly at the user's end rather than sending it to the macro cloud datacenter for processing. It reduces the latency and enhances the performance of the edge-devices over the edge-network [8]. All the IoT-enabled devices, connected devices, remote sensors of various machines, and other components generate the massive amount of data dynamically. Hence, to process the bulky data itself by the edge devices is the cumbersome and the center of the problem. The data is being offloaded to the fog-nodes for further processing. In Edge Computing, data-streaming and processing in real-time are being improved and latency is reduced to some extent. It allows smart handheld devices and applications to respond to the generated data instantaneously without further delays. Edge computing is highly worthy in driverless cars, real-time data analytics, traffic and weather forecast, etc. Edge will enhance and improve features as:

- **Agility:** Data processing is taking place thus enhancing the speed and agility.
- **Scalability:** It provides a scalability feature and expands and enriches the computing by integrating various IoT nodes over the edge network and expanding the company's network.
- **Security:** Cloud computing is a centralized platform where the processing of data is taking place centrally, making it most of the time unavailable. This leads to denial of service (DoS) and data is at threat of cyber-attacks while in edge. Data processing takes place within the edge-device while transmission takes place locally. Data remain within the reach of the edge network. Thus, the chances of security vulnerabilities are minimized.
- **Reliability:** Edge-devices and its nano data centres are connected to the network in a decentralized fashion, thus making it difficult for any possible failure. Even if failure happens, least are the chances of the entire shutdown of the services.
- **Versatility:** Online Streaming Content (OSC) such as Netflix, Amazon Prime, Zee5, and MxPlayer need a versatile platform. Edge is the convenient and feasible platform for delivering smooth content because it offers reduced network latency.

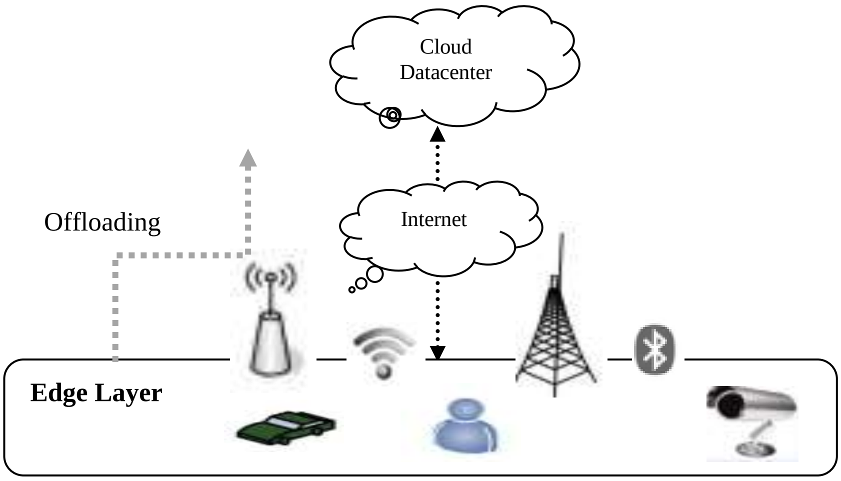

Figure 2-Architectural Overview of Edge Computing

3.3 Dew Computing

Among all definitions that have been put forward so far, we found the following as the best definition of dew computing; "It is an on-premises computer software-hardware organization paradigm in the cloud computing environment where the on-premises computer provides functionality that is independent of cloud services and is also collaborative with cloud services. The goal of dew computing is to fully realize the potentials of on-premises computers and cloud services" [9]. It provides offline cloud services to the end-users and integrates centric-cloud computing capabilities of smart edge-devices.

