Scheduling Models in Fog Computing

Abstract

A new computing paradigm known as "Fog computing" has been established as a result of the significant growth in the number of Internet of Things (IoT) devices and sensors in recent years. Because many IoT devices and sensors require latency-aware computation for real-time application processing, big data is the term for the enormous volume of data that connected objects in IOT environments produce. Data gathered by IOT devices is often handled on a cloud infrastructure because of the on-demand services and scalability capabilities of the cloud computing paradigm. However, for some IOT applications, executing requests only on the cloud is not an effective approach especially time-sensitive one.

Resource management and task scheduling are two of the main difficulties encountered when operating IOT apps in a fog computing environment. The main feature of the Fog Computing architecture is that it bypasses the larger Internet in favor of providing computing and data analytics services more quickly and close to the actual physical devices that produce such data, or at the network's edge. Additionally, it has confirmed the technique's significance for fog computing networks.

Keywords: Fog computing, Edge Computing and Scheduling in Fog Computing

1 Fog Computing

Fog computing is an architecture for computing, storing, and communicating that uses EDGE devices to carry out a sizeable chunk of computation, storing, and communicating locally before sending it over the Internet backbone.

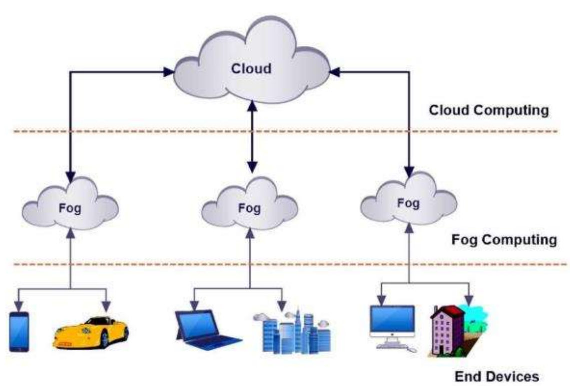

Fig-1: network

CISCO coined the phrase "fog computing," often known as "fogging." Fog computing seeks to bring the cloud closer to IOT devices. The main goal is to address the challenges that cloud computing has while processing IOT data and creating a bridge between the cloud and devices. This is the fundamental premise on which fog computing is conceptually based, and it aims to address the issues that cloud computing encounters while using IOT for data processing.

An example of a decentralized computing infrastructure or process is fog computing, in which computing resources are placed halfway between the data source and the cloud or another data center. A paradigm known as fog computing responds to user requests at edge networks. Typically, the hubs, routers, gateways, and other networking-related equipment at the fog layer carry out networking-related tasks. These gadgets are expected to be capable of concurrent processing and networking tasks, according to researchers. Despite the fact that these devices have less available resources than cloud servers, their geographical dispersion and decentralized nature enable them to provide dependable services with widespread coverage. The physical location of the devices in fog computing is significantly closer to the users than cloud servers are.

1.1 Fog Computing Architectural Design

In order to be widely used and adopted by the market, fog computing needs a standard architecture. There isn't a standard architecture readily available right now. However, Fog computing designs have been presented in numerous academic papers. In this part, the high-level architecture of fog computing is initially covered. We also list a few suggested Fog computing topologies. We conclude by presenting a thorough architecture for fog computing along with a thorough explanation of each component.

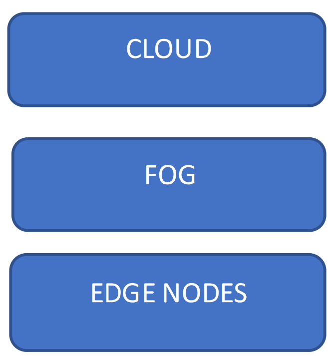

Fig-2: Three-layer architecture

1.2 HIGH-LEVEL ARCHITECTURE OF FOG COMPUTING

The Fog computing paradigm has three different layers in high-level architecture, with the Fog layer being the most important. All intermediate computing devices are housed in this layer. This plane, like the cloud, can use traditional virtualization technologies.

1.3 Applications of fog computing

1.3.1 Connected cars

There are currently self-driving or self-autonomous automobiles on the market, and they generate a lot of data. Based on the information provided, such as traffic, driving conditions, climate, etc., the data needs to be swiftly analyzed and processed. Fog computing helps with this. Other information, such as tracking and vehicle maintenance, is supplied straight to the manufacturer. With the aid of linked autos, edge and endpoint communication is made feasible.

1.3.2 Smart grids and smart cities

Real-time data is used by utility systems to efficiently run their operations. The remote data must be processed near to the location where it was generated. It's also possible that the data came from numerous sensors. Fog computing is made in a way that it can resolve both problems.

1.3.3 Real-time analytics

Fog computing deployments allow for the movement of data from the location where it is created to other locations. Real-time analytics using fog computing convey data from manufacturing systems to financial institutions using real-time data.

1.3.4 Fog Computing and the Healthcare Industry

To enhance its services and solutions, the healthcare sector frequently relies on innovative technologies. Similar to other technical developments, fog computing has been used to the advantage of the healthcare sector.

eHealth is one of the most important uses of fog computing in the medical field. The eHealth platform, available online and in print, gracefully leads healthcare stakeholders through the trajectory of healthcare, which frequently undergoes intriguing changes as a result of growing technological involvement and other structural changes.

They use a number of networks to link medical devices to cloud computing infrastructure. Information necessary for the treatment, payment, and recordkeeping processes is managed, transmitted, stored, and recorded by the application.

