REPUBLIC OF TURKEY

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

FOG AND CLOUD LOAD BALANCING USING REGRESSION BASED RECURRENT DEEP LEARNING ALGORITHM

Eftekhar Jumaah JAMEEL

Master's Thesis

Supervisor

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Istanbul, 2022

FOG AND CLOUD LOAD BALANCING USING REGRESSION BASED RECURRENT DEEP LEARNING ALGORITHM

Eftekhar Jumaah Jameel JAMEEL

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled “FOG AND CLOUD LOAD BALANCING USING REGRESSION BASED RECURRENT DEEP LEARNING ALGORITHM” prepared by EFTEKHAR JUMAAH JAMEEL and submitted on 12/5/2022 has been **accepted unanimously** for the degree of Master of science in Electrical and Computer Engineering.

____

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Supervisor

Thesis Defense Jury Members:

Asst. Prof. Dr. Abdullahi Abdu IBRAHIM	Faculty of Engineering and Architecture, Altinbas University	____________________
Asst. Prof. Dr. Hasan ABDULKADER	Faculty of Engineering and Architecture, Altinbas University	____________________
Asst. Prof. Sewale TAHA	Faculty of Computer Technologies, Beykent University	____________________

I hereby declare that this thesis meets all format and submission requirements of a Master’s thesis.

Submission date of the thesis to Institute of Graduate Studies: ___/___/___

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Eftekhar Jumaah JAMEEL

Signature

DEDICATION

To My wonderful parents, who never stop in giving themselves in countless ways… (my late father and my late mother). To the sources of hope, who stand by me in everything and lead me through the valley of darkness with light of hope and support (My brothers, sisters, and all my family members).To all the people in my life who touch my heart.

ACKNOWLEDGEMENTS

I indebted to ALMIGHTY ALLAH, the propitious, the benevolent and sovereign Whose blessing and glory flourished my thoughts and thrived my ambitions, giving me talented teachers, affectionate parents, sweet brothers, and sisters. Trembling lips and wet eyes, praise for HOLY PROPHET MUHAMMAD (P.B.U.H.) for enlightening our conscience with the essence of faith in ALLAH, converging all His kindness and mercy upon him.

The work presented in this manuscript was accomplished under the sympathetic attitude, politely behavior, animate directions, observant pursuit, scholarly criticism, cheering perspective and enlightened of my supervisor Assist. Prof. (Dr. Abdullahi Abdu Ibrahim) His thorough analysis and rigorous critique improved not only the quality of this dissertation, but also my overall understanding of Agricultural Extension. I am grateful to his ever-inspiring guidance, keen interest, scholarly comments and constructive suggestions throughout the course of my studies.

I would like to express my sincere gratitude to the head of the Electrical and Computer Engineering department (Prof. Dr. Mesut ÇEVİK) for giving me the opportunities to complete my studies and to overcome the difficulties that I faced during the study. I am also grateful for the insightful comments offered by the members of my Dissertation Committee. The generosity and expertise of the committee have improved this study in innumerable ways and saved me from many errors, those that inevitably remain are entirely my own responsibility. There are no proper words could ever adequately express my obligations to my affectionate and adoring my late father, mother, brothers, and sisters and all my family members whose hands always raised in prayers for me and without whose moral and financial support, the present distinction would have merely been a dream. They always acted as a lighthouse for me in the dark oceans of life path. My profound love thanks and indebtedness are due to my colleagues and friends for their moral support and motivation, which drives me to give my best. Finally, May ALLAH ALMIGHTY blesses all these people with long, happy and peaceful lives (Ameen).

ABSTRACT

FOG AND CLOUD LOAD BALANCING USING REGRESSION BASED RECCURENT DEEP LEARNING ALGORITHM

Jameel, Eftekhar Jumaah

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Asst. Prof. Dr. Abdullahi Abdu IBRAHIM

Date: May/2022

Pages: 62

Fog computing has been studied by a variety of academics, and they have identified concerns which need to be solved. It is the primary goal of this project to build and execute an energy conscious load balancer that will reduce energy usage and help in the distribution of loads. Furthermore, with chips becoming more compact and containing increasingly dense circuitry, the release of energy in the form of heat increases. This means even more energy consumption, as computational components do not operate well at very high temperatures and therefore are cooling systems. In this work we investigate the relative strengths and weaknesses of the different fog and cloud models and structures

Keywords: Fog, Cloud, Machine Learning, Artificial Intelligence, Genetic Algorithm.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

VANET	:	Vehicular Adhoc Network
IoT	:	Internet of Things
ANN	:	Artificial Neural Network
CNN	:	Convolutional Neural Network
VM	:	Virtual Machine
QoS	:	Quality of Service
SLA	:	Service Level Agreement

1. INTRODUCTION

1.1 BACKGROUND

Machine learning encompasses the technology in which a computer analyzes data to “learn” empirically without human involvement. Deep learning is a subset of machine learning. In machine learning, the computer receives data with an established set of resources to analyze, whereas in deep learning the computer takes unstructured data such as text, audio or video and determines on its own which resources are relevant to the analysis. In simple terms, the computer receives input pairs and samples and the corresponding outputs and is able to do the reverse process to find out what operations are needed to transform the input into the output. Modern deep learning models require massive amounts of compute and storage, preventing most organizations from building these systems themselves. So, as we found out during this research, companies overwhelmingly look to the cloud for deep learning projects. This approach allows companies to purchase the amounts of data storage and computing power they need for projects without the need to purchase, configure and maintain infrastructure in-house, producing significant savings over time. With a diverse and rapidly growing user base due to the high profile and potential of AI, the technology landscape changes rapidly.

New tools and methodologies are constantly available. Therefore, allowing users to develop on the platform with maximum flexibility is essential for long-term success in the cloud-based machine learning market. In short, cloud platforms need to support thousands of development tools and frameworks used today and tomorrow, with mandatory security and willingness to adhere to data processing and privacy standards. This is the third consecutive year this study has been conducted, and what we've seen in this period are transformative changes in model capabilities, computing power, and developer tools that enable new and exciting results. In 2018, we saw strong progress towards this end with 14% of projects being classified as in production, processing live data. This year brought another leap toward that effect with organizations ranging from 20-person startups to global Fortune 100 companies implementing deep learning in production, with 38% of projects in production. [2].

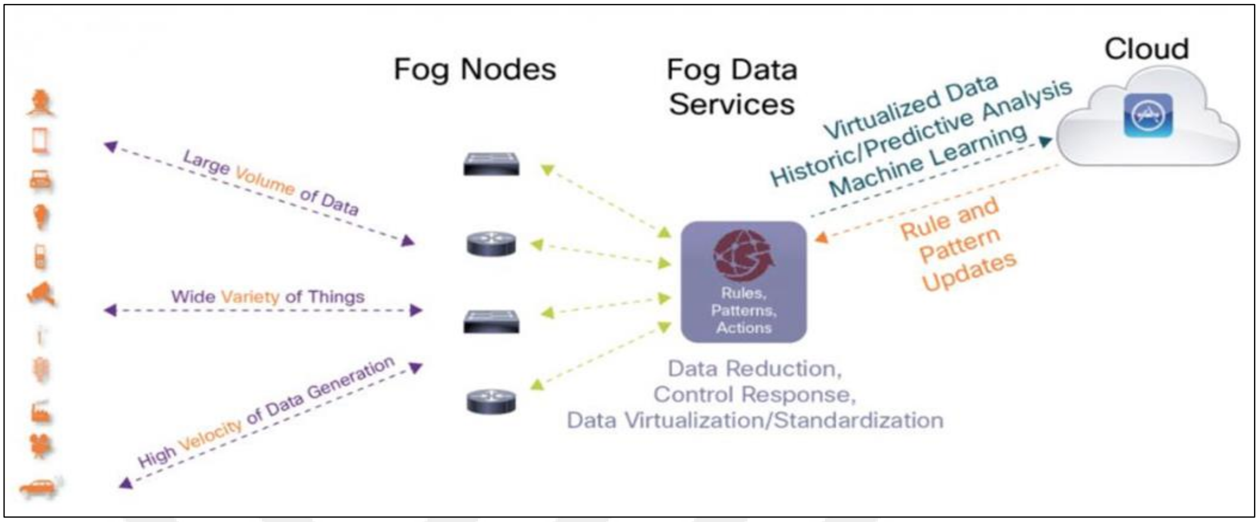

Figure 1.1: ML Data Processing Role in Fog Computing [1]

