RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES

Social Science Journal

Improving Location Awareness in Internet of Things (IoT) Framework Using Fog Computing

By

Aarti Rani
Research Scholar, School of Computer Applications, Lucknow (Uttar Pradesh), India
Corresponding Author: aarti.singh18oct@bbdu.ac.in.

Vijay Prakash
Professor, School of Computer Applications, Lucknow (Uttar Pradesh), India

Manuj Darbari
Professor, Department of Information Technology, BBDNITM, Lucknow
(Uttar Pradesh), India

Abstract

Internet of Things a new technology that can be expressed as a smart, interoperable node associated to a global network infrastructure. IoT also aims to establish the interoperability concept of anything from everywhere at any time. IoT which has recently experienced rapid development and can provide a variety of services is now the fastest-growing technology and has a significant impact on both social life and commercial environments. Devices are often maintained by a cloud, and their number is constantly increasing. The IoT and cloud-connected gadgets are getting increasingly sophisticated and diversified. Furthermore, these devices are dispersed across wide geographic areas and create enormous volumes of data. Furthermore, the IoT frequently need a diverse set of service providers. It's possible that end consumers may need applications that are location-aware. Although the more centralized services offered by cloud, inclusive services, due to geography it can be difficult for cloud systems to maintain position with mobile and scattered devices. To address these difficulties, introduce a platform that enables the deployment of cloud like functions close to the devices that need to be maintained or observed. This is referred to as Fog computing. In addition to providing networking, storage, and computing capabilities, it links end-user devices to cloud providers. Fog enables minimum latency, location / context awareness, and low bandwidth and storage services. Data is managed locally to speed up response times. A cloud-based analysis platform can be used to send compiled data. The Fog can also provide services such as security, filtration, and so on. By combining the link state method with Dijkstra, the suggested Fog framework consistently meets an accurate location-awareness. Additionally, a comparison study of network utilization and latency is provided. The comparative findings show that, in contrast to cutting-edge frameworks, the LAFF decreases latency and network utilization. Therefore, the suggested LAFF enhances QoS while utilizing remote computational servers for the Fog computing applications that are outsourced.

Keywords: Internet of Things, Fog computing, Location awareness, Location awareness Fog Framework

1. Introduction

The term "internet of things" (IoT) refers to the proposal of ordinary physical objects being associated with the internet and having the capability to recognize other devices. IoT offers a new era of interconnected objects and people in which quality of life is increased due to easier administration of cities and infrastructure, accessibility to healthcare services, and the prevention of calamities. The envisioned Internet of Things embraces a broad diversity of different intelligent devices with features such as automated, awareness, flexibility, and zero configurations [1]. The IoT is among the most talked-about breakthroughs, with the potential to offer our society with incalculable advantages. The IoT might be about to reach a point where many of the items in our environment will be able to access the internet and interact with one another without the need for human involvement [2]. By employing a diverse range of sensors to gather data from the surroundings and automate data storage and processing, the Internet of Things was essentially created to decrease the need for human data entry [3]. This approach makes it possible to make new discoveries that will lead to cutting-edge human-object interactions as well as Smart urban, infrastructures, and services that will enhance superiority of life and resource efficiency. It allows data created by smart devices to be integrated, transferred, and analyzed (e.g. sensors). According to McKinsey, the Internet of Things may have an annual economic effect of 11 trillion dollars by 2025, which is comparable to nearly 11% of the global GDP. They also predict that by 2025, one trillion IoT devices will have been installed. IoT and cloud computing together provide significant benefits for different IoT applications. However, because there are so many IoT devices with different platforms, creating new IoT applications is tough [4]. Globally centralized cloud data centers may struggle to handle massively dispersed IoT devices without sacrificing QoS, even if they can participate in present and emerging applications [5].The majority of today's IoT data processing systems send data to the cloud to be processed. This happens due to existing data analytics methods are not intended for real-time data processing and dispatching; rather, they are developed to manage massive volumes of data. With millions of objects providing that data, putting it all to the cloud is hardly feasible and neither appropriate for real-time decision-making [6]. The emergence of the Fog is due to the dynamic nature of IoT settings, as well as the accompanying real-time needs and rising processing capabilities of edge devices. Fog brings workability of cloud to the edge of networks, reducing latency by geographically distributing IoT application components and enabling mobile mobility [7].The combination of IoT with Fog has created a new business opportunity, in which a service provider deploys a network of Fog nodes throughout its geographic footprint and operates as a owner to a variety of tenants from various vertical industries. Local computing, networking, and storage resources are available on every Fog node [8].

The purpose of Fog in the IoT is to increase competence and boost proficiency and productivity while reducing the volume of data handled in the cloud. As a result, network edge devices will process and temporarily store sensor data rather than uploading it to the cloud, reducing network traffic and delay [9]. IoT creates a lot of data, which puts a strain on the cloud and necessitates a lot of network capacity to make sure that without delay, data is transmitted and retrieved. The transmission of data and the diffusion of information has become a problem. Cloud services are now being used by both small and large businesses to safeguard and preserve their data. However, the problem arises when nodes are required to meet local delay requirements by latency-sensitive applications. Cloud computing only just permits mobility, low latency, and global dispersion in addition to location awareness. Because the Cloud cannot provide all of the QoS (Quality of Service) needs in IoT, a novel architecture is required. To lessen the pressure on the cloud, the new design necessitates a speedy reaction to the underlying device [10]. Furthermore, the centralized cloud strategy is ineffective for IoT applications with time-sensitive processes or insufficient Internet access. Milliseconds can be critical in a variety of situations, including telemedicine and medical care. The same situation applies to communications between vehicles, where the centralized cloud method's delay cannot be accepted in order to prevent collisions or accidents. To address these issues, an enhanced Cloud architecture that increases ability and latency restrictions is necessary [11].

An examination of the state-of-the-art, features, and advantages of Fog computing as it relates to the Internet of Things is presented in this paper as an overview. By focusing on the idea of location awareness, the integration of IoT and Fog computing is also examined.