Fog computing makes it easier to diagnose and evaluate patients since experts may access electronic medical records (EMR) that include findings from tests like X-rays, ultrasounds, CT scans, and MRIs. Additionally, it secures the data in a private cloud.

1.4 Benefits of fog computing

Organizations now have more options for data processing because to the advancement of fog computing frameworks. For instance, some applications, such as those that require networked machines, must react fast to an occurrence. The devices and analytical endpoints establish low latency network connections. The data can be transmitted to analytical endpoints even with a slow network connection, which is not possible with data centers. An additional benefit is that a firewall can be used to defend a network.

1.4.1 Greater Business Agility

Programmers can easily construct fog programmes and deploy them whenever necessary by using the right tools. The system is driven by fog software to operate in accordance with consumer preferences.

1.4.2 Better Security

The same controls, procedures, and policies you employ in other areas of the IT environment may be used to protect fog nodes.

1.4.3 Deeper Insights with Privacy Control

Instead of transmitting the sensitive information to the cloud for analysis, it might be examined locally. The information gathering, analyzing, and storing equipment can be monitored and controlled by the IT personnel.

1.4.4 Reduced Operation Price

By processing specific data locally rather than transferring it to the cloud for additional analysis, fog computing can save bandwidth.

2 edge computing

Edge computing is a distributed information technology (IT) architecture wherein the client data is processed as near to the original source as is practical at the network's edge.

The lifeblood of contemporary business is data, which offers invaluable business insight and supports real-time control over crucial corporate operations. The quantity of data that can be routinely acquired from sensors and IoT devices working in real time from remote places and hostile operating environments is enormous, and it is available to organizations today practically anywhere in the world.

Instead of sending unprocessed data to a centralized data center, this technology processes and analyses data right where it is produced, whether that be on the floor of a factory, in a retail establishment, a large utility, or all throughout a smart city. The only information sent back to the primary data center for review and other human interactions is the outcome of that computing work at the edge, such as real-time business insights, equipment maintenance predictions, or other useful information.

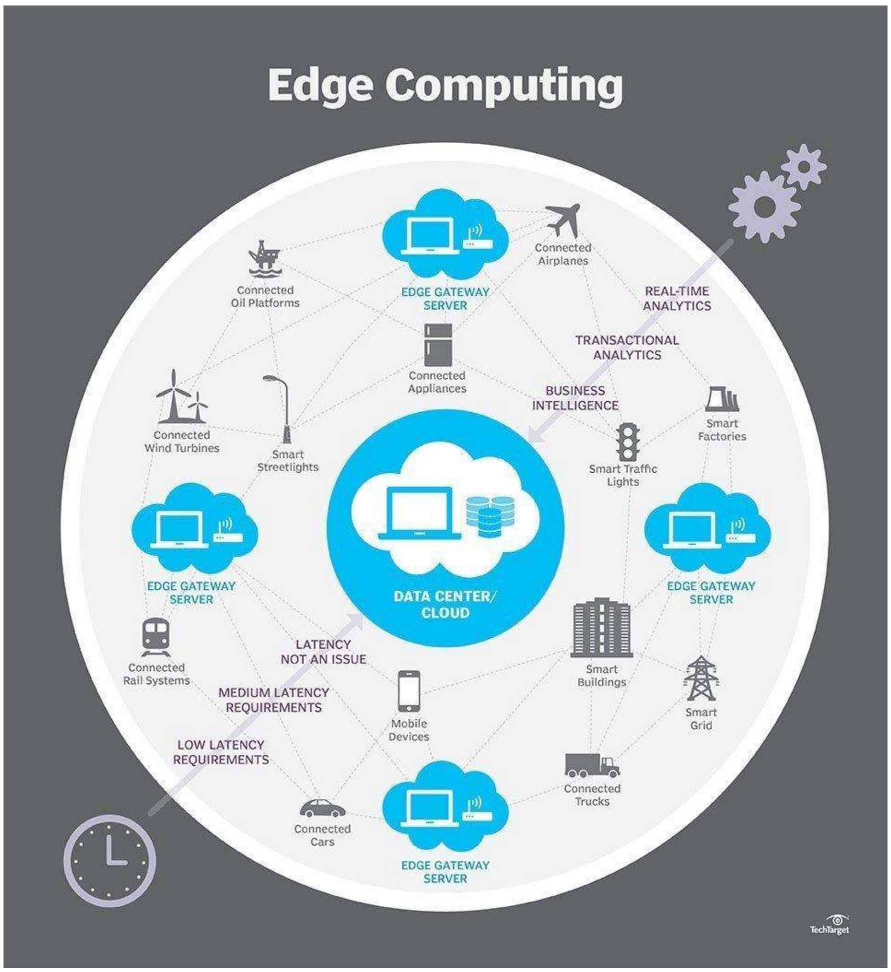

Fig-3: edge network

3 Cloud vs edge vs Fog Computing

The underlying problems of cloud computing, such as inconsistent latency, a lack of mobility support, and location awareness, remain unresolved despite the growing popularity of the technology.

While cloud computing focuses primarily on providing resources scattered throughout the core network, fog computing, also known as edge computing, can address those issues by offering elastic resources and services to end users at the network's edge.