1.2 PLANNING THE PROBLEM

The problem of saving energy and controlling the temperature of computers has been around since the appearance of the first computer. With the increase in processing capacity, computer components consume more energy. Furthermore, with chips becoming more compact and containing increasingly dense circuitry, the release of energy in the form of heat increases. This means even more energy consumption, as computational components do not operate well at very high temperatures and therefore are cooling systems. According to [3] the search for technologies that could achieve energy savings had a short pause in the 1960s, when there was a transition from valves, used in the first electronic computer to transistor technology in the 1960s , but soon the density of the components increased, making the need for cooling techniques return. In the 1990s, the switch from transistor bipolar technology to complementary metal oxide semiconductor (CMOS) technology caused another pause, due to the large reduction in energy dissipation per component but it is expected that by 2025 the energy consumption of data centers such as clouds and fog sites will surpass any other technology [5] as shown in the figure below”

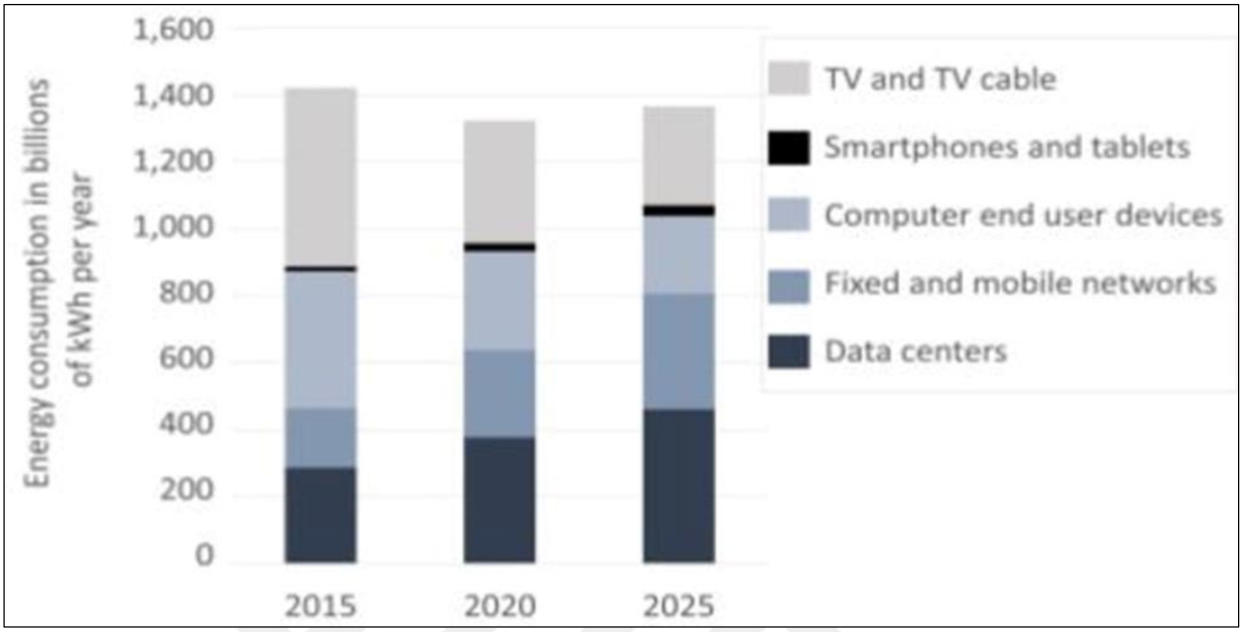

Figure 1.2: Energy Consumption in Data Centers by 2025 [5]

1.3 THESIS OBJECTIVES

The main objectives of this thesis include:

A. The goals and motivations behind the deep learning in fog and cloud optimization.

B. The implementation strategy and associated benefits of using deep learning.

C. The structures, methods, and other development tools in use.

D. The relative strengths and weaknesses of the different fog and cloud models and structures.

1.4 THESIS CONTRIBUTION

Fog computing has been studied by a variety of academics, and they have identified concerns which need to be solved. It is the primary goal of this project to build and execute an energy conscious load balancer that will reduce energy usage and help in the distribution of loads. In order to test the method that was created, establishing a fog computing platform by assigning a job and assessing the outcomes, the energy-aware load balancing method was found to be effective when it was tested.

1.5 THESIS ORGANIZATION

The thesis is structured as follows: Section 2 is where we review some of the previous work and implementation on smart grids. Section 3 is where we give a background about all the components. Section 4 is where we explain our model in details and implementation of that model in details, Section 5 is where we simulate our model and record the results obtained. Section 6 is where we conclude our work and put a future scope into perspective.

2. LITERATURE REVIEW

The following is a list of different works that are relevant to our work and how they explain the principles of fog computing.:

Pearson, Siani, et al. (2010) [4] I composed a piece of academic writing for my professor. Cloud computing is a new paradigm that is becoming more popular for large-scale infrastructures. On-demand provisioning and a pay-per-use business model work together to save expenses by pooling computer and storage resources across many users. Furthermore, these new features have a direct impact not just on IT investment, but also on traditional security measures such as trust and privacy protection. Many of these systems will have to be rethought because of the new paradigm that has emerged. In this essay, the author discusses the security and privacy issues associated with cloud computing. They also spoke about how these problems may be resolved".

Dlamini, M. T. et al. (2011) [5] In the words of the authors of this study, "Cloud computing is a new paradigm for the production, distribution, and consumption of information technology resources and services through the Internet." There are several benefits to implementing this new computing paradigm, including cost savings, increased service availability, enhanced IT operations efficiency and flexibility, and improved IT operations efficiency and flexibility. According to the findings of the research, security is a big concern for many of these companies. The security risks around "cloud computing" are being misconstrued as a result of this, and it is past time to clear things up. Further investigation is required in order to distinguish between reality and fiction. Because of this, the purpose of this research is to identify true security concerns from cloud computing's hype, fear of the unknown, and ignorance of the technology. This research attempts to advance the current discussions on cloud computing security in order to clear away the "foggy cloud" that has surrounded the issue. Decision-makers should be made aware of the security problems associated with cloud computing so that they may make educated judgments.".

Stolfo, Salvatore J., et al. (2012) [6] In their research, they said that "cloud computing has the potential to fundamentally revolutionize the way we use computers, as well as how we access and maintain our personal and business information." As a consequence of these new computer and communication paradigms, there are new data security concerns to consider. Data theft efforts, especially those perpetrated by employees of cloud service providers, have been demonstrated to be unsuccessful when encrypted data is used to protect the data. They come up with a novel approach to cloud data protection by using offensive decoy technology. Because of this monitoring, they were able to discover any unusual patterns of data access. Whenever an intruder is suspected of unauthorized access, we launch a disinformation campaign against him by sending him enormous amounts of decoy material back to him. This prevents the authentic information of the user from being abused. Through trials carried out in a local file environment, it has been shown that this method may provide unprecedented levels of user data protection in a Cloud context".

Madsen, Henrik, et al. (2013) [7] This paper discusses contemporary computer paradigms and the most crucial components of their reliability. Fog computing is being investigated as a non-trivial extension of the Cloud, as well as the reliability of smart device networks, which are also being investigated. It is conceivable to build a reliable fog computing platform by merging the reliability requirements of grid and cloud computing paradigms with the reliability requirements of sensor and actuator networks in a distributed computing environment".

Hong, Kirak et al. (2013) [8] They claim that "the Internet of Things (IoT) is creating a new environment, notably the ubiquitous deployment of wireless and sensor devices, that will enable a wide range of future Internet applications." They explain that In this research, Mobile Fog was presented as a high-level programming paradigm for future Web apps that are dispersed, huge, and latency sensitive. Mobile Fog is a distributed, large-scale, and latency sensitive programming paradigm for future Internet applications. For the purpose of demonstrating the utility of Mobile Fog, the researchers looked at a variety of use cases for the programming paradigm, including camera networks and connected automobile applications. In addition, they employed simulation to evaluate the overall performance of the application".

Chou, Te-Shun et al. (2013) [9]In this paper, Cloud computing provides a powerful platform for a wide range of activities for both individuals and businesses, including the use of online storage space, the implementation of commercial applications, the development of bespoke computer software, and the creation of a realistic network environment, among other things. In this study, three cloud service models were assessed, and the nature of the cloud service models was utilized to determine the risks and hazards associated with cloud computing security. Using real-world cloud attacks, the hackers demonstrated their ability to breach cloud computing infrastructures. Countermeasures against cloud security breaches are also provided.