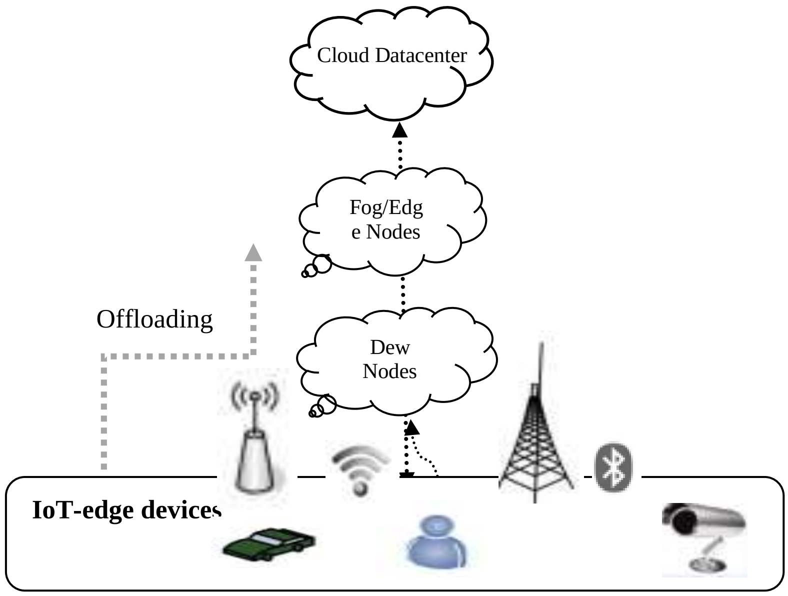

Figure 3- Architectural Overview of Dew Computing

3.4 Mist Computing

Mist can be a small fog that manages the droplets in the cloudlets in a cloud domain. As cloudlets are farthest to fog nodes, the edge-devices are far from the centre cloud, while mist fills up the gaps among the fog, edge and cloud nodes and acts as a bridge among them. Mist is nearest to the fog and is smaller than it but larger than the cloudlets [4].

In a broad vision, although the edge, mist, and dew computing are defined separately, the combined propose of these decentralized paradigms is to minimize the hops count between the end-users/IoT-devices and the centralized cloud. This makes the IoT-environment accessible without service delays in fog network by reducing the latencies over the cloud computing paradigm. Fog deploys various resources across the cloud network and the services and applications located at the edge-network will improve the overall performance of the system compared to running the application over the cloud computing.