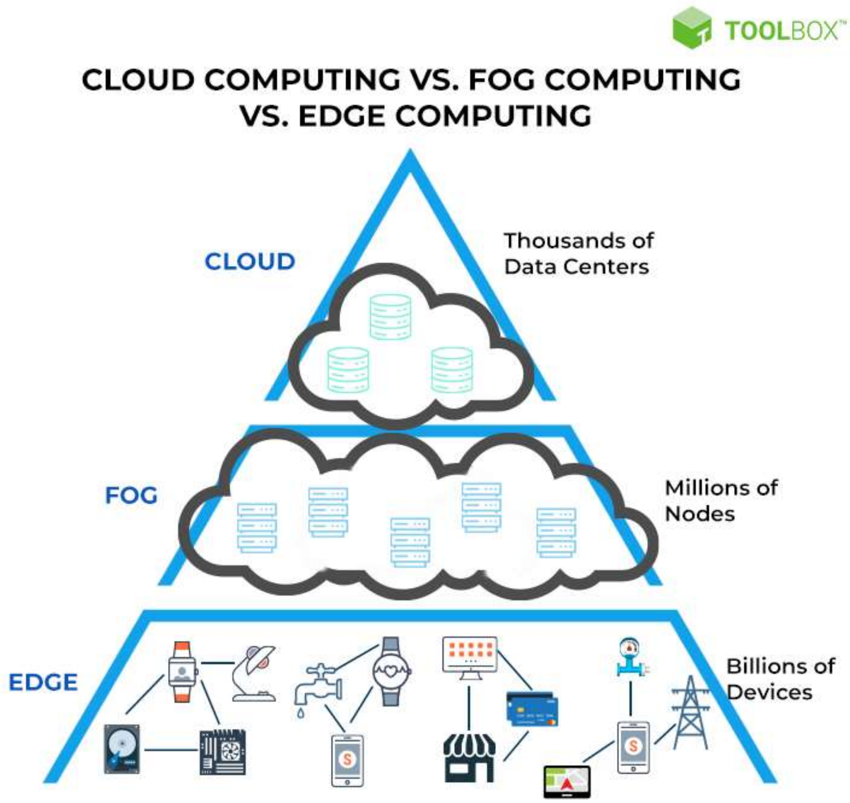

Fig-4: different layers and applications

4 Taxonomy

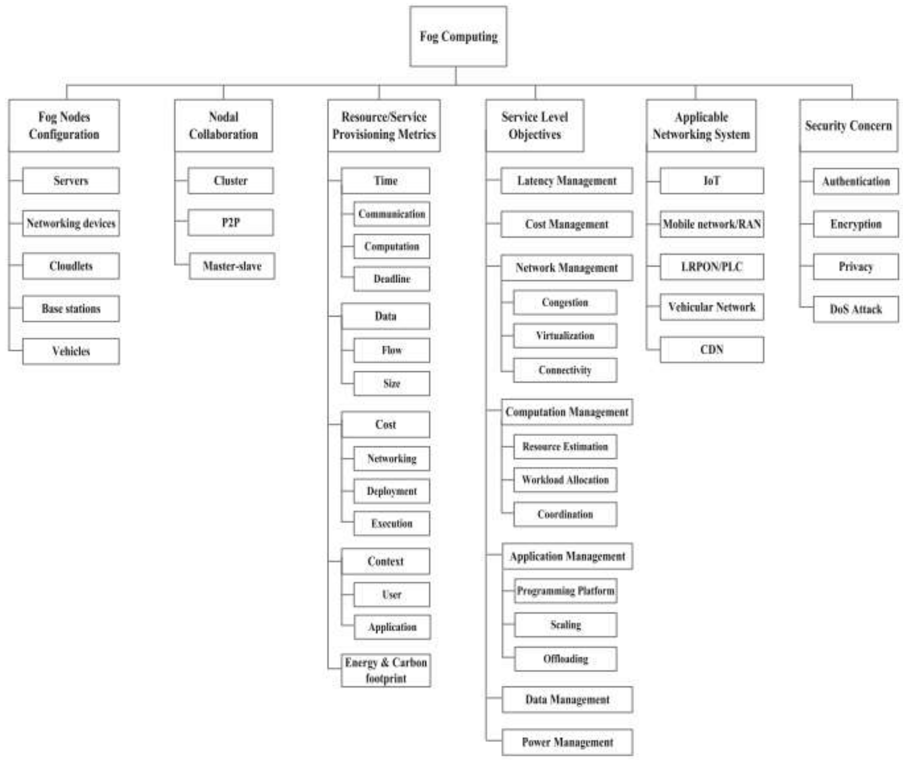

Fig-5: Taxonomy

4.1 Servers

The Fog servers are geo-distributed and installed in a lot of common locations, such as bus stops, malls, streets, parks, etc. These Fog servers are virtualized and furnished with networking, storage, and computation resources, just as lightweight Cloud servers. Fog servers have been regarded as the primary functional element of fog computing in numerous works. Other publications classify Fog servers based on their functionalities such as cache servers, compute servers, storage servers, etc. while some papers describe Fog servers based on their physical dimensions such as micro servers, micro datacenters, nano servers, etc. Fog computing is improved by using a server-based Fog node architecture to increase compute and storage capacity. It does, however, restrict how widespread the execution environment can be.