Modi, Chirag et al. (2013) [10] Cloud computing provides consumers with scalable on-demand services that are more flexible and need less infrastructure investment," argues the author. Because Cloud services are delivered via the Internet using old network protocols and formats, as well as the risks provided by newer architectures, security and privacy concerns have arisen. When it comes to cloud computing, security and privacy are constantly on the minds of those who are considering it. Specifically, this article looks at how these flaws manifest themselves and what may be done to make the system more robust to attacks.

Stojmenovic, Ivan et al. (2014) [11] According to the author, "fog computing" allows cloud computing and services to be extended to the edge of a network by extending them to the cloud. Users may access Fog's services either via a web browser or through a mobile application on their smartphone. This article examines a variety of real-world applications, including the Smart Grid, intelligent traffic lights in vehicular networks, and software defined networks. They discuss the origins of fog computing as well as its advantages. More security and privacy issues have been raised as a result of the current Fog computing paradigm being implemented.

Shankarwar, Mahesh U., et al. (2014) [12] Cloud Computing is always growing and displaying continual progress in the realm of computing," he wrote. A wide range of computing services, including cloud storage, cloud hosting, and cloud servers, are making it a popular choice for businesses and academics alike. On the other hand, there are a number of cloud security and privacy concerns to be addressed. The cloud computing paradigm still faces a significant hurdle in terms of security. Data leakage and the disclosure of personal data privacy are among these issues. There are several hazards to the user's sensitive data on cloud storage because of the cloud's security and privacy. Security and privacy concerns and solutions are examined in this study. In addition, cloud computing offers a wide range of security and privacy options.

Firdhous, Mohamed et al. (2014) [13] It represents the utility-based cloud computing" paradigm, which is a new computer paradigm in which computing resources are made available over the Internet at a cost-per-use basis. Cloud computing offers various advantages to users, including decreased costs, the elimination of system management responsibilities, more flexibility, greater reliability, and the ability to operate from any location. Cloud computing is becoming increasingly popular. Even while cloud computing has many advantages, it also has certain limitations that must be considered. The same reason that is considered a benefit is also responsible for the emergence of these limits. By putting cloud data centers on the Internet, a third party has gained access to sensitive information, resulting in significant and unanticipated network delay as well as unknown security issues. There are certain networks, such as sensor networks and the Internet of Things, where it may not be desirable to have location independence of processing in the cloud. They are referred to as location aware services since they need speedy processing depending on the user's location. For these reasons, cloud researchers have developed a new sort of cloud computing called fog computing, which puts the cloud system either on the edge of or very close to a private network as a workaround for the problems mentioned above.

Loke, Seng W et al. (2015) [14] is distinguished by its concentration on services and applications provided to mobile users via the use of airborne computing infrastructure. When considering the usage of drones as a service or as a fly-in, fly-out infrastructure, it is vital to underline the data management and system design challenges that might arise from their use. It is necessary to address the problems of Big Data that arise as a result of these applications, as well as optimizing airborne and ground infrastructure for the highest possible quality of service and quality of experience, as well as situational awareness, scalability, reliability, efficient scheduling, and interaction with users and drones via physical annotations." Optimization of airborne and ground infrastructure for the highest possible quality of service and quality of experience.

Bitam, Salim et al. (2015) [15] As explained by the author, "Cloud Computing" is a network access architecture that strives to "openly and ubiquitously share a massive number of computer resources." The majority of the time, digital customers may rent them from a service provider through the internet. Because of the increasing number of traffic accidents and public dissatisfaction with vehicle networks, current intelligent transportation system solutions are focused largely on improving road safety and preserving passenger comfort for passengers. The transportation services provided by the VANET-many Cloud are investigated, and various future study issues, such as security and privacy, data aggregation, energy efficiency, interoperability, and resource management, are highlighted.

Roman, Rodrigo et al. (2016) [16] According to the report, "cloud computing is unable to provide important requirements (such as low latency and jitter, context awareness, and mobility support) that are critical for many applications" (e.g. vehicular networks, augmented reality). In order to address these requirements, a range of paradigms, such as fog computing and mobile edge computing, have emerged in recent years. However, despite the fact that there are numerous similarities across these edge paradigms, there has been minimal examination of synergies in the present research. The majority of security assessments focus primarily on one edge paradigm while overlooking the others, which is especially true in this subject's unique case. The fundamental goal of this study is to explore the security threats, problems, and procedures that are present in all edge models, with an emphasis on the synergies and collaboration possibilities that may be realized. According to our results, all edge paradigms should take into consideration the evolution of other edge paradigms as they develop.

3. MATERIALS AND METHODS

3.1 GENERAL INTRODUCTION

During the panoply of definitions of the Cloud, we adopt the definition as follows: the Cloud is a concept which consists of deporting to remote servers a large quantity of recordings and computer processing, traditionally located on local servers, or on a user's workstation. A cloud offers computing services, in the form of on-demand services, we see that a user can access the Cloud servers thanks to an identification system, via an access device, such as a computer or a tablet, and an Internet connection. These servers are located in rooms dedicated to the execution of IT tasks related to the services offered. These rooms, called data centers, can contain thousands of servers. They are laid out in such a way as to easily install and administer not only the servers, but also the utility equipment of the data center. Cloud Computing adds to the notion of outsourcing services the notion of elasticity of resources and invoicing on use. These notions are possible thanks to the resource virtualization technique [4].

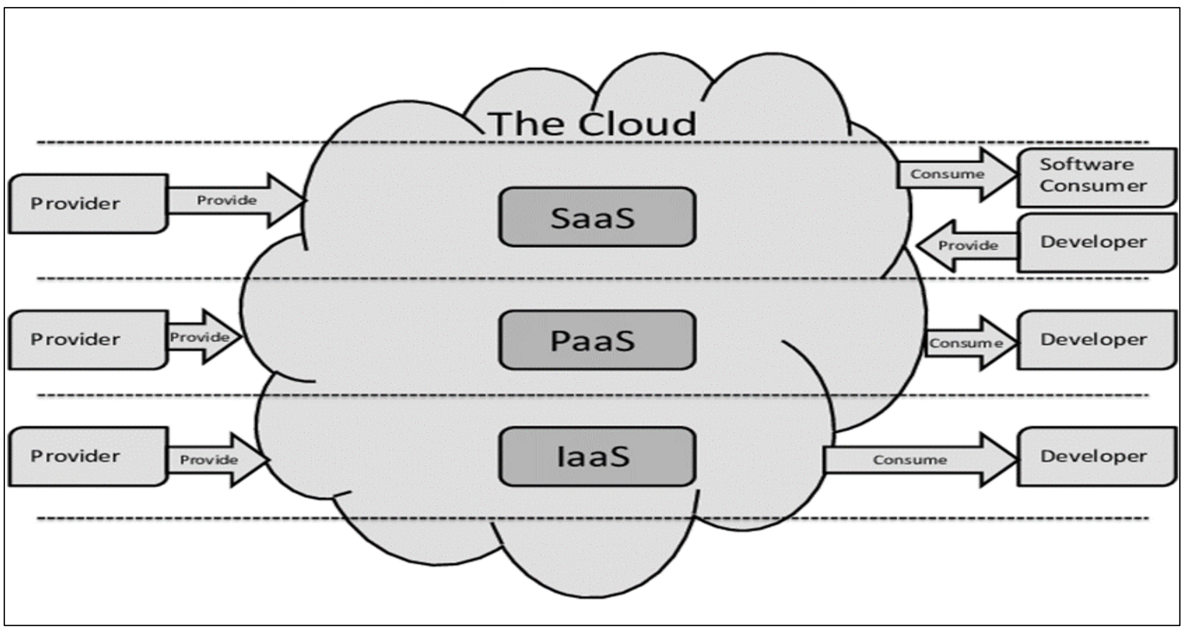

Figure 3.1: Cloud Computing Architecture [6]