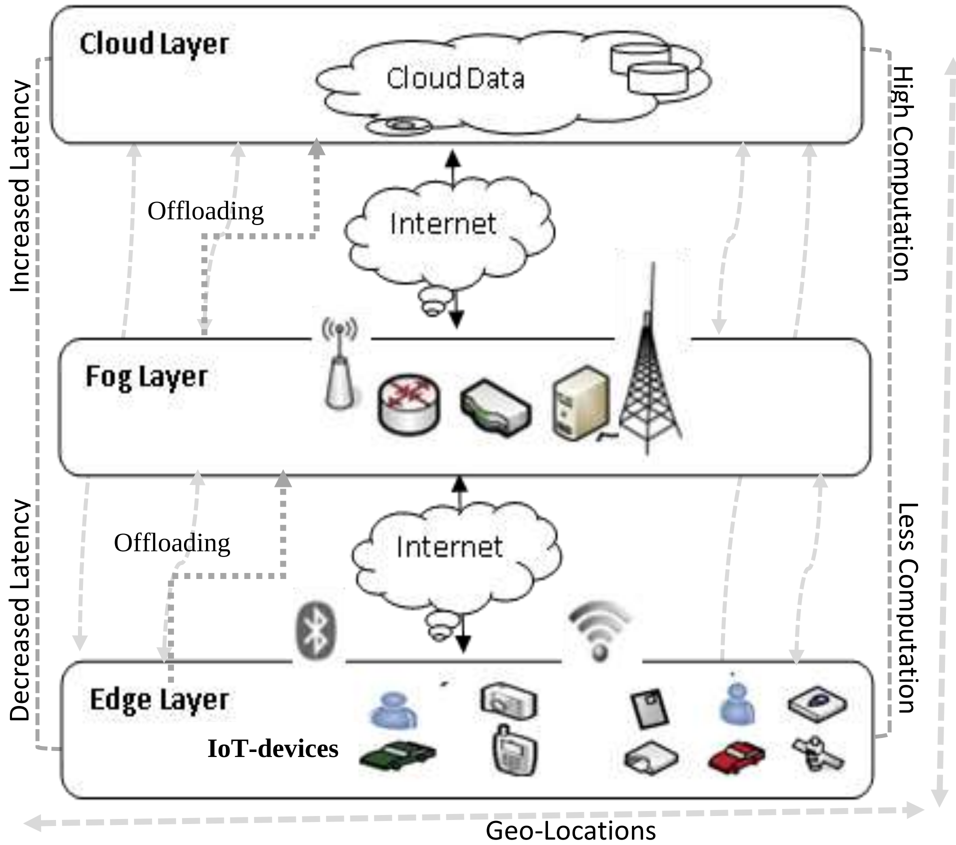

Figure 4-Communication among various nodes in Fog Computing

3. Problem Statement: Cloud Latency

In IoT environment, the various nodes are connected and exchange the data with each other via internet. This raises many challenges like RT, DT, latency, real-time decision making, and location awareness that have a direct impact on the functioning of an IoT-enabled application. Among all the issues and challenges that cloud computing is facing right now, the biggest one is the latency. The complex and high latency of centralized cloud computing for real time-sensitive applications running over the cloud network is a challenging issue. The high-quality multimedia activities such as online video streaming, gaming consoles, HD Netflix and Amazon Prime content, alongside the telephonic conversation, utilize the maximum internet bandwidths that need a sophisticated computing platform. The cloud causes time and jitter delays, network congestion, and service downtime of the real-time sensitive applications, affecting the web application to load smoothly. This is a hectic process when it comes to the real-time scenarios. Users are experiencing the new heights in distributed technologies, and the network delays may have the catastrophic effect on the customers whose business is solely dependent on the network speed. Better web experience is the first priority of the end-users for smooth functioning. On-demand cloud services need agility in network speed. Latency is the biggest challenge for everyone and needs to be addressed as soon as possible. In the broad vision, the latency of network-connected devices can be described as the time required for data transmission from the source to the sink over the network link. As in distributed computing platforms, these interconnected computing nodes possess the inherent latency and they vary across all the three cloud deployment service models.

4.1 Factors Causing High Latency in Cloud-environment

Cloud computing possesses high computation power and storage capability. However, in recent times, the emergence of IoT computing platform generated excessive data and processing it over a centralized computing model that lacks the sufficient network bandwidth, which creates a bottleneck situation due to the bulk-sized data and leads in complex latency to the time-sensitive systems. For decentralized platform like Internet of Things, heterogenic services and applications are sent over the cloud platform and the problem of latency is becoming highly complex. In order to run the time-sensitive applications and scale up the computational resources, it is important for IoT to leverage the cloud resources and overcome the high latency" [19]. Some of the factors that are the main cause of high latency in cloud computing are as below:

- **Distributed Computing:** The enterprise datacentres contain all the data and infrastructure of an organization and is being distributed across the globe. Latency issues are being high and complex in nature. The variance in data transmission occurs across different nodes over the internet, leading to high complexity in cloud latency.
- **Infrastructure Virtualization:** The delays of data transmission in the virtualized platform are high, which leads to the worst impact on cloud latency by increasing its complexity.
- **The dearth of Measuring Tools:** The ping and trace-route are used as diagnostic tools to measure delay transits and show the routing path. However, dynamic modern web applications do not use the Internet Control Message Protocol (ICMP), but instead of it, these network applications use Hypertext Transfer Protocol (HTTP) and File Transfer Protocol (FTP).
- **Quality of Service and Network Traffic Priorities:** The covenants, such as SLAs and QoS among the various cloud participants, aim to enhance the computing experience over the internet. In the cloud environment, various applications and services have different network latency tolerance.
- **Unavailability of Cloud Providers:** In a distributed cloud environment, a client is not aware of where the cloud data centres are located. Searching nearer or farthest data centres leads to searching complexity and inversely increase the cloud latency. The search efficiency is high in data centres that are located near the source of origination.
- **Network Traffic Workload:** Surfing experience over the internet is not always smooth, rather it varies periodically. As in peak or prime hours, network lagging is always quite common, while in the rest of the hours, network lagging is less. The high latency is also occurring because of the geographical elevation. The congestion in the network is another reason for high latency, as several users are online and simultaneously accessing the online real-time content (e.g. gaming and video streaming) causing the lag in the network.