In an effort to lessen data loads in the cloud and bring computer resources closer to consumers, Fog computing is a novel standard in computing. This is made possible by Fog nodes such base stations, sensors, and Fog servers. Fog enabled location awareness capabilities show that there are many devices in the real world with a small coverage area. As a result of the connection being activated, the location privacy of the user is exposed [12]. Like traditional cloud computing, Fog concept with constrained capabilities that distributes networking, processing, and storage services among a number of end devices. It's a strong choice for IoT applications that require minimum latency [13]. Fog Computing defined by OpenFog Consortium [14] as "a system-level horizontal architecture that distributes processing, storage, control, and networking resources and services anywhere along the continuum from Cloud to Things". In order to bring processing, storage, and networking resources nearer to the end devices, between the cloud and these gadgets, Fog serves as a bridge. This type of gadget is known as a Fog node. In any location with a network connection, they may be installed. Industrial devices, switches, gateways, embedded systems, and surveillance cameras can all be considered Fog nodes because they have computing, storage, and a network connection [15].Some Specifications and advantages of Fog Computing are:

Table1: *Specifications and advantages of Fog Paradigm*

Specifications of Fog computing	Fog computing advantages
Low latency and awareness of location	Greater Privacy
Regional distribution	Productivity
Extensibility	Reduced cost of Bandwidth
Support for mobility	Lower operating expense
Real-time interactions	Greater business agility
Heterogeneity	Improved response time
Interoperability	Enhanced Co0mpliance
a feature that supports online monitoring and interacts with the cloud	Overall increase in speed and Efficiency

For computation, storage, and processing, Fog architecture takes advantage of end-device services (switches, routers, multiplexers, and so forth). A large network of networked devices is created by combining hardware, software, and logical and physical network components as part of the Fog computing architecture. Fog architecture has several important topological and protocol properties, as well as physical and geographic dispersion of Fog nodes. Fog is a means of shifting portion of a data center's services to the network's edge. Between consumer devices and conventional cloud computing data centres, the Fog distributes limited computation, storage, and networking resources.

The main concept behind the Fog is to make available low latency for time-sensitive IoT applications [16].Three layered architecture in which the lowest layer contains individual IoT end-nodes. The middle of the model is the Fog layer, and it has Fog nodes with relatively independent decision-making capabilities. The most significant influence on Fog node performance comes from acquired traffic and interactions with the cloud layer. A cloud server can be found at the upper layer, which is referred to as the cloud layer [21].

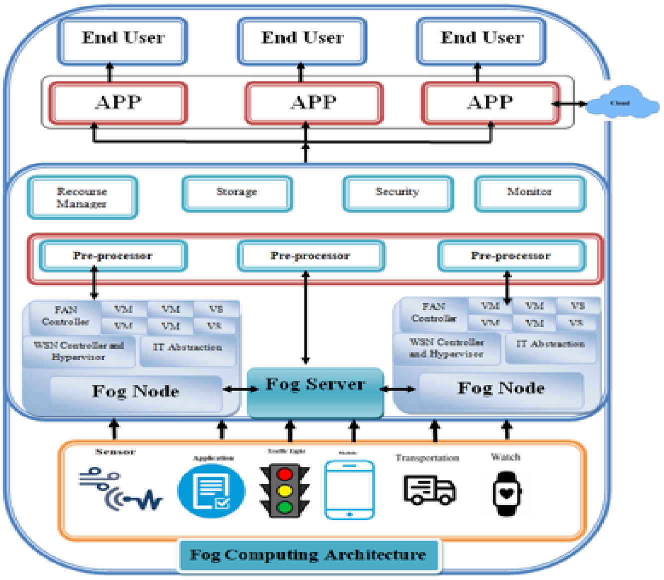


Fig.1. *Architecture of Fog Computing.*

Various Fog reference architectures have been established by different researchers. These designs are based on a number of topologies that are pertinent to user applications and services. The findings of Muntjir et al. [19], Ranesh et al. [17], Aazam and Huh et al. [18], and Mukherjee et al. [20] all support this. They suggested a framework for Fog paradigm with seven levels:

Fig.2. *Levels of Fog Computing Architecture*

2. Location Awareness

Location awareness is a feature of appearance awareness that notifies other user or application about a device's physical place. Applications that know where the device is might be required by end users. Automobiles, intelligent traffic lights, and other devices are a few examples. These applications might need information exact to their location. Although the cloud may provide more centralized and worldwide services, sustaining location awareness with mobile and dispersed devices can be challenging for cloud systems. Under the Fog computing framework, location awareness is used to minimize delay, processing times, and network use while utilizing the least amount of resources possible [22]. Ubiquitous computing makes use of the device's location to improve the user experience, and perhaps the most significant element is the ability to customize the facilities that are made accessible to the client, such as instantly locating and remembering other nearby devices, and then trying to offer support to the user after proper user authentication. The location context encompasses not just where a person is, but also who that client is and who else is in close proximity to that user. Such context can also contain the user's previous use history, allowing a universal system enabling location context to automatically manage device volume or notifications signals based about whether the user is in a populated area. The capacity of a device to directly or indirectly identify its position in respect of coordinates with reference to a basis of comparison is referred to as location awareness. To identify the position, many sensors or navigational instruments can be employed. A computer system's ability to recognize and react to a real-world situation is referred to as context-aware computing. According to this viewpoint, context encompasses a diversity of characteristics such as the Identification of the user, the user's current geographical position, the climate, the time of day, the weather, and whether the user is conscious or asleep, driving or strolling. Location and identity are the most important factors of context Systems that are aware of a user's position can respond to it in one of two ways: automatically (for example, by alerting them to a nearby hazard) or in response to a user request. Numerous applications for location awareness exist, such as asset monitoring, ground surveying, navigation, and emergency response. Event is peroxide by symbolic place. Collation may be used to learn social roles and interactions, Media access control address of a device in a Wi-Fi network, Coordination of change can refer to both activity and method of movement [23]. Here are a few examples of location awareness applications:

Navigation:
Aviators, mariners, and professional drivers are all concerned with direction and reckoning. The work entails determining the current position, as well as the range, route, and time to the goal, in real time.

Surveying:
It's the navigation's static partner. It is necessary for determining property ownership as well as civil engineers and architects who plan development projects.

- Software programmes
- Vehicle-based GPS systems
- Device management in healthcare.

- Camera memory cards that automate the location tagging of a photograph
- Supply chain management (SCM)
- Routing and storage

Fig.3. *Location awareness applications*

Applying the concept of minimal privilege through location-aware access control involves making some resources accessible only from specified locations [24]. According to Usman Khan[25] "In the future, the necessity to offer position awareness for any device, sensor, or vehicle, whether fixed or moving, will become increasingly significant. Healthcare, security, agriculture, environmental research, military operations, emergency response, industrial automation, self-driving cars, robots — the list goes on and on. The Internet of Things' practically infinite potential necessitates the development of sophisticated decentralized algorithms."

3. Related Work

Data is sent to a local node rather than the cloud when it is gathered by IoT devices and Fog computing resources. By employing Fog nodes closer to the data source, queries may be processed more quickly than by sending they back to data centres for analysis and action. A large, dispersed network would include Fog nodes placed at various key locations to allow for the retrieval and local processing of crucial information. Location awareness requires a sense of where are the devices. Intelligent systems have a significant capability called location awareness. It entails the use of context information, or data about a user's environment, behavior, and surroundings, to improve the delivery of services in support of context-aware apps. Location awareness gives pertinent information about the present state of the environment and permits adaptive action in response to environmental changes. The notion of location awareness first originated in the realm of ubiquitous computing, in which devices and sensors are dispersed across an environment. [26].