4.2 Networking Devices

In addition to their usual networking functions (routing, packet forwarding, analogue to digital signal conversions, etc.), devices like gateway routers, switches, set-top boxes, etc., could serve as potential Fog computing infrastructure. Various system resources, such as data processors, expandable primary and secondary memory, programming platforms, etc., are built into several existing networking devices. In other works, specialized networking devices such as Smart gateways and IoT Hub have been introduced as Fog nodes in addition to ordinary networking devices. Although the physical variety of the devices has a substantial impact on service and resource provisioning, distributed deployment of networking equipment aids in the widespread adoption of fog computing.

4.3 Cloudlets

The middle tier of the end device, cloudlet, and cloud hierarchy is where cloudlets are located and are classified as micro-clouds. In essence, cloudlets were created to provide Cloud-based services to users of mobile devices and can work in conjunction with MCC. Numerous publications refer to cloudlets as fog nodes. Cloudlet-based fog computing is highly virtualized and capable of managing several endpoints at once. Even after being deployed at the edge, cloudlets can occasionally function as centralized components because of structural limitations. This is because fog computing, which struggles to serve the Internet of Things, still has major centralized computation restrictions.

4.4 Base Stations

For flawless communication and data signal processing in mobile and wireless networks, base stations are crucial elements. Recent studies have found that conventional base stations with certain storage and processing capabilities are acceptable for fog computing. Small cell access points [30], Road Side Units (RSU), and other devices can be deployed as potential Fog nodes just like conventional base stations. For fog-based extensions of the Cloud Radio Access Network (CRAN), the Vehicular Adhoc Network (VANET), etc., base stations are preferred. But networking interference and expensive deployment costs make it difficult to create a dense Fog environment using base stations.

4.5 Vehicles

Fog nodes can be moving or parked cars at the edge of a network with computing resources. A distributed and highly scalable fog environment can be created using moving vehicles as fog nodes. However, in such a setting, maintaining the appropriate QoS coupled with the promise of privacy and fault tolerance will be quite difficult.

4.6 Nodal Collaboration

The literature specifies three methods for nodal collaboration in fog computing: cluster, peer to peer, and master-slave.

4.7 Cluster

By grouping together, fog nodes located at the edge can maintain a cooperative execution environment. The homogeneity of the Fog nodes or their location might also lead to cluster formation. While forming a cluster among the nodes, computational load balancing and the development of functional sub-systems can also be given higher attention. Collaboration on a cluster-based basis is efficient for utilizing multiple Fog nodes' capabilities at once. However, dynamic cluster formation heavily depends on the load at hand and the availability of Fog nodes, while static clusters are challenging to scale in runtime. In both situations, networking overhead is crucial.

4.8 Peer to Peer

Peer-to-peer (P2P) cooperation between nodes is highly popular in fog computing. P2P collaboration is possible in both flat and hierarchical structures. P2P collaboration between Fog nodes can be categorized as domestic, local, non-local, etc. in addition to base on proximity. Through peer-to-peer (P2P) collaboration, virtual computer instances are shared between the nodes in addition to the processed output of one node appearing as input to another node. Fog nodes can be easily enhanced for P2P cooperation, and nodes can be made reusable. But in P2P nodal collaboration, dependability and access control-related problems are common.

4.9 Master-Slave

Master-slave based nodal collaboration has been extensively discussed in a number of papers. A master Fog node often manages the functions, processing load, resource management, data flow, etc. of underlying slave nodes through master-slave based collaboration. Additionally, a hybrid collaborative network can be formed within the Fog computing environment using a master-slave paradigm, clusters, and P2P based nodal connections. Due to this type of functional decomposition, however, real-time data processing requires significant bandwidth for communication between the master and slave Fog nodes.

4.10 Resource/Service Provisioning Metrics

Numerous elements, including as time, energy, user application context, etc., have been discovered to play significant roles in resource and service provisioning in the extant literature on fog computing.

4.11 Time:

Time is regarded as one of the key aspects in the fog computing paradigm for effective resource and service provisioning.

4.11.1Computation time

refers to the amount of time needed to complete a task. An application's computation time is mostly determined by the resource configuration of the environment in which it is executing or by the task's schedule, which can be modified in response to the current workload. Additionally, task computation time aids in determining the active and inactive times of various applications, which has a big impact on resource and power management in Fog.

4.11.2 Communication time

In a fog computing environment, communication time essentially refers to the networking delay for exchanging data items. It has been covered in two ways in the literature: Fog nodes to Fog nodes, and end device/sensors to Fog nodes. When choosing the best Fog nodes for a task, the required communication time reflects the network context.

4.11.3 Deadline

The deadline indicates the longest service delivery delay that a system will tolerably accept. The achievement of deadlines has been regarded in certain articles [30, 32] as a crucial criterion for assessing the system's quality of service. Basically, how latency sensitive and latency tolerant an application is depending heavily on the service delivery deadline. For effective service and resource provisioning in fog computing, it is also possible to investigate the impact of other time-based metrics such as data sensing frequency of end devices/sensors, service access time in multi-tenant architecture, expected service response time, etc.

4.12 Data

The Fog computing literature frequently mentions input data size and data flow characteristics among other data-centric metrics.

4.12.1 Data size

The magnitude of the data indicates how much information needs to be processed using fog computing. Data size has been examined in relation to the requests' computational space needs in a number of works. Additionally, bulk data gathered from dispersed devices or sensors may also have elements of big data. Provisioning resources and services in this situation in accordance with the data load may be a successful strategy. Additionally, data size influences whether associated computational activities should be processed locally or remotely.