Indeed, resource virtualization offers a layer of abstraction that emulates the behavior of hardware resources (processor, memory, etc.) of a server [5], [6]. It is also defined as being a set of techniques for grouping on a single physical medium computer resource that allow specific tasks to be carried out separately, as if they were executed on different physical media. As for the implementation of resource virtualization, it consists of providing a logical environment independent of the physical support. With the concept of virtualization, also emerges the notion of virtual machine (VM) which is defined as an execution environment which reproduces the behavior of a server [7]. A virtual machine is composed of a processor, RAM (Random Access Memory), hard disk and network interface cards which are virtualized components. This is the unit responsible for executing tasks sent by users. Indeed, after receiving these tasks, a task manager sends each task to the virtual machine's queue responsible for its execution. This task manager uses a scheduling algorithm that defines how tasks are assigned to virtual machines. On the other hand, the specification of the hardware characteristics of virtual machines, desired by the user of a Cloud, constitutes one of the challenges of Cloud service providers, which leads to de ne a contract called SLA (Service Level Agreement). An SLA contract contains a set of clauses that the supplier undertakes to respect [8]. Furthermore, the contract describes the criteria that have been agreed upon by both clients and cloud providers in terms of resources, service level agreements (SLAs), customer expectations, and cloud provider obligations. When it comes to service providers, the execution of such agreements is crucial because the absence of such agreements may drive customers away and harm the company's growth. Quality of service (Quality of Service: QoS) is measured by the ability of an application to execute, as well as the dependability and availability of the application [9]. A variety of metrics can be used to evaluate the overall quality of a service. These indicators include response time and server utilization, to name a couple of examples. The response time is the amount of time that elapses between the time that a task is submitted to the server and the time that the server returns the result of that task's processing. The proportion of tasks handled by the servers is calculated by multiplying the number of tasks handled by the servers by the total number of tasks received by the Cloud by 100 to arrive at the percentage of tasks handled by the servers. Our research is aimed at reducing the amount of energy consumed by clouds. One approach of reducing this use is through migration, which includes shifting resources from one server to another [10]. It may also be used to host several virtual computers on a single server, which is quite convenient. Server consolidation is an ultimate objective that involves shutting down servers that are only sometimes used in order to reduce the amount of data center space required. However, while the DVFS (Dynamic Voltage and Frequency Scaling) method enables for the frequency and voltage of the server processor to be modified, this has an impact on the server's overall energy consumption, as well as the time it takes for each application to execute. [11].

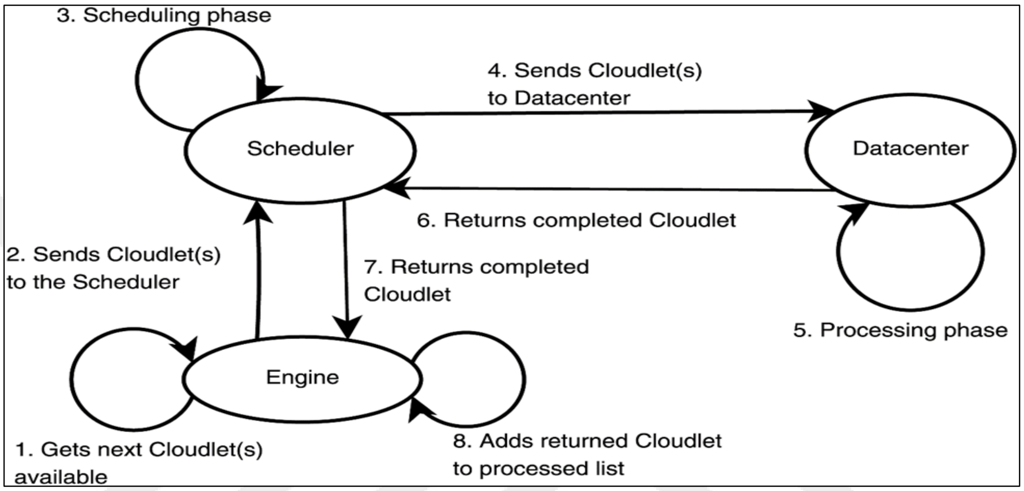

Figure 3.2: Cloud Computing Dynamic Voltage and Frequency Scaling [8]

3.1.1 Cloud Computing Energy Optimization

The minimization of energy consumption is a hot topic that concerns several areas [18]. In this context, it is important to properly present the framework of our research, hence the characterization of cloud computing. In this chapter, we will first introduce the functional characteristics of cloud computing, followed by the deferent cloud models and topologies. Then, we will present the advantages and disadvantages of this environment. Subsequently, the deferent techniques for reducing energy consumption will be presented. We end this chapter by presenting the research challenges related to the Cloud.

3.2 ENERGY CONSUMPTION CARICHTARISTICS

In this section, we will present the main functional characteristics of the Cloud which are: on-demand self-service, resource sharing, measured service, universal access, as well as the elasticity of resources [19].

3.2.1 On-Demand Self-Service

On-demand self-service allows the user of a Cloud to be able to manipulate remote resources (servers, networks, storage, applications, etc.) in real time, according to their needs, and without having to go through manual configuration

3.2.2 Sharing of Resources

The material resources of a Cloud provider are shared and pooled between the different users of the services offered. This sharing makes the exact location of user data impossible to determine, which can pose a real problem for companies that are subject to numerous regulatory constraints regarding the location and control of data. In return, the economies of scale achieved reduce the costs of the cloud provider.

3.2.3 Metered Service

The consumption of cloud services is monitored and metered. It is the subject of reports whose performance indicators are aligned with those of the client. This concept is defined by setting up a meter to bill the consumption of a customer's services. These precise measurements allow billing of users who will only pay for the resources they have used and for the duration of consumption of these resources.

3.2.4 Universal Access

All Cloud resources must be accessible and made available to the user anywhere and simply through the network, regardless of the client used (server, PC, mobile client). Indeed, large providers distribute data centers around the world to provide access to their systems in less than 50 ms, from any location [19].

3.2.5 Elasticity

As the demand for computing resources grows, the Cloud's ability to scale to fulfill that demand grows as well. On the other hand, if these abilities are not used, they can swiftly erode and become ineffective. The elasticity of cloud services creates the appearance to users that they have a limitless amount of available capacity that may be used at any time. It helps corporate clients to handle spikes in activity that their own infrastructure could not handle, as well as to explore novel application possibilities that would otherwise be impossible.

3.3 CLASSIFICATION

In terms of models, clouds are characterized by service models and deployment models. In this section, we will present and explain these models.

3.3.1 Service Models

These models describe the different categories of services that are accessible to customers from a Cloud platform and that a provider can offer. The best known are: "software as a service" (Software as a Service: SaaS), "platform as a service" (Platform as a Service: PaaS) and "infrastructure as a service" (Infrastructure as a Service: IaaS). [21]. These three services are which compares the classic model to the three cloud service models, in terms of resource management. As shown in the figure, the company manages all its resources by itself in the classic model, whereas in cloud computing, the company manages part of the resources, and the rest is managed by a cloud provider. More specifically, the "software-as-a-service" or SaaS model is an economic model where the customer uses the supplier's applications, which are hosted on the Cloud. In this model, the application is managed by the client through an interface, usually a browser. The customer's responsibility begins and ends with the management of its own data and the interaction with the user. Everything that happens between the application and the infrastructure is the responsibility of the provider. SaaS solution vendors include Google Apps, Oracle on Demand, SQL Azure [5].

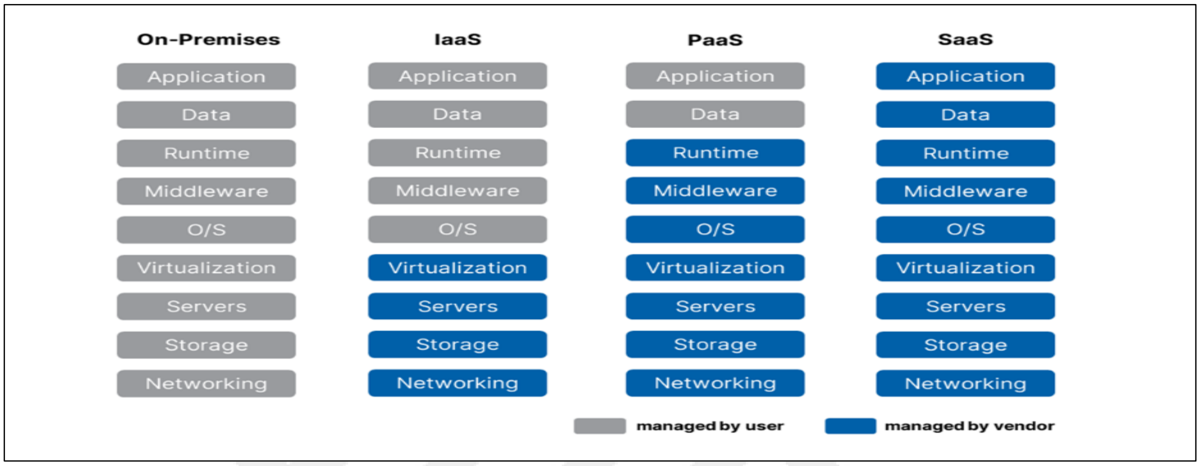

Figure 3.3: SaaS PaaS and IaaS Services in Clouds [20]

The "Platform-as-a-Service" or PaaS model provides build and runtime environments for applications using vendor-supported languages and tools. Thus, the user develops his own applications which are stored in the Cloud. Coffered, Cloud Center, GoogleAppEngine and Windows Azure Platform are among the providers of PaaS solutions [22]. The "infrastructure-service" or IaaS model is used for pooling infrastructure between customers and companies. In this model, the customer controls applications, server software, databases, virtual systems, while the provider manages virtualization, server hardware, storage and network infrastructure. Among the IaaS solution providers, we cite: Amazon Elastic Compute Cloud (EC2), Eucalyptus, RackSpace Cloud and Terremark [23]. These providers o er direct access to their hardware resources. With Amazon, considered a classic provider of IaaS-type solutions, a customer accesses a computer in the form of an image of a virtual machine, on which the operating system and applications are installed. All the service models cited above share the same characteristics [3]: Customers rent services instead of buying them, thus helping to transform capital expenditure into operational expenditure; Providers are responsible for all the infrastructure behind the operation of cloud services, in terms of maintenance, administration, capacity management and incident management; Services are, in theory, available much faster than those provided by traditional server-based solutions. Indeed, by adopting the Cloud, more storage, projects, and applications are guaranteed.