IoT localization [28] solutions that have been proposed. This article begins by explaining the most important KPIs for localization. Then, based on the KPIs, a full comparison of recently suggested localization strategies is offered. An overview, architecture, network topology, performance characteristics, and desired KPIs are all included in the comparison [27. Improved location awareness and multi-layer node management are provided in the DISSECT-CF-Fog simulation to assist the execution of IoT applications. Data is transferred immediately to the distant cloud server using the developed Framework, without any prior processing [29].

Using Cloud Foundry's role-based access control, the author [30] constructed location-aware access policies and created a model for location-aware accessing control and data security for CC systems. Using RFID technology, this research proposes a cost-effective solution to such security challenges. Because of its simplicity and intangibility with the item, RFID applications have exploded in popularity in recent years. The primary concept of this piece is that the subject cannot be in two locations at the same time. This strategy [31] allows us to keep just the appropriate people in the appropriate places. The system has the potential to improve resource governance. The set of EIDs identifying contact events is the essential tracking information in the CoViD-19 application scenario. Both DP3T and PEPP-PT have significant privacy features since they capture contact events rather than user IDs, as a traditional LBS would, furthermore to hard time of day information. Because of this, the communication through spatial rendering monitoring is specific to the user's moving reference frame [32].

To offer the best level of protection, provide a location-aware flexible dual-differential privacy protection technique [33]. The suggested location-aware Fog framework consistently satisfies QoS requirements by providing a precise location-aware algorithm. The proposed framework [34] for fog computing improves QoS when accessing remote computational servers for external application. In this work, author propose [35] a decentralized SOA framework relies location-aware detection approach for the Internet of Things called Swarm. We created the Swarm model to describe the localization information of services in a network of Internet of Things devices. The Swarm employs Semantic Web technologies for device communication. We build a system [36] that safeguards privacy to anticipate missing QoS values, enabling us to provide Web Service suggestions based on previous QoS experiences and user locations. Our protocol achieves user privacy by picking relevant Web Services without disclosing any personal information, encrypting the QoS and location, and more. Fog computing platform [37] based on UAVs for Internet of Things applications. To successfully enable dynamic IoT applications, Fog capabilities may be quickly deployed in remote or difficult places. Suggest that managing the shift to location awareness is a good idea [38]. By placing state-related components and processing power on temporary servers at the network edge, fog systems lessen the impact of this. While consistency processes are successful even when deploying fragments at different places, each fragment may be deployed at the most useful surrogate, which improves performance. In this work [39], a spatially aware state deployment service is created by integrating the state deployment algorithm with sustainable distribution support architecture. Researcher suggests [40] Splendor, a brand-new service discovery paradigm that supports mobile consumers and services in public settings. Splendor prioritizes privacy and security. In order to reduce the network infrastructure requirements for service discovery, location awareness are incorporated for location-dependent services discovery. They review the Splendor system's functionality and present our experimental findings. A popular approach [41] for managing dispersed services, devices, and users effectively and flexibly combines sophisticated agent technology with service discovery. Authors examine the DF's searching strategy in this research and create a build a widespread location-based service in a context-aware setting as this setting provides an excellent way to gather necessary data. Location-based service in a context-aware environment since this atmosphere offers a good means to acquire required information. Splendor is a brand-new service discovery paradigm that helps mobile users and services in public situations, according to the researcher. Splendor places a high priority on safety and privacy. For location-dependent service discovery, location awareness is implemented to lessen the load on the communications infrastructure. They provide the results of our analysis and explain about how the respective systems work.

A popular approach for managing dispersed services, devices, and users effectively and flexibly combines sophisticated agent technology with service discovery. The authors of this work analyze the DF's search technique and develop a widely used location-based solution in a context-aware environment since it offers an easy approach to obtain the required data.

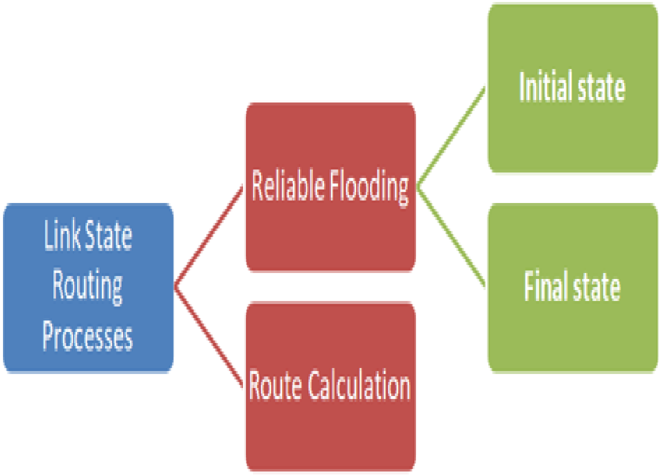
4. Location Awareness Based Fog Framework

Because of recent improvements in mobile computing, location-based services have grown increasingly popular in recent years. Location awareness stands for location-based services, which refers to the providing of services based on geographic location of different devices. . Fog computing is included into the proposed location awareness framework (LAFF) to minimize resource consumption while reducing latency, service time, and network usage. A location-aware algorithm used by LAFF can pinpoint the precise position of various Fog nodes. The proposed Framework basically has two steps. In the first step link state routing protocol is used and in second step finding shortest path use Dijikstra's Algorithm. Information exchange between nodes allocated by the routing protocol is the task of the node. Additionally, it aids in the transmission of data packets and information between nodes. The nodes are directly connected to the network and are aware of the full network structure thanks to the routing protocol.

4.1To find the shortest path for location awareness using link state routing and dijkstra's algorithm

By use of the link state routing method, every router in the network shares information about its neighborhood with every other router. Every router uses the Link State Routing Algorithm, an internal protocol, to communicate information or knowledge about the other routers on the network. Every router computes its routing table using a distributed technique called the link state routing algorithm. A router can create its routing table using information of the network topology. Each router builds a routing table that is shared with the other routers in the network, enabling quicker and more dependable data transmission. Only if there is a change in the information does this exchange of information occur. Consequently, the routing of link states method works well. The optimum route from the source to the destination is provided by a routing protocol, which is a routing algorithm.