5. FSWA- Fog-based Spider Web Algorithm

The FSWA is working in heuristic-foraging manner, like ADT graph search, for the best possible local access points with enough fog serving nodes (Vertices) for better resource optimization with minimal edge distance among various nodes. It searches in proximity, i.e. the nearest and closest points such as those in Markov chain [20]. The fog node lies in between the IoT-edge devices and cloud, reducing the response time and service delays. As the data sent from user's edge-device towards fog node, if already availed by some other users, to avoid the congestion that leads to the maximum service delays or complex latency, data is being offloaded to the nearest available fog nodes for minimal delay or waiting time, or else to the cloud datacenters. As shown in Figure-5, the IoT edge devices send their requests to the local points via both the wired and wireless LANs. As long as the request is within the serving domain range, response time will be minimum (RL). Once the user's data goes out of the boundary, users can experience the maximum delays in response (CL). Fog nodes are placed in between the IoT-edge platform and centralized cloud platform, minimizing the Round-Trip Time (RTT) [21]. The FSWA minimizes the chances of the edge-devices generated data to get out of the boundary. Once the request from the edge-user is sent, it is directed towards the nearest available fog node, instead of being directed to the centralized cloud server, to reduce the service delay time. The interactions among the various fog neighbouring nodes occur both centrally as well as in distributed mechanism. In the central mechanism, an authentic fog node controls the other proximity nodes, whereas in the distribution fashion, the nodes are being controlled and the interaction is universal among all available fog nodes. There are several computing components being employed in fog platform and can be seen in day to day life aspects, including health centres, traffic and transportation (roads, railways, shipyards, aviation etc.), smart homes, and educational and government sectors. It is vital for fog components to work in coordination for proper and smooth function. Figure-3 demonstrates the latency and bandwidth offered by the fog and cloud domains. Here, RTT is a metric used to measure the two parameters the and calculated values. Table 1 shows that the Fog domain is having improved round-trip time while it is vice-versa in the cloud domain for cloud users [22].

5.1 Evaluation

Figure-5 demonstrates that the edge-enabled IoT device generates data and sends it to the nearest possible fog-node. The communication at the edge network is smoother and faster than that in the cloud datacenter. These IoT edge-devices are being inherited with inbuilt capabilities to operate without the intervention of cloud datacentres. Further, if the fog-nodes are not capable of providing the necessary resources, the requests are directed towards another fog-node (fog-node here is a mini server).

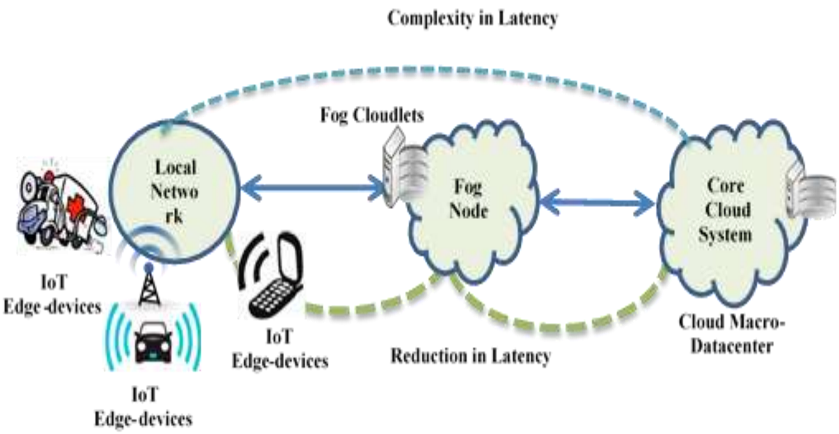

Figure 5-Fog-Edge nodes data flow in network topology.

As shown in Figure-6, FSWA will search the nearest available neighbouring fog-nodes. Those nodes within the boundary or proximity of edge-device can be traced quite fast, like when a spider catches prey in its web within the boundary. The available nodes will respond to the request of the edge-user. From various local positions, the IoT equipment can interact with fog-nodes in a hassle-free fashion where they enhance the searching capability and reduce the response delay time.