3.4 TOPOLOGY DEPLOYMENT

The topology of a data center defines the way in which the physical equipment constituting the data center are interconnected [26]. In this section, we will introduce the most used data center topologies, namely tree topology and recursive topology [27], [28].

3.4.1 Tree Topology

Tree topology is also known as hierarchical topology. In this topology, the network is composed of three levels. These levels are core, aggregation and access. The kernel represents the top level of the tree topology. Indeed, it contains switches that are directly connected to lower-level switches (aggregation switches) [6]. The core represents the access point of the cloud. Indeed, requests sent by cloud users first pass through this level. The aggregation level contains switches connected to both access switches and core switches. Its main role is to route cloud user requests to the appropriate access switches [6]. The access level provides the connectivity to allow access to available server resources. The design of this level is influenced by decision criteria, such as server virtualization. Among the most used switches at the access level, we can cite: EoR (End-of-Row) type switches and ToR (Top-of-Rack) type switches [6]. The variants of the tree topology most used in data centers are: Basic Tree and Fat Tree [27], [29]. The basic Tree topology consists of two or three levels of switches and routers, with the servers as leaves. In the three-tier topology, as shown in gure 2.3, we find the kernel at the root of the tree, the aggregation in the middle of the tree, while the last tier consists of the access switches connected to servers. For the two-tier topology, aggregation switches do not exist. Core switches are interconnected. The disadvantage of this topology is that the number of servers is limited by the number of core switch ports. The Fat Tree topology is an extended version of the tree topology [30]. In this topology, the interconnection between the levels follows certain rules. Let n be the number of physical ports of the access switches. In this configuration, half of these ports (n 2) are connected to the physical ports of the servers, while the other half of the ports of the access switches are connected to the ports of the aggregation switches.

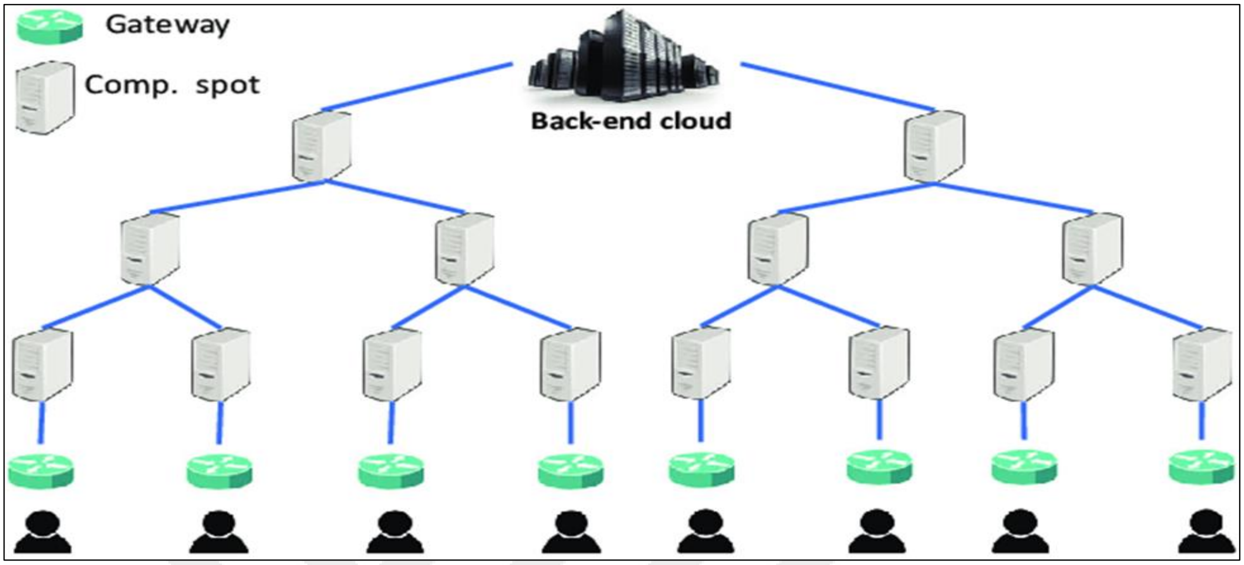

Figure 3.4: Tree Topology in Clouds [27]

3.4.2 Recursive Topology

Recursive topologies are used for bandwidth-intensive services, providing high network capacity for data center traffic [31]. These topologies are also based on the notion of levels. However, they use lower-level structures, such as cells, to build higher level structures. Thus, servers in recursive topologies can be connected to switches at different levels or even to other servers. Among these topologies, mention may be made of: The subgroup is represented by DCell0 and each server in the subgroup is connected to a server in another subgroup [28]. Thus, the construction of the DCellk topology is recursive.

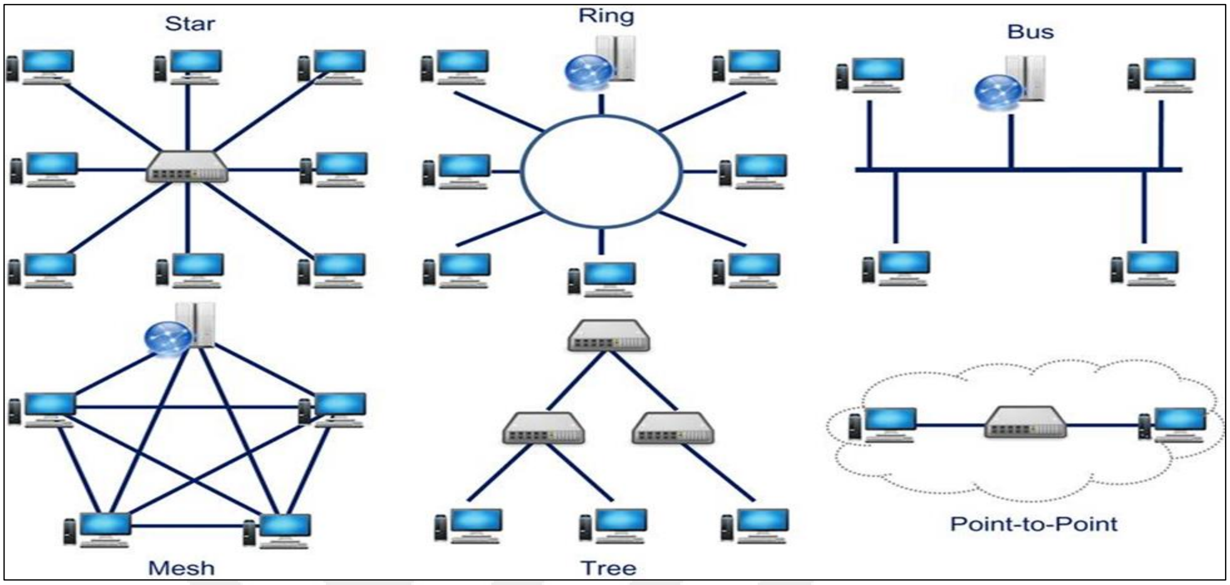

Figure 3.5: Recursive Topology in Clouds [29]

3.5 ADVANTAGES AND DISADVANTAGES

Cloud computing remains surrounded by several ambiguities, especially when it comes to clearly identifying the advantages as well as the disadvantages. In this section, we will first present the advantages, then the disadvantages related to the different aspects of cloud computing, such as business management, security, service availability and energy consumption. Regarding business management, Cloud Computing offers companies various advantages. Indeed, the Cloud allows companies to save time and increase productivity thanks to the availability and elasticity of IT resources. In addition, it offers billing systems based on the actual consumption of services, as opposed to flat-rate systems for which the customer pays even if he does not consume anything [32]. In order to assist in better collaboration between individuals, access to company applications and data can be done from any terminal and in real time. The company that opts for Cloud Computing also experiences an organizational gain. Indeed, the consistency of the software park is no longer achieved through the installation and updating of applications on the terminals, but simply amounts to using the latest version of applications available on the Cloud. This consistency of the software fleet thus facilitates the modification of data and synchronization with online documents. Regarding security at the

Cloud level, it is ensured by redundancy at the level of data recording [33], which is advantageous. Indeed, in the event of a data center failure, the user who uses the Cloud will not lose his documents, because his documents are saved there in several copies. It is therefore safer to opt for the Cloud in terms of data storage instead of using your personal hard drive [33]. To have a quality service, the availability of the services o ered by a Cloud provider must be between 98 and 99.99%, including server downtime for maintenance or for unexpected interruptions [32].