4.2 Link State Routing Processes



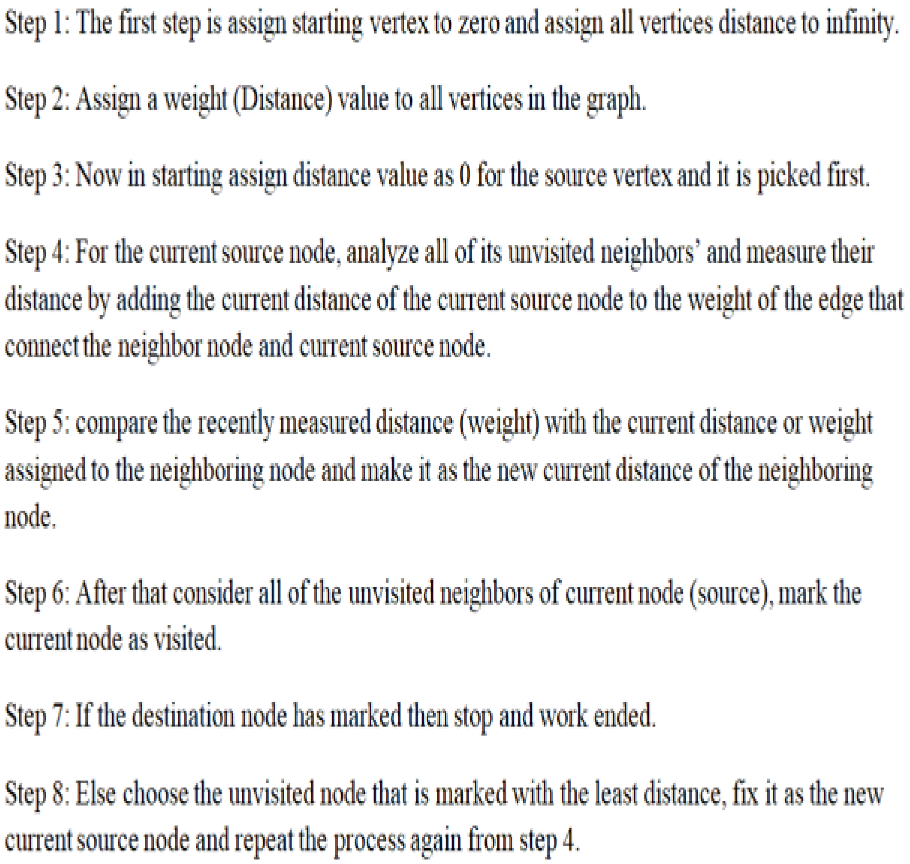
a) Reliable Flooding

- Initial state: Every node initially understands the value of its neighbors.
- Final state: The entire graph is known to each node.

b) Route Calculation

Dijkstra's method is applied to the network by each node to determine the best paths between all other nodes. The shortest path algorithm used here is single-source. The Dijkstra is a graph traversal method. In a computer network, there are transmitter and receiver nodes. Certain frames or messages are transmitted from the sender to the receiver, but by the time the recipient is able to receive the message, it has already experienced a number of iterations. It will choose the route that travels the fewest distances between sender and recipient.

Algorithm:



Step 1: The first step is assign starting vertex to zero and assign all vertices distance to infinity.

Step 2: Assign a weight (Distance) value to all vertices in the graph.

Step 3: Now in starting assign distance value as 0 for the source vertex and it is picked first.

Step 4: For the current source node, analyze all of its unvisited neighbors' and measure their distance by adding the current distance of the current source node to the weight of the edge that connect the neighbor node and current source node.

Step 5: compare the recently measured distance (weight) with the current distance or weight assigned to the neighboring node and make it as the new current distance of the neighboring node.

Step 6: After that consider all of the unvisited neighbors of current node (source), mark the current node as visited.

Step 7: If the destination node has marked then stop and work ended.

Step 8: Else choose the unvisited node that is marked with the least distance, fix it as the new current source node and repeat the process again from step 4.

5. Methodology

The Dijkstra is a graph traversal method. In a computer network, there are transmitter and receiver nodes. The sender transmits certain frames or messages to the receiver, however, the message has gone through multiple versions even by period the recipient is ready to receive it. The quickest route to take the message from sender to receiver will be discovered. Consider the network structure shown below, which has nodes from A to H. We must look at the shortest route between A and D, with A acting as the sender and D as the recipient.

```
Algo=LAFF (Location awareness Fog Framework).
Input=No of Nodes, Weights
Output=Shortest path to the nearest load free node.
Start
Step1: Use depth first traversal method to find all the possible routes from each node.
Step2: Print table or data frame of all vertices and routes within those.
Step3: Create a graph using the information from step 2 to visualize the topology.
Step4: Determine the load-free minimum path between the source and destination nodes.
```

Stop

Consider a network structure for calculating shortest path, the figure shows the connections between F1 (Fog Node 0) to F6 (Node 5). In this scenario the base position node is F1, while the destination node is F6. Now we need to consider which route is the shortest.

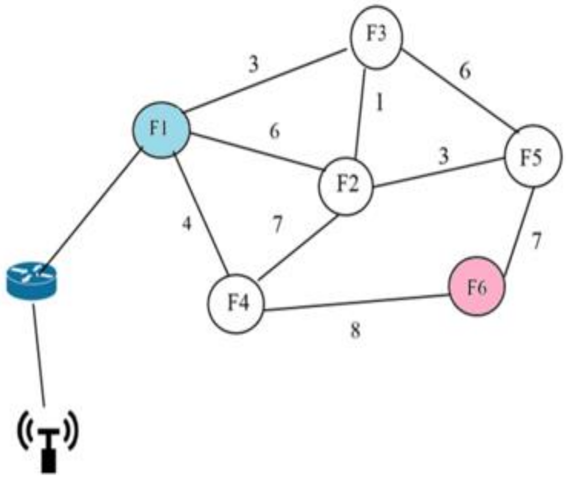
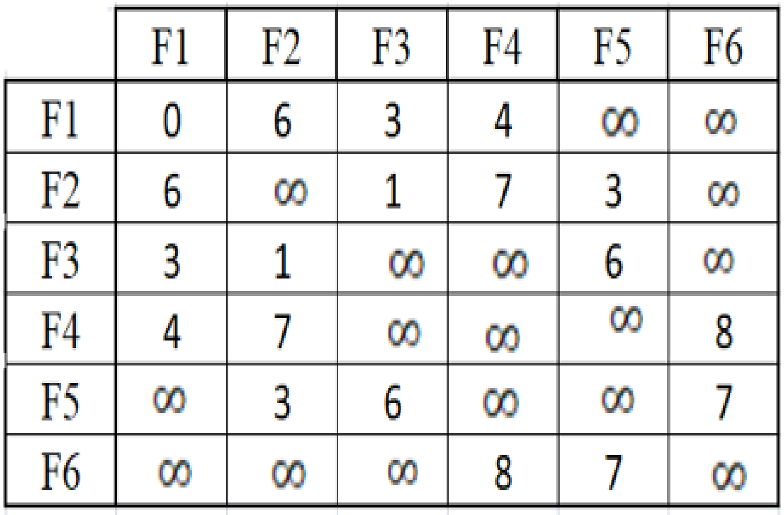


Fig.5.*Graph for Node Representation*

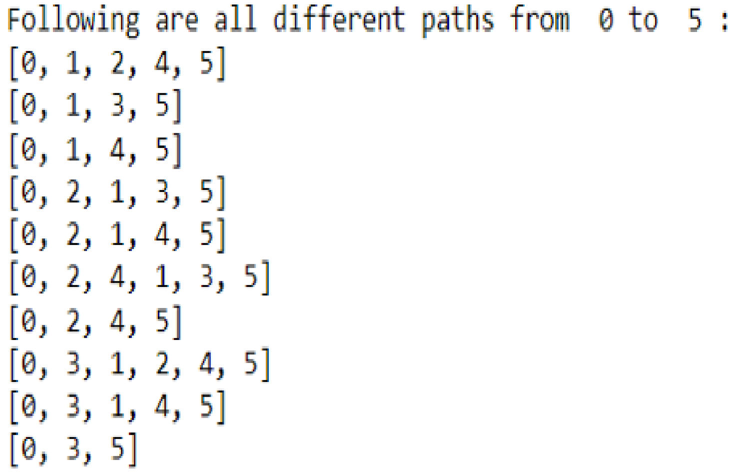
Every route's visited and unvisited nodes list were noted in a table to determine the shortest path.