4.12.2 Data flow:

The properties of data transmission are defined by data flow. Event-driven or real-time data flow can greatly affect the delivery of resources and related services throughout the Fog computing ecosystem. Additionally, abrupt changes in data flow can occasionally encourage dynamic load balancing across the nodes. Additionally, for resource and service delivery in fog computing, the efficiency of heterogeneous data architecture, data semantic norms, and data integrity needs can be explored.

4.13 Cost:

When it comes to providing fog resources and services, there are some circumstances where cost-related factors from the perspectives of service providers and users have a significant impact.

4.13.1 Networking cost:

The amount spent on bandwidth and related costs directly affect networking costs in a fog computing environment. In some works, the costs associated with data uploading from end devices or sensors and inter-nodal data sharing have been regarded as components of networking costs, while in other works, the costs associated with network latency caused by bandwidth problems have been regarded as components of networking costs.

4.13.2 Deployment cost:

In the context of fog computing environments, deployment costs are mostly related to costs associated with placing infrastructure. Cost-effective infrastructure deployment has been viewed in several articles as facilitating effective resource and service provisioning. Costs associated with infrastructure deployment can be compared between installing Fog nodes in a network and setting up virtual computing instances on those nodes.

4.13.3 Execution cost:

The computational costs incurred by Fog nodes when running applications or performing tasks are referred as execution costs. Although resource provisioning and invoicing for execution cost are common practices in other computing paradigms, very few works have employed this parameter in fog computing. Total execution costs have been computed in these works based on task completion times and resource use costs per unit of time. For resource and service provisioning in fog computing, costs associated with security precautions, the highest price a user is prepared to pay for a service, and migration costs can also be taken into account.

4.14 Context:

Context describes the state or status of a particular entity in several contexts. User and application-level context for resource and service provisioning has been studied in works of fog-based research.

4.14.1 User:

When allocating resources for that user in the future, user context factors including user characteristics, service usage history, service relinquish probability, etc. might be employed. Users' service input, such as Net Promoter Score (NPS), and user requirements, may also be utilized to plan the provision of services and resources. Users' density, mobility, and network status have also been taken into account for service providing in other works.

4.14.2 Application context:

The operating requirements of various applications can be thought of as application context. Operational needs can influence how resources and services are provisioned. These requirements include those for task processing (CPU speed, storage, memory), networking, etc. Current task loads for various applications have also been taken into account as application context in other works.

Additionally, contextual information in fog computing can be explored in terms of the nodal characteristics, application architecture, execution environment, etc. These contexts, along with others, can be extremely important for provisioning resources and services. Therefore, it is crucial to thoroughly research the effects of each contextual aspect.

4.15 Energy Consumption and Carbon Footprint:

Energy-related concerns have received more emphasis in a number of works when offering Fog resources and services. These works have extensively emphasized the energy consumption of all devices in a home-based Fog computing environment as well as the tradeoff between energy and latency in various stages of Fog-Cloud interaction. In another study, the carbon emission rate of various nodes in relation to per-unit energy consumption was taken into account for resource provisioning. Energy features of end components, such as residual battery lifetime and energy-characteristics of communication channel, can also be investigated in the provisioning of Fog resources because end devices/sensors are energy-constrained.

4.16 Service Level Objectives:

Numerous original Fog node architectures, application programming platforms, mathematical models, and optimization techniques have been suggested in the literature to achieve particular SLOs. The majority of the SLOs achieved are management-oriented and deal with concerns like latency, power, cost, resource, data, application, etc.

4.17 Latency Management:

In essence, latency control in fog computing prevents the total service delivery time from going above a predetermined limit. A service request's maximum tolerated latency or an application's QoS requirement can both fall within this threshold. Effective nodal collaboration initiation has been highlighted in some works to ensure proper latency management and allow computation tasks to be carried out by collaborated nodes within the set latency constraints. In another study, the distribution of computation tasks between the client and Fog nodes was done to reduce the overall computation and communication latency of service requests.

Additionally, a low-latency Fog network design has been suggested for latency management in another piece of study. The main goal of this effort is to choose the Fog network node that delivers services with the lowest latency.

4.18 Cost Management:

Capital Expenses (CAPEX) and Operating Expenses are two terminologies that can be used to talk about cost management in fog computing (OPEX).

The deployment of distributed fog nodes and the networks that connect them is the major source of CAPEX in fog computing. In this instance, a key factor in reducing the CAPEX in fog computing is the optimal placement and number of fog nodes. In order to address this problem, a network design for fog computing has been presented. This architecture maximizes the locations and numbers of fog node deployment while minimizing the total CAPEX in fog computing.

Fog nodes have been discussed as virtualized platforms for starting VMs in another piece of work. The pricing varies from provider to provider for the execution of data processing tasks in these VMs. As a result, it is possible to use the cost diversity of various Fog nodes and providers to reduce OPEX in fog computing. In light of this, a method to select an appropriate group of Fog nodes for hosting VMs has been put out in that work with the intention of reducing the OPEX in fog computing.

4.19 Network Management:

Core-network congestion control, support for Software Define Network (SDN)/Network Function Virtualization (NFV), assurance of seamless connectivity, etc. are all examples of network management in fog computing.

4.19.1 Network congestion:

The fundamental cause of network congestion is rising network overhead. The simultaneous interactions of end components with cloud datacenters can significantly raise the burden on the core network since, similar to IoT, end devices/sensors are widely dispersed over the edge. In this situation, network congestion will happen and the system's performance would suffer. In order to allow local processing of the service requests, a layered design of the fog node has been developed. Due to this, Clouds receive condensed versions of service requests, which cause less network congestion even when they receive large numbers of requests.