The Cloud potentially integrates local networks and remote networks in the link between client systems and servers. The failure, even transient, of a remote network is not treated in the same way as that of a local network. In the event of disappearance or unavailability of part of the network infrastructure, since the basic data exists in several instances, a priori on several distinct sites having different local networks, it is possible in theory to achieve a available data instance. The adoption of green computing represents another advantage that the Cloud offers. Indeed, Green computing or green IT brings together all the technologies making it possible to limit energy consumption and the waste generated by the IT industry [34].

Thanks to server consolidation and virtualization, the number of servers is reduced, which limits the power consumption of UPSs and the consumption of air conditioners to cool the UPSs and servers [35]. On the other hand, the Cloud has several disadvantages. In terms of business management, the Cloud disrupts business habits, which can be a risk factor. In fact, in a classic scheme, any service must be approved by the financial department. With the Cloud, the user can buy the service himself and pay for it with his personal credit card, which represents a great risk for the company. According to some analysts [36], Cloud Computing represents a threat to the security of information controlled by a third party. Indeed, the risks incurred when the data is consolidated on a single server are very high, because during a computer failure, it may prove di cult to recover all the information in the absence of a copy already saved.

Therefore, the duplication of information on several remote sites is imperative for security. On the other hand, to overcome the problem of confidentiality of documents, computer encryption seems to be the best solution. However, Cloud Computing essentially uses virtual machines, so the main danger comes from the limited number of ciphers linked to the encryption that these machines can exploit. Thus, a study conducted by McKinsey [36] also shows that security is the main obstacle to the adoption of cloud services. Indeed, for the purposes of its survey, McKinsey questioned 462 IT decision makers and 264 non-IT decision makers on the main barriers to the adoption of cloud computing. The result is unequivocal: safety management comes first. The management of compliance risks as well as the company's exposure to these risks come in second place. It is also about compliance with legal obligations in terms of confidentiality of information concerning customers, employees, electronic payments, etc.

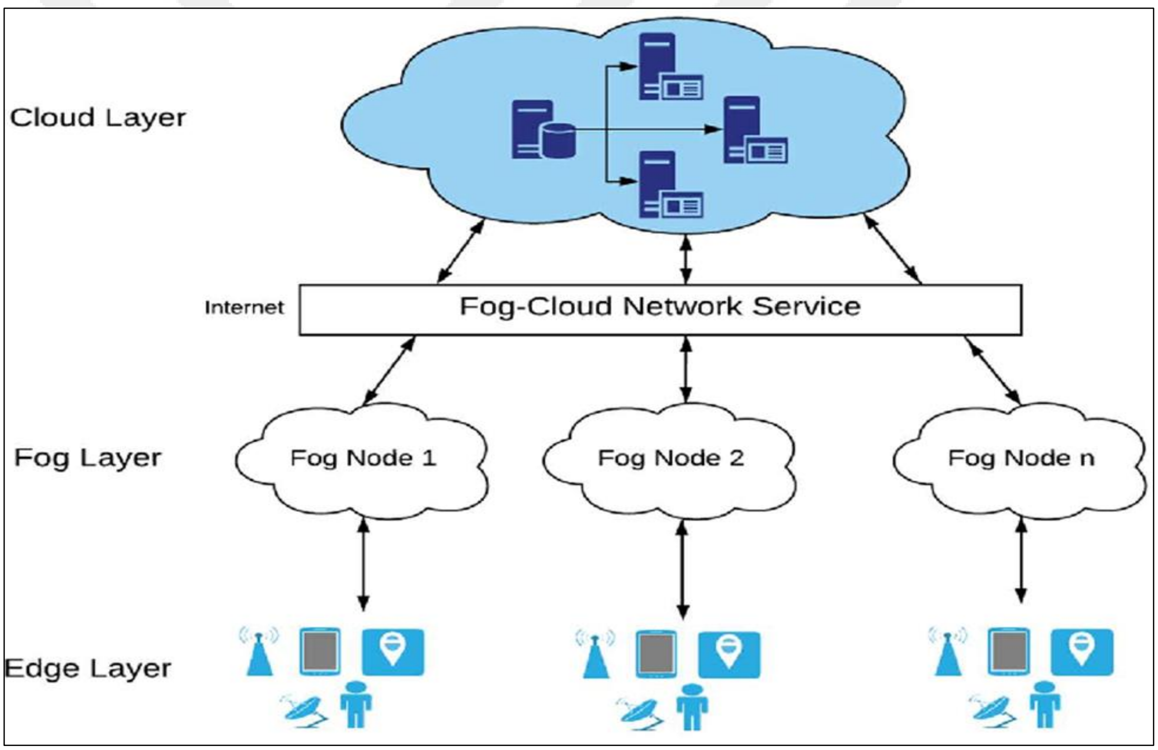

Figure 3.6: Cloud and Fog Network Service [31]

3.6 REDUCING ENERGY CONSUMPTION

Several techniques exist to reduce energy consumption at the level of the various components of a data center in this section, we will present the main energy reduction techniques, either at the level of physical equipment, such as the DVFS technique [39], or at the level of virtual resources, such as the scheduling process [40] and migration. virtual machines [41].

3.6.1 DVFS Technique

Servers and switches, which are joined together in a preset configuration, are the most important components of a data center. The total energy consumed by a data center is the sum of the energy consumed at the server level and the energy consumed at the switch level. It has been demonstrated that the quantity of power consumed by the switches is directly proportional to the level of workload balancing [42]. However, one aspect of our study involves investigating server-level workload balancing. There are a variety of approaches that may be used to reduce energy usage in data centers, including lowering the energy consumption of servers. We will thus concentrate on hardware devices that incorporate both software and physical components, such as virtual machines and hard drives, in order to better understand them.

However, the DVFS technique is more focused on the CPU, which accounts for the majority of a server's energy use [43]. This strategy attempts to lower the energy usage of cloud servers by utilizing the idle period of each server's processor. [39].In effect, a processor executes multiple tasks with inter-task relationships (eg, precedence constraints) simultaneously. These inter-task relationships usually result in dead time (idle time) between tasks. This dead time associated with a task is used to execute the task with minimum voltage and frequency. This in turn leads to a reduction in the energy consumed. Indeed, the power used in a processor is defined as follows and expressed in Watt (W) [44] the value of electrical intensity that the processor can support by applying a voltage of 1 V. This parameter C is a constant that characterizes each processor. The frequency f is the switching frequency of the processor expressed in Hz.

This frequency represents the speed with which the processor processes the instructions. V is the processor supply voltage expressed in Volts (V). it is deduced that the decrease in the energy consumed by the processor requires a change in the level of the torque (f, V), by exploiting the idle time of the processor. which illustrates how the DVFS technique works. We have three representations of the power used by a processor. From equation 2.1, this power is proportional to the frequency f and to the square of the voltage V 2. In order to verify the impact of the variation in frequency and voltage on energy consumption, A the power P, calculated from a frequency f and of a voltage V. These frequency and voltage values are used by the processor to execute a task which ends at t1, a time before t2 which represents the start time of the next task. In this case, the processor experiences an idle time between t1 and t2.