Table 2: *Shows Traversal of nodes*



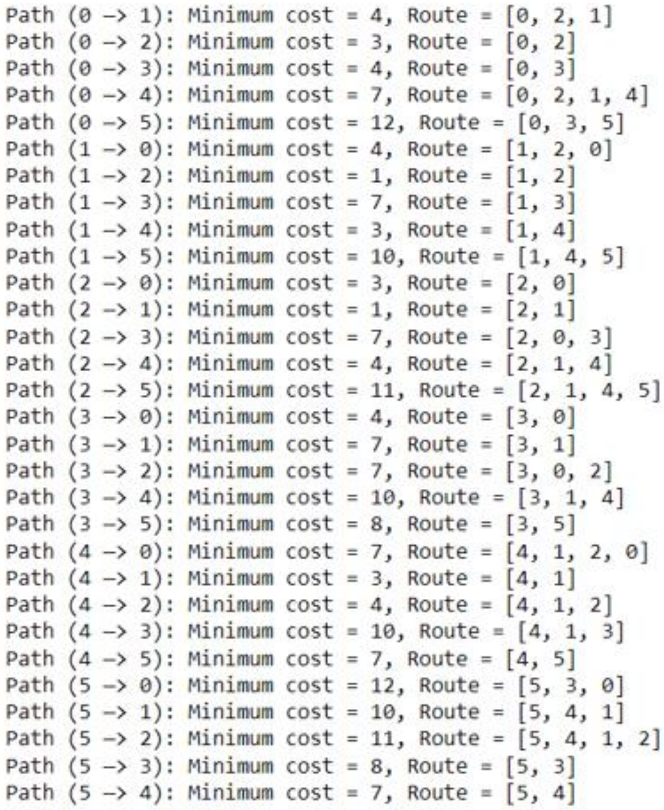
	F1	F2	F3	F4	F5	F6
F1	0	6	3	4	∞	∞
F2	6	∞	1	7	3	∞
F3	3	1	∞	∞	6	∞
F4	4	7	∞	∞	∞	8
F5	∞	3	6	∞	∞	7
F6	∞	∞	∞	8	7	∞

Table 3: *Shows routes from source to destination*



```
Following are all different paths from  0 to  5 :
[0, 1, 2, 4, 5]
[0, 1, 3, 5]
[0, 1, 4, 5]
[0, 2, 1, 3, 5]
[0, 2, 1, 4, 5]
[0, 2, 4, 1, 3, 5]
[0, 2, 4, 5]
[0, 3, 1, 2, 4, 5]
[0, 3, 1, 4, 5]
[0, 3, 5]
```

Table 4: *Shows routes from source to destination*



```
Path (0 -> 1): Minimum cost = 4, Route = [0, 2, 1]
Path (0 -> 2): Minimum cost = 3, Route = [0, 2]
Path (0 -> 3): Minimum cost = 4, Route = [0, 3]
Path (0 -> 4): Minimum cost = 7, Route = [0, 2, 1, 4]
Path (0 -> 5): Minimum cost = 12, Route = [0, 3, 5]
Path (1 -> 0): Minimum cost = 4, Route = [1, 2, 0]
Path (1 -> 2): Minimum cost = 1, Route = [1, 2]
Path (1 -> 3): Minimum cost = 7, Route = [1, 3]
Path (1 -> 4): Minimum cost = 3, Route = [1, 4]
Path (1 -> 5): Minimum cost = 10, Route = [1, 4, 5]
Path (2 -> 0): Minimum cost = 3, Route = [2, 0]
Path (2 -> 1): Minimum cost = 1, Route = [2, 1]
Path (2 -> 3): Minimum cost = 7, Route = [2, 0, 3]
Path (2 -> 4): Minimum cost = 4, Route = [2, 1, 4]
Path (2 -> 5): Minimum cost = 11, Route = [2, 1, 4, 5]
Path (3 -> 0): Minimum cost = 4, Route = [3, 0]
Path (3 -> 1): Minimum cost = 7, Route = [3, 1]
Path (3 -> 2): Minimum cost = 7, Route = [3, 0, 2]
Path (3 -> 4): Minimum cost = 10, Route = [3, 1, 4]
Path (3 -> 5): Minimum cost = 8, Route = [3, 5]
Path (4 -> 0): Minimum cost = 7, Route = [4, 1, 2, 0]
Path (4 -> 1): Minimum cost = 3, Route = [4, 1]
Path (4 -> 2): Minimum cost = 4, Route = [4, 1, 2]
Path (4 -> 3): Minimum cost = 10, Route = [4, 1, 3]
Path (4 -> 5): Minimum cost = 7, Route = [4, 5]
Path (5 -> 0): Minimum cost = 12, Route = [5, 3, 0]
Path (5 -> 1): Minimum cost = 10, Route = [5, 4, 1]
Path (5 -> 2): Minimum cost = 11, Route = [5, 4, 1, 2]
Path (5 -> 3): Minimum cost = 8, Route = [5, 3]
Path (5 -> 4): Minimum cost = 7, Route = [5, 4]
```

Optimal route between nodes F1 to node F6. Last but not least, the quickest route is
F1 → F4 → F6.

Additionally, the shortest path's weight is 4+8=12.