4.19.2 Virtualization:

Significant academic effort has already been given to the virtualization of traditional networking systems. One of the main drivers of a virtualized network is SDN. SDN is a networking method that implements in software on different servers and decouples the control plane from networking hardware. The ability to support NFV is one of the key features of SDN. Basically, NFV is an architectural idea that allows software to do conventional networking tasks like network address translation, firewalling, intrusion detection, domain name service (DNS), and caching by virtualizing those tasks. SDN and NFV have a significant impact in cloud-based environments because of the variety of services they offer. As a result, various research papers have proposed new network architectures for fog computing to support SDN and NFV.

4.19.3 Connectivity:

Despite their physical diversity, connectivity provides flawless communication between end devices and other entities such as the cloud, fog, desktop computers, mobile devices, end devices, etc. As a result, network resource discovery, communication maintenance, and compute capacity are made simpler. This problem has previously been addressed in a number of publications on fog computing, and novel architectures for fog nodes, such as the IoT Hub [24] and fog networking, such as vehicular fog computing, have been developed. Additionally, a policy-driven architecture for fog computing has been established enabling protected connectivity among the devices.

4.20 Computation Management:

The achievement of proper computational resource management in fog computing is one of the most important SLOs. Resource estimation, workload distribution, resource coordination, etc. are all examples of fog computing resource management.

4.20.1 Resource estimation:

Resource estimate in fog computing aids in the allocation of computational resources in accordance with some criteria, allowing for the achievement of desired QoS and the imposition of accurate service prices.

Resource estimation strategies are produced in the literature currently available in terms of user characteristics, experienced QoE, aspects of service accessing devices, etc.

4.20.2 Workload allocation:

In fog computing, workload distribution should be done in a way that maximizes resource use and reduces extended periods of computational inactivity. More specifically, a balanced load across several components is guaranteed. A scheduling-based workload allocation policy has been implemented in a Fog-based research project to balance the computing load on Fog nodes and client devices. As a result, overhead on both sides is reduced and QoE is improved. Another piece of work has developed a workload allocation approach that balances delay and power consumption in the interaction between fog and clouds.

4.20.3 Coordination:

Due to their heterogeneity and resource limitations, coordination between various Fog resources is absolutely crucial. Because fog computing is decentralized, large-scale applications are typically distributed over different fog nodes. Without adequate coordination of the Fog resources in such situations, it will be difficult to achieve the desired performance. Given this information, in a focused

For Fog resource management, a graph-based resource coordination paradigm has been suggested.

4.21 Application Management:

Effective programming platforms are crucial for ensuring good application management in fog computing. Scalability and compute offloading capabilities are not the only important factors in application management.

4.21.1 Programming platform:

A programming platform offers the tools required to create, construct, and run applications, such as interfaces, libraries, run-time environments, etc. Fog computing is dynamic, making it difficult to provide effective programming support for complex applications. A new programming platform called Mobile Fog has been proposed to address this problem. For creating large-scale applications over heterogeneously distributed devices, Mobile Fog provides a more straightforward abstraction of programming models. In another study, a programming environment based on distributed data flow technique has also been created for application creation in fog computing, in addition to managing resources during the execution of programmes.

4.21.2 Scaling:

Scaling refers to an application's ability to adjust and maintain service quality despite an increase in application users and unanticipated events. Application scheduling and user service access are two other areas where scaling approaches can be used. Fog computing has recently proposed an architecture for a QoS aware self-adaptive scheduler to support scalable scheduling of data stream applications. This scheduler does not require global environmental information and can scale applications as users and resources increase. Besides, based on distance, location and QoS requirements of the service accessing entities, an adaptive technique for user's service access mode selection has also been proposed\sin Fog computing.

4.21.3 Offloading:

Offloading strategies make it possible for end devices with limited resources to send their computational jobs to certain devices with abundant resources for execution. In a mobile cloud context, computational offloading is quite prevalent. But recently, computation offloading support for mobile applications in fog computing has been highlighted in multiple articles as a part of improving Fog computing's compatibility with other networking systems. Offloading approaches have been discussed in these publications in terms of both the availability of resources and distributed computing for mobile applications.

4.22 Data Management:

Another significant SLO that must be met for effective performance is data management. Data management in fog computing has been studied from a variety of angles in several research studies. For data management policy in fog computing, proper data analytic services and resource allocation for data pre-processing have been prioritized. Additionally, low latency aggregation of data from dispersed endpoints or sensors can be taken into account for effective data management. Additionally, the storage capacity of end devices and sensors is not very high. In this situation, storage expansion in fog computing for maintaining end entity data can have a significant impact. Storage expansion in fog computing for mobile devices has therefore been studied along with application management as a crucial component of data management.

4.23 Power Management:

Power management as a service for various networking systems can be effectively provided by fog computing. A service platform for fog computing that can enable power management in a home-based IoT network with customizable consumer control has been put out in [34]. In some circumstances, Fog computing can also control how much energy is used by centralized Cloud datacenters. The kind of apps that are running heavily influences how much power cloud datacenters need. In this scenario, fog computing can support cloud datacenters by offering the necessary infrastructure to run a number of energy-intensive applications. So as to ensure effective power management for Cloud datacenters, energy consumption in those facilities will be reduced. Additionally, it is possible to limit carbon footprint emissions by regulating electricity in fog computing.