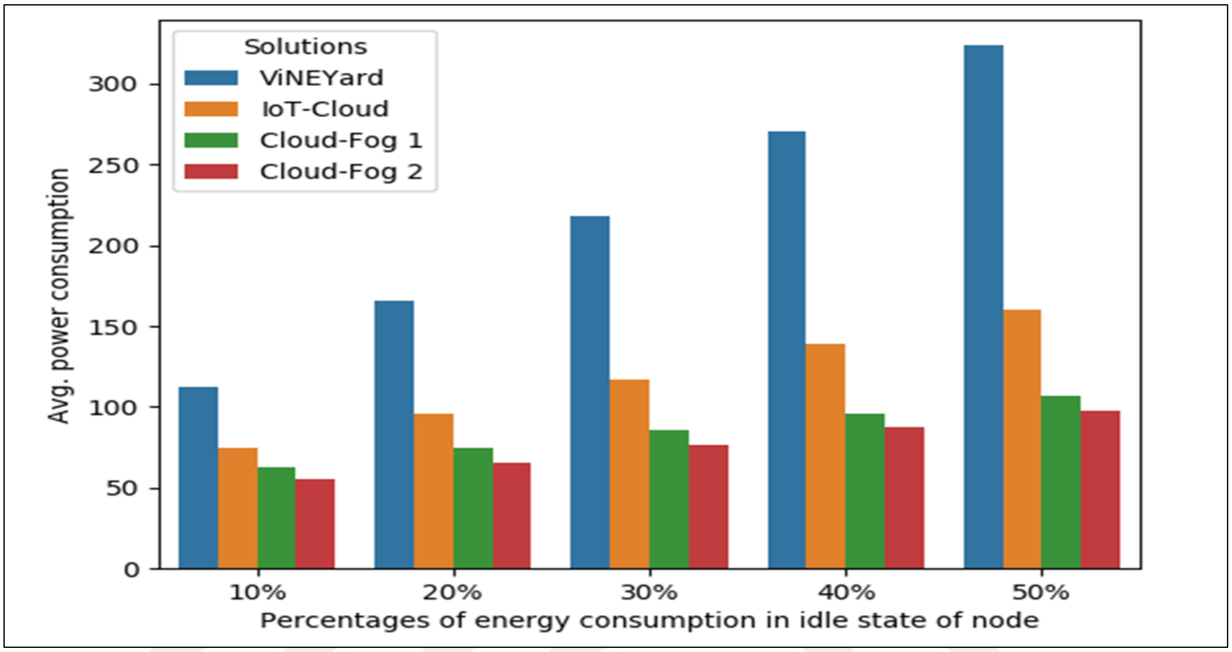

Figure 3.7: Average Power Consumption in Cloud and Fog Computing [33]

To reduce the power, we will start by reducing the frequency. In this example, we are going to divide the frequency f by two, then we are going to present the power, corresponding to this frequency, the power has been halved (P/2). Thus, the execution time is doubled, since the frequency is the inverse of the execution time. However, this task execution time t2 has not timed out for the next task to execute. presents the power calculated from the frequency f/2 and the voltage V/2. This power was decreased to P/8, and the execution time did not exceed the timeout, since the voltage does not influence the execution time. Thus, the energy consumed has been reduced. From this fact, we deduce that the DVFS technique decreases the energy consumption, by reducing the frequency and the voltage. into account time constraints (delay constraints) and constraints on the use and availability of resources. Before tackling the scheduling problem, it is important to de ne the basic concepts of the scheduling process, such as tasks, resources and constraints. [40], [45].

A task is an elementary entity located in time, by a start date, an end date, and whose completion requires a previously defined duration. It is made up of a set of operations which requires certain resources for its execution. These resources are technical means intended to be used for carrying out a task and available in limited quantity. Within the framework of a Cloud, several types of resources are to be distinguished: virtual machines, CPU and disks. A resource is renewable if, after having been allocated to one or more tasks, it is available again. As for the constraints, they express restrictions on the values that the decision variables can take. We distinguish the time constraints which express the constraints related to the time allocated to the duration of the tasks (starting time, stopping time), and the constraints of use of the resources which express the nature and the quantity of means used by the tasks. , as well as their availability. which illustrates a scheduler made up of a controller which implements a scheduling algorithm and a queue which contains tasks on which we apply the scheduling process.

$$E_{DC^i}^{static} = \sum_{a \in CN^i} (P_a^{idle} \cdot T) + \sum_{b \in SN^i} (P_b^{idle} \cdot T) + \sum_{c \in NE^i} (P_c^{idle} \cdot T) \quad (1)$$

Let Edc be the number of tasks in the le, task i, i = 1, ..., m-1 and task j, j = i+1, ..., m, represent any task in the. This scheduling process consists of the following steps: Task prioritization phase: this phase establishes the order of the starting tasks according to their priorities. More precisely, the starting tasks are those sent by the user from the Cloud, and which arrive at the scheduling system as being the entry point of the Cloud. In our study, we consider that no task has priority, so that after this phase, we have an ordered list which constitutes. In our case, this order is ordered according to the principle of "first come first served". Resource allocation phase: this phase reserves or allocates a set of resources necessary for task scheduling. At this phase, the controller calculates the number of virtual machines hosted in one or more servers, among the three servers. These virtual machines are necessary for the processing of each task i, j, i = 1, ..., m-1, j = i+1, ..., m. Scheduling phase: this phase selects the resources (virtual machines, CPU, disk, memory) among those previously allocated. This phase schedules each task according to an algorithm that defines how tasks are assigned to virtual machines through the switches in the data center. In the example each task i is assigned to a VM installed on server

1. The way of assigning tasks to virtual machines has an impact on energy consumption. In other words, the scheduling process plays an important role in minimizing the energy consumed. Indeed, the energy consumption depends on the number of resources allocated during the resource allocation phase of the scheduling process.

3.6.2 Migration of Virtual Machines (VMs)

Migration of VMs consists of transferring VMs from one server to another. This transfer includes the transfer of memory, CPU, and disk to the destination server. In effect, this transfer is carried out through the switches that link the servers. However, the migration process can also be used in an Intercloud. In other words, the migration can be performed between servers located in different clouds. As part of our research, we are studying the migration carried out in the same Cloud, which makes it possible to solve, among other things, problems of fault tolerance, or continuity of services, when a particular server must be replaced for Hosted Services maintenance or upgrade tasks. Migration techniques are classified into cold migration (non-live migration) and hot migration (live migration) [41].

Cold migration of a virtual machine consists of performing the transfer of this machine by deactivating it in the source server before the transfer. Once the transfer is complete, the virtual machine becomes active again in the destination server. In this type of migration, the total migration time is equal to the downtime of the VM, which represents the interval between the instant of deactivation of the virtual machine at the source server and the instant of reactivation of this machine at the destination server.

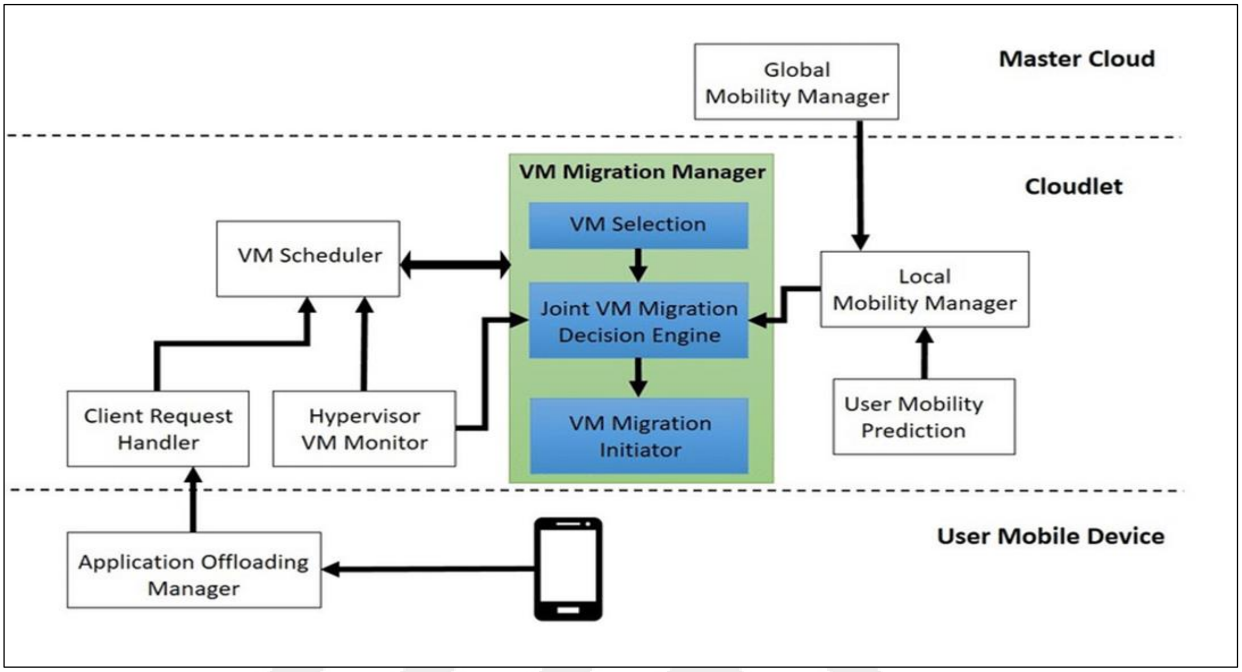

Figure 3.8: Transferring VMs from One Server to Another

3.7 FOG COMPUTING

Over the years, cloud computing has established itself as the main option to provide computing and storage resources for IoT devices and applications. The communication between these two paradigms allows the computational demand to be met through data centers geographically located in different points around the world. However, a common problem in cloud computing is latency. Since applications may be available for geographically distributed devices, the connection to data centers may depend on factors such as location and bandwidth. This also directly affects real-time applications. To overcome these challenges, Bonomi et al. (2012) proposed a new paradigm called fog computing.