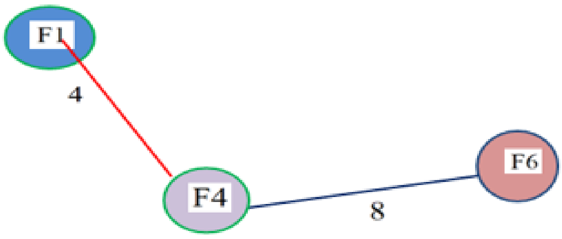


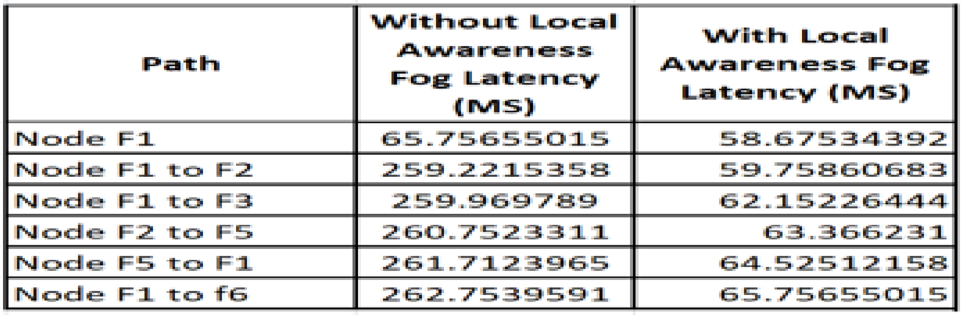
Fig.6.*Graph for Shortest path between 3 Nodes*

Technology that can determine the location of a person, device, or other moving items comprises sensors and ways for doing so. Location awareness is the ability of a device to actively or passively ascertain its coordinates in relation to a point of reference. The state link routing method with Dijkstra offers considerable benefits, including a quick convergence, frequent updates, Link State Protocols, and comprehension of the network. When it comes to IoT application work allocation challenges in cloud and fog environments, location awareness is crucial. Link-state protocols function best when: A hierarchical network design is present, which is typical in big networks. The implemented link-state routing protocol is well known to the administrators. For the purpose of determining the shortest path, each router in link state creates a topological map of the network on its own.

6. Experimental Setup

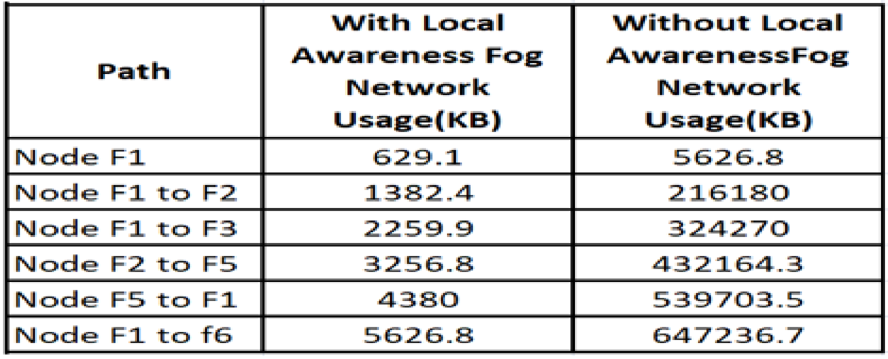
We created a scenario to demonstrate the importance of location awareness. In this part, the results from the suggested fog-based framework for improving location awareness are shown and contrasted with cloud-based deployment in conditions of reducing latency and optimizing network utilization. In comparison to the cloud-based approach, experimental findings indicate that the Fog-based deployment delivers improved location awareness with low latency and minimal network utilization. Table 4 displays the network usages and latency with and without location awareness findings for the Fog environment as well as the outcomes of the cloud-based deployment. To determine the findings, we used iFogSim to run the Table 5 and 6 scenarios. Our major goal was to reduce latency through network utilization analysis and location awareness. The comparison of location awareness regarding improvement in latency for both the cloud and the Fog context is shown in graphs. The consequences reveal that the latency of the cloud considerably rises as the number of nodes rises, but the fog's latency stays relatively constant. When there are more nodes in the Fog, only then does the Fog node allocated for that exacting place analyze the air data for that region. The cloud server analyses the sensed data from every place in the case of cloud, as a result, the latency inevitably grows as the number of nodes rises. The findings of network use for the Fog scenario are shown in Table 6. The testing outcomes for the two assessment factors of latency and network uses for both the Fog and cloud-based implementations show the efficacy of the suggested architecture inspired on fog for air quality monitoring system. Figure 7 and Figure 8 showing the topology of different comparatively to the cloud-based deployment two scenarios. Fog-based architecture for air quality monitoring systems will provide information about nodes that can process data promptly and cut down on the time spent looking for processing nodes. The findings also make it possible for us to see how fog computing can work in IoT settings where shorter response times are essential. Due to its low latency and reduced network utilization, the fog-based structural design is excellent for real-time events and applications.

Table 5: *Comparison graph of with and without Local awareness*

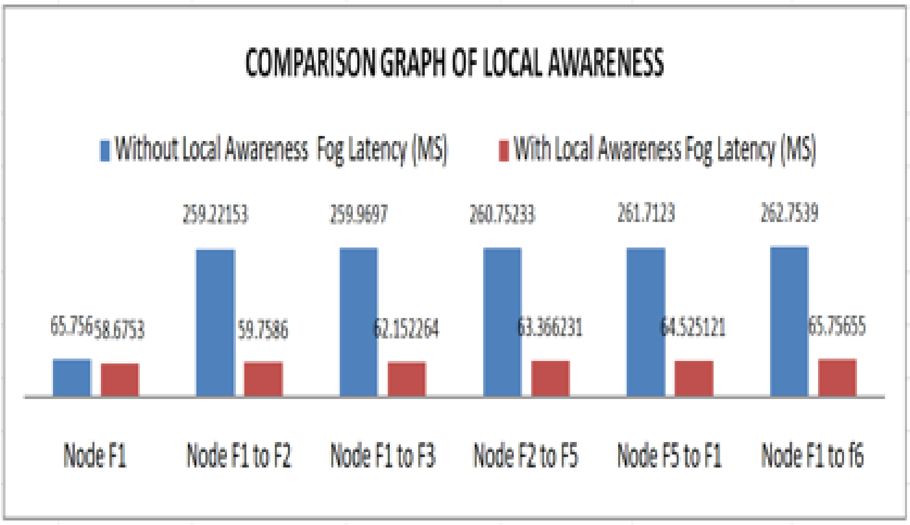


Path	Without Local Awareness Fog Latency (MS)	With Local Awareness Fog Latency (MS)
Node F1	65.75655015	58.67534392
Node F1 to F2	259.2215358	59.75860683
Node F1 to F3	259.969789	62.15226444
Node F2 to F5	260.7523311	63.366231
Node F5 to F1	261.7123965	64.52512158
Node F1 to f6	262.7539591	65.75655015