4.24 Applicable Network System:

A key component of IoT is fog computing. The use of fog computing in different networking systems, such as the mobile network, content distribution network, radio access network, vehicular network, etc., has also been highlighted in recent research works.

4.25 Mobile Network/Radio Access Network:

The use of fog computing has been investigated in numerous research studies for the mobile network, another networking system. The investigation of Fog computing's compatibility with 5G mobile networking has, in essence, received a lot of attention in these works. In comparison to current cellular systems, 5G allows for faster communication, greater signal capacity, and significantly less latency in the delivery of services. Other mobile networks, such as 3G, 4G, etc., can use fog computing in addition to 5G. A different study has also looked closely at the power-delay tradeoff driven workload allocation in the fog cloud for mobile based communication. A network's radio connections are used by the Radio Access Network (RAN) to make it easier for individual devices to communicate with other network entities. There has already been a lot of research interest in cloud assisted RAN, or CRAN.

2.26 Long-Reach Passive Optical Network/Power Line Communication:

The Long-Reach Passive Optical Network (LRPON) has been developed to support wireless, industrial, and residential backhaul services that are latency-sensitive and bandwidth-intensive. In addition to spanning a broad region, LRPONs make the process of network consolidation simpler. Fog computing and LRPONs have been combined in [40] for the purpose of optimizing network design. A popular communication technique in the Smart Grid is power-line communication (PLC). In a PLC, data and alternating current (AC) are conveyed concurrently through electrical wiring.

4.27 Content Distribution Network:

High performance and availability are guaranteed by the Content Distribution Network (CDN), which is made up of dispersed proxy servers that provide content to end users. Fog nodes are viewed as content servers in various research studies based on Fog [19, 42] to support content distribution using Fog computing. End users can get fog-based content services with only a very slight latency since fog nodes are scattered over the network's edge. As a result, ensuring high performance in content delivery will be simpler.

4.29 Vehicular Network:

The autonomous construction of a wireless network for data exchange and resource augmentation among cars is made possible by the vehicular network. Vehicles are equipped with networking and computing capabilities in this networking system. Vehicles located at the edge of a network are sometimes referred to as fog nodes in research studies to support vehicular networks powered by fog computing.

4.30 Security Concern:

Since the underlying network between end devices/sensors and cloud datacenters is where fog computing is located, its security vulnerability is very high. However, security issues in fog computing have been discussed in terms of user authentication, privacy, secured data exchange, DoS attack, etc. in the literature that has already been written.

4.31 Authentication:

In systems based on fog, user authentication is crucial for thwarting intrusion. Unwanted access to the services is intolerable in any way because Fog services are utilized on a "pay as you go" basis. In addition to user authentication, the secured Fog computing environment has also seen device authentication, data migration authentication, and instance authentication.

4.32 Privacy:

Data processed by fog computing comes from end devices and sensors. These data are occasionally discovered to be closely related to users' circumstances and interests. Proper privacy guarantee is therefore seen as one of the key security issues in fog computing. The need for more research into the privacy problems in fog-based vehicular computing has been highlighted.

4.33 Encryption:

In essence, fog computing is an addition to cloud computing. Via some circumstances, data that has been processed in fog computing must be sent to the cloud.

It is imperative to encrypt these data in Fog nodes since they frequently include critical information. A data encryption layer has been added to the suggested Fog node architecture to cater for this circumstance.

4.34 DoS Attack:

Fog nodes struggle to handle a high volume of concurrent queries because of their resource limitations. In this situation, the effectiveness of fog computing can be significantly diminished. DoS attacks can be extremely important in causing such severe service interruptions in fog computing. Fog nodes can be kept busy for a longer duration by simultaneously making a large number of pointless service requests. Resources for hosting practical services become unavailable as a result. This type of DoS attack in fog computing has been thoroughly discussed in.

4.35 Gap Analysis and Future Directions:

Fog computing extends Cloud-based facilities and is located closer to end users. Fog computing plays very important functions in providing services to widely dispersed end devices and sensors. Therefore, from both an academic and commercial perspective, fog computing has recently emerged as one of the main study areas. Table 1 is a quick summary of a few studies that have been reviewed and are mentioned in the literature on fog computing. Although the literature on fog computing has identified many crucial elements, there are still some problems that must be resolved if this field is to advance further. The gaps in the body of literature as well as a number of potential future research directions have been discussed in this section.

4.36 Context-Aware Resource/Service Provisioning:

In fog computing, context awareness can result in effective resource and service delivery. Contextual data in fog computing can come in a variety of formats, such as;

- Environmental context: Time (Peak, Off-peak), Location.
- Application context: Application architecture, Latency sensitivity, etc.
- User context: Activity, Mobility, Social interactions, etc.

• Device context: Remaining battery life time, Available resources, etc.

• Network context: Network traffic, Bandwidth, etc.

Although contextual information has been taken into account in several works of fog-based research, many crucial aspects of contextual information remain unexplored. A potential area for Fog-based research is the examination of various approaches to use contextual information in resource and service management.

4.37 Sustainable and Reliable Fog Computing:

Fog computing's economic and environmental impact is greatly enhanced by sustainability. However, there are other difficulties with the overall sustainable architecture of fog computing, including QoS assurance, service reusability, and energy efficiency.