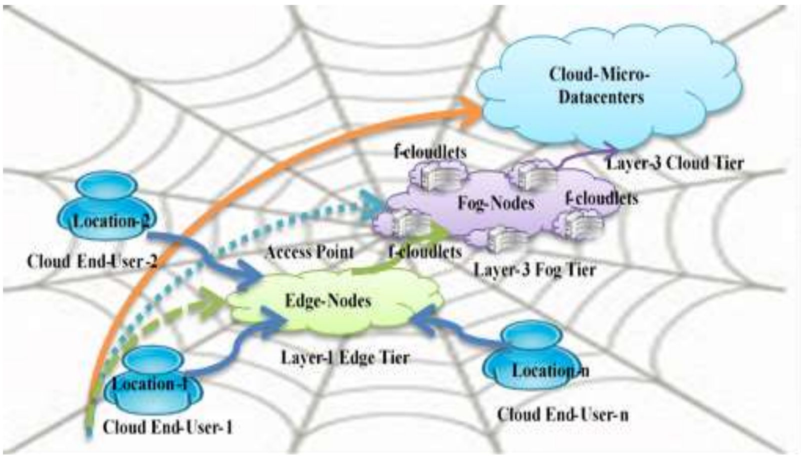

Figure 6-FSWA traversing the nearest

Figure-7 shows the visitation process among various IoT/edge-nodes and fog-nodes. The visitation from one node to another makes one hop. The hops count is the distance from edge-device to the nodes deployed in a fog. The decrease in hops count, leads to the low latency and enhances the searching for the desired resources. The nodes are dispersed geographically over the Fog domain, any requisition from edge-node will be served within the fog layer by foglets.

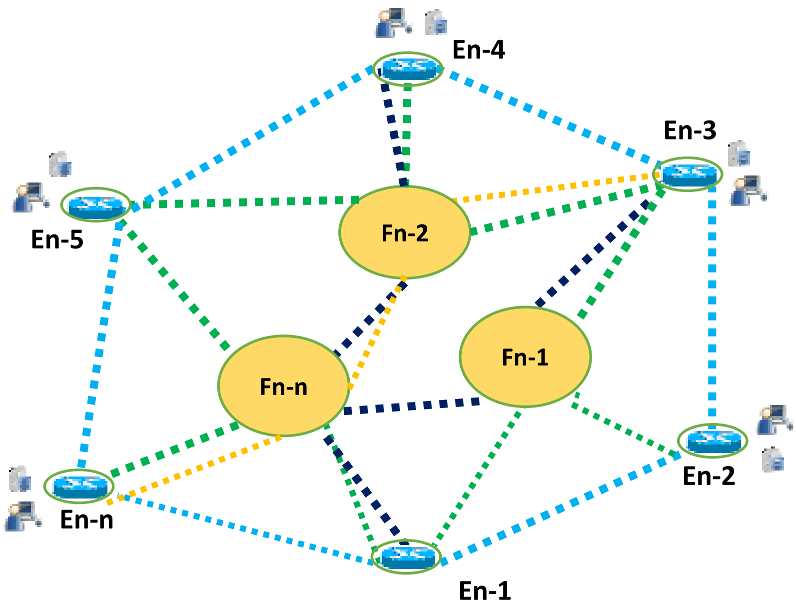

Figure 7-Fog-topology connecting various edge-nodes in proximity

Table 1- Shows the visitation of various edge and fog-nodes via FSWA

Notation	Visitation	Hops-count	$RT_{(ms)}$	$DT_{(ms)}$
……	$Edge_{node}$ to $Edge_{node}$ direct Visitation	$Hc=d_{e2e}$ (n)	Minimal, < d_{e2e} (n+1)	Minimal
……	$Edge_{node}$ to Fog_{node} direct Visitation	$Hc=d_{e2f}$ (n)	Minimal, < d_{e2f} (n+1)	Minimal
……	$Edge_{node}$ to $Edge_{node}$ indirect Visitation	$Hc=d_{e2e}$ (n+1)	Minimal, > d_{e2e} (n)	Minimal
……	$Edge_{node}$ to Fog_{node} indirect Visitation	$Hc=d_{e2f}$ (n+1)	Minimal, d_{e2f} (n)	Minimal

6. Conclusions

In this manuscript, we have precisely explained the decentralized fog computing paradigm and defined some of the similar and related de-centralized computing platforms, such as edge, mist, dew computing and its key components. We have described various fog characteristics, including the latency, and further highlighted the various parameters causing the high latencies in the cloud environment. Further, we proposed the fog-based Spider Web Algorithm with ADT-graphic traversal methodology to improve the latency and RRT interactions among various nodes in fog computing. FSWA searches the available proximal computing resources, reduces RT and DT, and enhances the latency over the edge network in IoT environment.