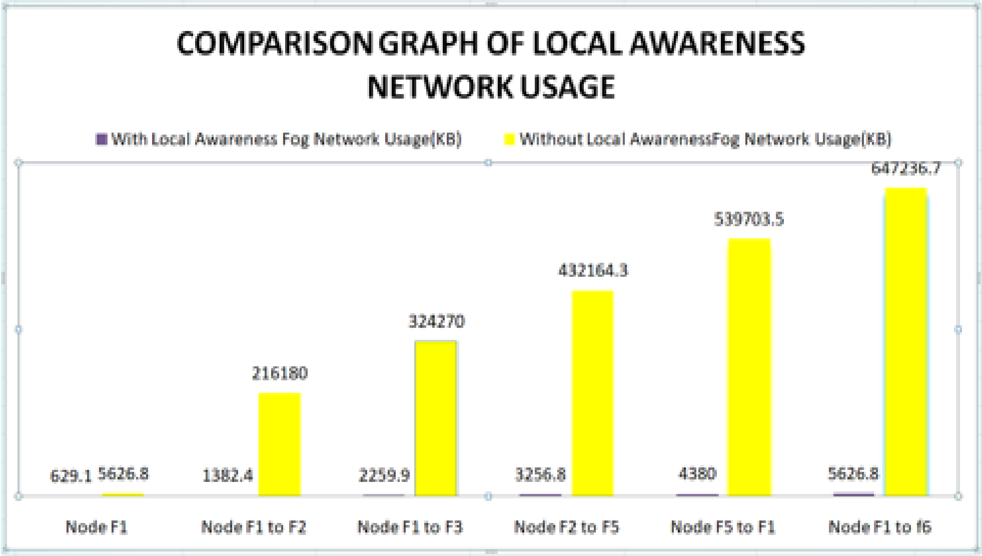
Table 6: *Comparison graph of network usages with and without Local/Location awareness*



Path	With Local Awareness Fog Network Usage(KB)	Without Local AwarenessFog Network Usage(KB)
Node F1	629.1	5626.8
Node F1 to F2	1382.4	216180
Node F1 to F3	2259.9	324270
Node F2 to F5	3256.8	432164.3
Node F5 to F1	4380	539703.5
Node F1 to f6	5626.8	647236.7



Graph 1: *Comparison graph of location awareness in form of latency*



Graph 2: *Comparison graph of improved location awareness in form of network Usage*

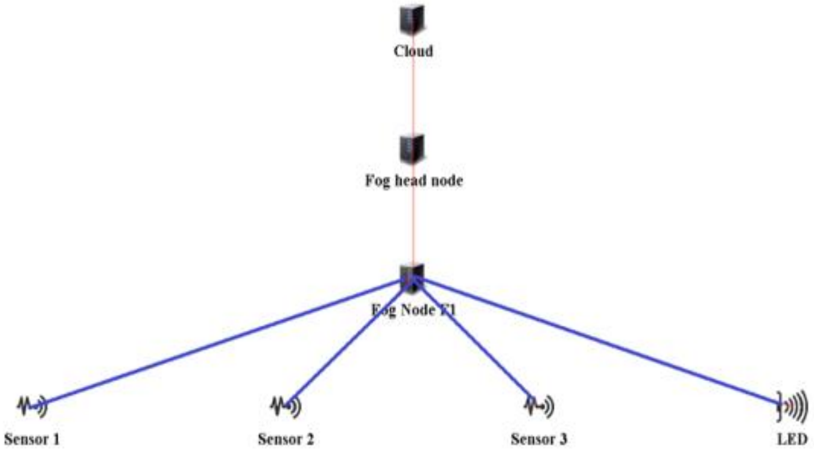


Fig 7: *IFogsim Topology Fog Node F1 connected with 3 Sensors and one LED*

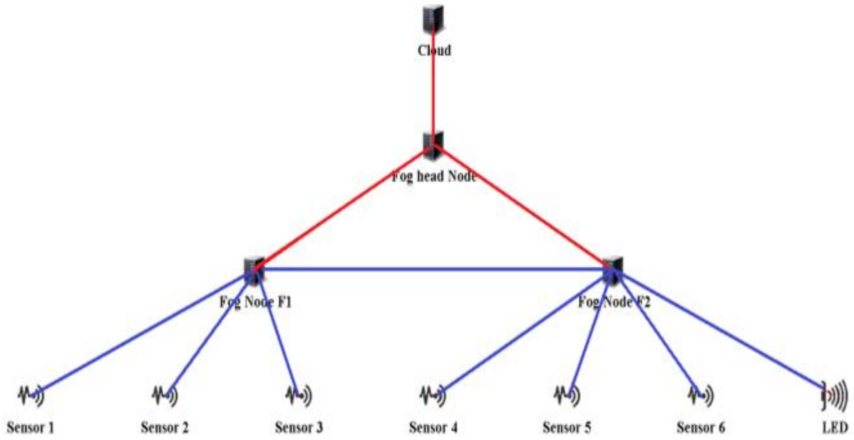


Fig 8: *IFogsim Topology Fog Node F1 and F2 connected with 6 Sensors and one LED*

7. Conclusion

Location awareness describes a presence-technology component that communicates to another application or user information about the precise location of a device. By lowering network use, service time, and latency, the location-aware algorithm is meant to provide rich user experiences and higher QoS. As a result, the recommended LAFF Therefore, while leveraging remote computational servers for the contracted Fog computing applications, the suggested LAFF enhances quality of services. QoS when using remote computational servers for the contracted Fog computing applications. Link State Routing Protocol, which is employed in the development of bigger networks. We focused on link state and Dijkstra's algorithm in this research since they are two of the most appropriate techniques for determining the shortest path for the supplied vertices. The system we propose for load balancing in Fog will be developed further in the future.

References

G. M. Lee, K. J. Yun , ”The internet of things: A problem statement”, IEEE International Conference on Information and Communication Technology Convergence (ICTC), pp. 517518. , November 2010

Atlam, H.F.; Alenezi, A.; Walters, R.J.; Wills, G.B. An Overview of Risk Estimation Techniques in Risk-based Access Control for the Internet of Things. In Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security (IoTBDS 2017), Porto, Portugal, 24–26 April 2017; pp. 254–260

Giang, N.; Kim, S.; Kim, D.; Jung, M.; Kastner, W. Extending the EPCIS with Building Automation Systems: A New Information System for the Internet of Things. In Proceedings of the 2014 Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Birmingham, UK, 2–4 July 2014; pp. 364–369.

Ai, Y.; Peng, M.; Zhang, K. Edge cloud computing technologies for internet of things: A primer. Digit. Commun. Netw. 2017, in press.

N. Kumar, J. J. P. C. Rodrigues, M. Guizani, K. R. Choo, R. Lu, C. Verikoukis, and Z. Zhong, “Achieving energy efficiency and sustainability in edge/Fog deployment,” IEEE Communications Magazine, vol. 56, no. 5, pp. 20–21, May 2018.

Gupta, H., Vahid Dastjerdi, A., Ghosh, S. K., & Buyya, R. (2017). iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of

Things, Edge and Fog computing environments. Software: Practice and Experience, 47(9), 1275-1296.

Chang C, Srirama SN, Mass J. A middleware for discovering proximity-based service-oriented industrial internet of things. Services Computing (SCC), 2015 IEEE International Conference on, IEEE, 2015; 130–137.

Yang, Y. FA2ST: Fog as a Service Technology. In Proceedings of the 2017 IEEE 41st IEEE Annual Computer Software and Applications Conference, Turin, Italy, 4–8 July 2017; p. 708.