5 Scheduling Model in Fog Computing:

SCHEDULING OBJECTIVES

The goal of scheduling is to find the best solution for scheduling a set of tasks or workflows on a set of machines. Scheduling parameters play an important role in the scheduling problem's success. Based on the service approach, scheduling parameters are divided into two groups: consumer services and service providers.

5.1 SCHEDULING PROBLEMS:

The network's edge nodes and core cloud nodes are here connected or extended by fog. In fog computing, base stations (BS) and relay nodes (RNs) are typically used. When fog is unable to reply to the relay node, very seldom do the edge nodes directly communicate with the cloud network. In this scenario, the relay nodes are referred to as edge nodes.

So, we need to schedule the entire network, and the allocated schedule should be the best one possible. We can schedule the network by following some steps.

Task, resource, job, and workflow scheduling are five of the most important scheduling issues in fog computing that we take into consideration. Several works on fog computing scheduling are now in existence. After reading carefully through the most recent research papers on scheduling, we choose the papers that are most pertinent, compare them, and group them into the following categories based on the nature of the scheduling issue.

Resource Scheduling:

Finding the best-suited resources for the customer is the goal of resource scheduling, which also aims to improve resource utilization, cut down on processing delays, and improve quality of service (QoS).

Here, we want to make the best use of the available resources and speed up the processing to improve network performance. Therefore, we must evaluate the available resources and allocate them appropriately.

Task Scheduling:

The goal of task scheduling is to distribute a collection of jobs among fog nodes in order to satisfy QoS criteria while minimizing task execution and transmission times. In this case, the task will be delegated to the base stations in the fog layer so that execution time will be reduced and the task will execute quickly. However, in some cases, the base stations in the fog may be overloaded or unable to schedule the requested task, in which case the edge node will directly communicate with the cloud and the task will execute, potentially taking longer than expected.

Resource Allocation:

In order to satisfy QoS requirements while minimizing task execution and transmission times, task scheduling assigns a set of tasks to fog nodes. In this case, the task will be delegated to the base stations in the fog layer so that execution time will be reduced, but in some cases, the base stations in the fog may be overloaded or unable to schedule the requested task, in which case the edge node will directly communicate with the cloud and it will execute, which may result in a longer execution time, but the task will still be completed.

The dynamic and complicated nature of the Cloud-Fog environmental paradigm in computer systems makes resource allocation difficult for the providers. To accomplish the goals of the overall FC environment, resource allocation for addressing the particular demands of the FC setting will be planned. By properly utilizing the resource capabilities and taking care of the system requirements, the resource allocation process may be made to conform with network delay and low latency management.

Workflow Scheduling

Scheduling workflows seeks to distribute computer resources with various processing powers to the jobs of workflow applications, which can reduce the make-span and cost.

Job Scheduling

The goal of job scheduling is to distribute a group of tasks among the fewest possible resources. the quickest CPU execution time with minimal space

REFERENCES:

- Bonomi, Flavio, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. "Fog computing and its role in the internet of things." In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pp. 13-16. 2012
- Naha, Ranesh Kumar, Saurabh Garg, Dimitrios Georgakopoulos, Prem Prakash Jayaraman, Longxiang Gao, Yong Xiang, and Rajiv Ranjan. "Fog computing: Survey of trends, architectures, requirements, and research directions." *IEEE access* 6 (2018): 47980-48009.
- Matrouk, Khaled, and Kholoud Alatoun. "Scheduling Algorithms in Fog Computing: A Survey." *Int. J. Networked Distributed Comput.* 9, no. 1 (2021): 59-74.
- Mahmood, Zaigham, and Muthu Ramachandran. "Fog computing: Concepts, principles and related paradigms." In *Fog Computing*, pp. 3-21. Springer, Cham, 2018.
- Waqas, Muhammad, Yong Niu, Manzoor Ahmed, Yong Li, Depeng Jin, and Zhu Han. "Mobility-aware fog computing in dynamic environments: Understandings and implementation." *IEEE Access* 7 (2018): 38867-38879.
- Waqas, M., Niu, Y., Ahmed, M., Li, Y., Jin, D. and Han, Z., 2018. Mobility-aware fog computing in dynamic environments: Understandings and implementation. *IEEE Access*, *7*, pp.38867-38879.
- Mahmud, Redowan, Ramamohan Arao Kotagiri, and Rajkumar Buyya. "Fog computing: A taxonomy, survey and future directions." In *Internet of everything*, pp. 103-130. Springer, Singapore, 2018.
- Mahmud, R., Kotagiri, R. and Buyya, R., 2018. Fog computing: A taxonomy, survey and future directions. In *Internet of everything* (pp. 103-130). Springer, Singapore.
- https://www.researchgate.net/publication/329229804_Mobility-Aware_Fog_Computing_in_Dynamic_Environments_Understandings_and_Implementation?enrichId=rgreq-caad4b7acfc13de0b6f962f689fe1377-XXX&enrichSource=Y292ZXJQYWdlOzMyOTIyOTgwNDtBUzo3MDM3MDk0MDczNzEyNjRAMTU0NDc4ODgwNTgzNw%3D%3D&el=1_x_3&_esc=publicationCoverPdf
- Bittencourt, Luiz F., Javier Diaz-Montes, Rajkumar Buyya, Omer F. Rana, and Manish Parashar. "Mobility-aware application scheduling in fog computing." *IEEE Cloud Computing* 4, no. 2 (2017): 26-35.