References

1. Aazam, M., Khan, I., Abdullah A. and Huh, E. **2014**. The cloud of Things: Integrating the Internet of Things and cloud computing and the issues involved. 414-419.
2. Ab Rashid D., Ravindran, D. **2019.** Fog Computing: An Efficient Platform for the Cloud-resource Management. *International Journal of Emerging Technologies and Innovative Research*. **6**(2):7-12.
3. Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J. and Jue, J. **2018**. All One Needs to Know about Fog Computing and Related Edge Computing Paradigms: A Complete Survey.1-48
4. Uehara, M. **2017.** Mist Computing: Linking Cloudlet to Fogs. **726**: 201-213.
5. Aazam, M., Huh, E. Fog Computing: The Cloud-IoT/IoE Middleware Paradigm. *IEEE Potentials*. **35**(3):40–44
6. Yousefpour, A., Ishigaki, G. and Jue, J. **2017.** Fog Computing: Towards Minimizing Delay in the Internet of Things. 17-24.

7. Ab Rashid D., Ravindran, D. **2019**. Fog Computing Resource Optimization: A Review on Current Scenarios and Resource Management. *Baghdad Science Journal*. **16**(2): 1-9.

8. Mir, M. and Ravindran, D. **2017.** LETISA: Latency optimal Edge computing Technique for IoT based Smart Applications. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology.* **2**(4): 688-694

9. Wang, Y. **2016**. Definition and Categorization of Dew Computing. *Open Journal of Cloud Computing.* **3**(1): 1-7.

10. Abrol, P., Gupta, S. and Kaur, K. **2016**. Analysis of Resource Management and Placement Policies Using a New Nature-inspired Metaheuristic SSCWA avoiding premature convergence in Cloud. *IEEE Xplore*.653-658.

11. Cuong, D., Tran, N., Pham, C., Alam, M., Hyeok S., Jae H. and Choong S. **2015**. A Proximal Algorithm for Joint Resource Allocation and Minimizing Carbon Footprint in Geo-distributed Fog computing. *IEEE Xplore*.324-329.

12. Gorlatova, M., Inaltekin, H. and Chiang, M. **2018**. Characterizing Task Completion Latencies in Fog Computing. *IEEE.*1-13.

13. Kejiang, Ye. & Xu, Cheng-Z. **2018**. Reducing Tail Latency of Interactive Multi-tier Workloads in the Cloud. *IEEE Xplore***.** 162-163.

14. Aazam, M. and Huh, E. **2014**. Fog Computing and Smart Gateway Based Communication for Cloud of Things. 464-470.

15. Hung, P., Aazam, M., Nguyen, T. and Huh, A. **2014**. A solution for optimizing recovery time in cloud computing. *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*. **1**: 1-8.

16. Yousefpour, A., Ishigaki, G., Gour, R.and Jue, J. **2018**. On Reducing IoT Service Delay via Fog Offloading. *IEEE Internet of Things Journal*. **5**(2): 988-1010.

17. Gu, L., Zeng, D., Guo, S. Barnawi, A. and Xiang, Y. **2015**. Cost-Efficient Resource Management in Fog Computing Supported Medical CPS. *IEEE Transactions on Emerging Topics in Computing*. **5**(99): 1-1.

18. Jung-Hoon L. Sang-Hwa C. and Won-Suk K. **2019**.Fog server deployment technique: An approach based on computing resource usage. *International Journal of Distributed Sensor Networks*. **15**(1): 1-19.

19. Haouari, F., Faraj, R. and M. AlJa'am, M. **2018**. Fog Computing Potentials, Applications, and Challenges. *IEEE Xplore*.399-406.

20. Tang, B., Chen, Z., Hefferman, G., Pei, S., Wei, T., He, H. and Yang, Q. **2017**. Incorporating Intelligence in Fog Computing for Big Data Analysis in Smart Cities. *IEEE Transactions on Industrial Informatics*.**13**(5): 2140-2150.

21. Jagtap, V. **2014**. Survey of Different Swarm Intelligence Algorithms. *International Journal of Advance Engineering and Research Development*. **1**(12): 86-90.

22. Shojaiemehr, B., Rahmani, A. and Qader, N. **2017**. Cloud Computing Service Negotiation: A Systematic Review. *Computer Standards & Interfaces*. 10.1016/j.csi.2017.08.006.