Wen, Z.; Yang, R.; Garraghan, P.; Lin, T.; Xu, J.; Rovatsos, M. Fog orchestration for internet of things services. IEEE Internet Comput. 2017, 21, 16–24

Internet of things market forecast:. Cisco. [Online]. Available: http:// postscapes. com / internet-of-things-market-size

Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog computing and its role in the internet of things. In Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing-MCC ’12, Helsinki, Finland, 17 August 2012; pp. 13–15

Gu, B., Wang, X., Qu, Y., Jin, J., Xiang, Y., & Gao, L. (2020, October). Location-aware privacy preserving scheme in SDN-enabled Fog computing. In International Conference on Security and Privacy in Digital Economy (pp. 176-190). Springer, Singapore.

Verma, M.; Bhardwaj, N.; Yadav, A.K. Real Time Efficient Scheduling Algorithm for Load Balancing in Fog Computing Environment. Int. J. Inf. Technol. Comput. Sci. 2016, 8, 1–10.

Definition of Fog Computing. Available online: https://www.openFogconsortium.org/#definition-of-Fogcomputing (accessed on 24 March 2018).

Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are. White Paper. 2016. Available online: http://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf (accessed on 8 April 2018).

Shi, Y.; Ding, G.; Wang, H.; Roman, H.E.; Lu, S. The Fog computing service for healthcare. In Proceedings of the 2015 2nd International Symposium on Future Information and Communication Technologies for Ubiquitous HealthCare (Ubi-HealthTech), Beijing, China, 28–30 May 2015; pp. 1–5.

Naha, R. K., Garg, S., Georgakopoulos, D., Jayaraman, P. P., Gao, L., Xiang, Y., & Ranjan, R. (2018). Fog computing: Survey of trends, architectures, requirements, and research directions. IEEE access, 6, 47980-48009.

Aazam, M., & Huh, E. N. (2015, March). Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT. In 2015 IEEE 29th International Conference on Advanced Information Networking and Applications (pp. 687-694). IEEE.

Muntjir, M., Rahul, M., & Alhumyani, H. A. (2017). An analysis of Internet of Things (IoT): novel architectures, modern applications, security aspects and future scope with latest case studies. Int. J. Eng. Res. Technol, 6(6), 422-447.

Mukherjee, M., Shu, L., & Wang, D. (2018). Survey of Fog computing: Fundamental, network applications, and research challenges. IEEE Communications Surveys & Tutorials, 20(3), 1826-1857.

Maiti, P., Sahoo, B., & Turuk, A. K. (2022). Low Latency Aware Fog Nodes Placement in Internet of Things Service Infrastructure. Journal of Circuits, Systems and Computers, 31(01), 2250017.

Ikpehai, A.; Adebisi, B.; Rabie, K.M.; Anoh, K.; Ande, R.E.; Hammoudeh, M.; Gacanin, H.; Mbanaso, U.M. Low-Power Wide Area Network Technologies for Internet-of-Things: A Comparative Review. IEEE Internet Things J. 2019, 6, 2225–2240.

Ahmadi, Z., Haghi Kashani, M., Nikravan, M., & Mahdipour, E. (2021). Fog-based healthcare systems: A systematic review. Multimedia Tools and Applications, 1-40.

Gajjar, Manish J. (2017). Mobile Sensors and Context-Aware Computing || Context-aware computing. , (), 17–35. doi:10.1016/B978-0-12-801660-2.00002-1

André Van Cleeff, Wolter Pieters, and Roel Wieringa.Benefits of location-based access control: A literaturestudy. In Green Computing and Communications(GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'lConference on Cyber, Physical and Social Computing(CPSCom), pages 739–746, 2010.

Available from https://incompliancemag.com/improving-location-awareness-for-the-internet-of-things/

Available from http://www. zeitgeis tlab.ca / doc/ context_and_self_awareness_in_Fog_and_mist_computing.html

Khan, M. A., Saboor, A., Kim, H. C., & Park, H. (2021). A Systematic Review of Location Aware Schemes in the Internet of Things. Sensors, 21(9), 3228.

Markus, A., Dombi, J. D., & Kertesz, A. (2021, March). Location-aware Task Allocation Strategies for IoT-Fog-Cloud Environments. In 2021 29th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP) (pp. 185-192). IEEE.

Bhadoria, R. K., Saha, J., Biswas, S., & Chowdhury, C. (2021). IoT-based location-aware smart healthcare framework with user mobility support in normal and emergency scenario: a comprehensive survey. Healthcare Paradigms in the Internet of Things Ecosystem, 137-161.

Zeng, W., Bashir, R., Wood, T., Siewe, F., Janicke, H., & Wagner, I. (2020). How location-aware access control affects user privacy and security in cloud computing systems.

Azhaguramyaa, V. R., & Srinivasan, K. (2020, February). Location-aware security system for smart cities using iot. In International Conference on Advanced Machine Learning Technologies and Applications (pp. 559-569). Springer, Singapore.

Schmidtke, H. R. (2020). Location-aware systems or location-based services: a survey with applications to CoViD-19 contact tracking. Journal of Reliable Intelligent Environments, 6(4), 191-214.

Gu, B., Wang, X., Qu, Y., Jin, J., Xiang, Y., & Gao, L. (2020, October). Location-aware privacy preserving scheme in SDN-enabled Fog computing. In International Conference on Security and Privacy in Digital Economy (pp. 176-190). Springer, Singapore.

Shaheen, Q., Shiraz, M., Hashmi, M. U., Mahmood, D., & Akhtar, R. (2020). A lightweight location-aware Fog framework (LAFFF) for QoS in Internet of Things paradigm. Mobile Information Systems, 2020.

Badsha, S., Yi, X., Khalil, I., Liu, D., Nepal, S., Bertino, E., & Lam, K. Y. (2018). Privacy preserving location-aware personalized web service recommendations. IEEE Transactions on Services Computing.

Mohamed, N., Al-Jaroodi, J., Jawhar, I., Noura, H., & Mahmoud, S. (2017, August). UAVFog: A UAV-based Fog computing for Internet of Things. In 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/ SCALCOM/ UIC/ATC/CBDCom /IOP/SCI) (pp. 1-8). IEEE.

Cooper, A., Mulligan, D., Schulzrinne, H., & Wilde, E. (2010). Challenges for the location-aware web.

Lima, D., & Miranda, H. (2022). A geographical-aware state deployment service for Fog Computing. Computer Networks, 216, 109208.

Zhu, F., Mutka, M., & Ni, L. (2003, March). Splendor: A secure, private, and location-aware service discovery protocol supporting mobile services. In Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003.(PerCom 2003). (pp. 235-242). IEEE.

Na, S. J., Kim, D. U., Yoon, Y. J., & Shin, D. R. (2008, February). Design of location-based directory facilitator in context-aware environment. In 2008 10th International Conference on Advanced Communication Technology (Vol. 3, pp. 2107-2110). IEEE.