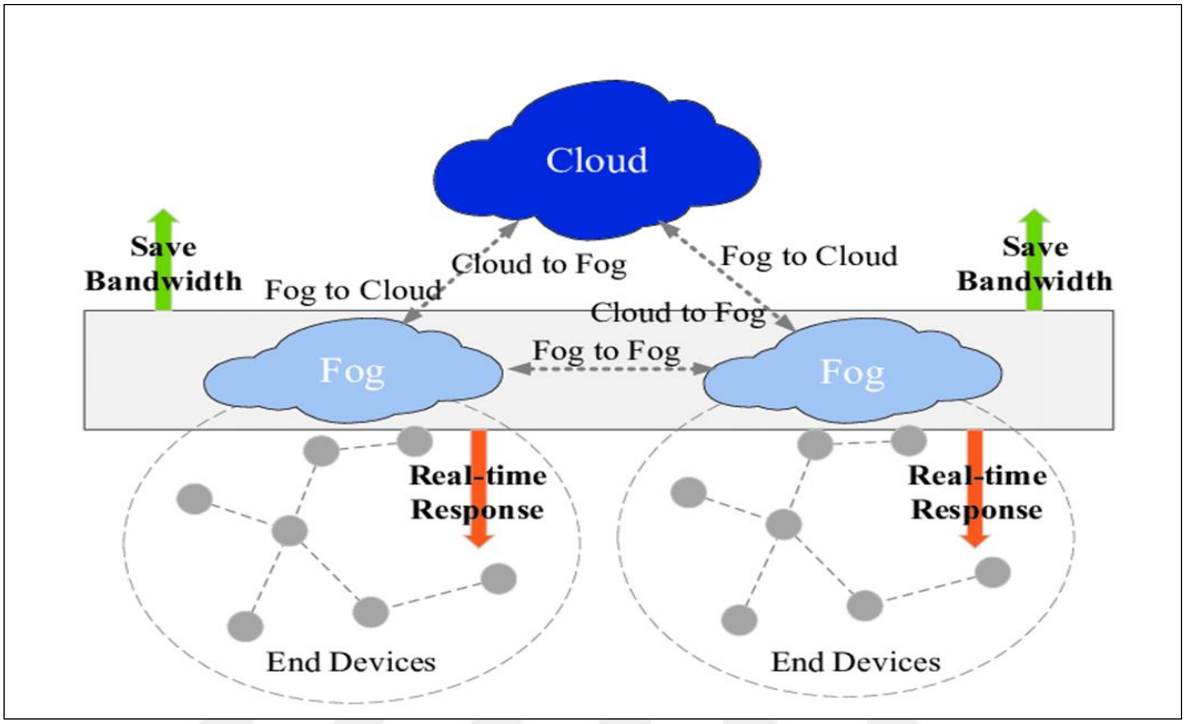

Figure 3.9: Fog Computing Architecture [42]

3.7.1 Characteristics

Fog computing, a new paradigm that mixes distributed computing, networks, and storage, allows cloud data centers to interface with edge devices. A summary of its most distinguishing traits is provided below.

The heterogeneous nature of fog computing is a distinguishing characteristic. It is a virtualized platform that provides cloud computing as well as processing, network, and storage capabilities to end devices of various forms and shapes.

Because of the high-quality services provided by fog computing devices, both mobile and fixed end devices can profit from the widely disseminated adoption of fog computing technology.

As a solution to the lack of end-user support at the network's edge, the fog computing concept was established in order to provide high-quality services to users at the network's perimeter.

A real-time interface is required by many fog computing systems, such as traffic monitoring systems, for them to function properly.

Mobility support is required for many applications in fog computing settings, including the ability to connect directly with mobile devices via protocols such as Cisco's location/ID separation protocol, which separates host identification from host identity.

Large-scale sensor networks, which can be used for environmental monitoring or smart grid applications, can be built using inherently scattered systems, which need the utilization of distributed computing or storage resources.

Fog computing nodes, such as the mobile device gateway and most wireless access points, are frequently encountered in wireless networks.

It is necessary for fog computing components to be able to communicate with one another in order to provide a wide range of services. Smart homes, health care, connected vehicles (SVCs), and big data analytics are just a few of the applications for fog computing, which has many of the same characteristics as fog computing but is not as expensive. Fog computing has been advocated for a variety of applications in the literature, and we'll go over a few of those in greater detail further down the page.

3.7.2 IoT Applications in Fog Computing

In the literature, fog computing has been supported by a range of applications that have been developed. The three major application types are as follows: I real-time applications; (ii) near real-time applications; and (iii) network-introduced applications.

Real-time apps can be classified as either urgent or instant, depending on the user's demand for urgency or immediate action.

When an event happens and the data is used to process it, near real-time applications are susceptible to a temporal delay owing to data processing or network transmission. Smart traffic lights (as well as video streaming games and healthcare) are two examples of applications.

As instances of this tendency, we can point to smart grids, smart cities, and smart vehicles, among others. Fog computing can be introduced into a network to reduce the amount of traffic passing through the processing core for applications that do not require real-time processing and transmission. Real-time applications, on the other hand, are largely reliant on reaction time to achieve the intended results. To do this, efficient resource management is essential to select the suitable items that correspond to application requests while also ensuring that the relevant Quality of Service (QoS) criteria are satisfied.

Table 3.1: QoS Features Compared in Cloud and Fog Computing

Feature requirements	Cloud Computing	Fog Computing
Latency involved	High, depends on User to DC route	Low
Response time	Several Minutes	Milliseconds
Service Location	Inside Cloud Data Center via internet	Edge of Cloud network
Time for data storage	Months or years as per contract	Transient
Hops between user & server	Multiple	One
Location Awareness	None, need manual routing	Very local
Architecture	Centralized	Distributed
Last mile connectivity	Broadband, MPLS, Leased line	Wireless
Attach probability on data	High	Low
End to end Security	Cannot be defined or controlled	Can be defined
Nodes to collect data	Very few	Unlimited
Mobility support	Limited support	Supported

3.7.3 Resource Provisioning

It is an important problem in many different fields of study to consider how resources are allocated. Technical breakthroughs have produced results that, in a sense, highlight the evident necessity of making the most efficient use of available resources. Enhancing provisioning efficiency in this context is associated with the optimization of several objectives, such as lowering costs, lowering energy consumption, and increasing the utilization of resources. Resource provisioning is the process of allocating, de-allocating, and monitoring system resources, and it is a subset of system administration. Several steps are required in order to provide effective and quality-aware resource management, also known as resource provisioning. These actions are listed below. When evaluating the quality of service, a variety of elements must be considered, including service level agreements (SLAs) and time-dependent demand (QoS). Resources must be changed more frequently in response to changing system workloads, which increases the complexity of the resource provisioning technique over time. The most significant resource provisioning aspects of the fog computing architecture are explored in detail in the following paragraphs.

Each piece of a network, from sensors to actuators to fog computing devices, is monitored for resource use and availability. They keep track of the performance and status of the applications and services that are running on the infrastructure, and they pass this information along to other services as needed to keep things running smoothly.

Resource management is a critical component of the architecture because it allows it to meet application-level quality of service requirements while also preventing resource waste. It is made up of components that manage resources in a consistent manner. To pick the most appropriate candidates to host an application module, provisioning components must be able to track available resources using information provided by the monitoring service. In this case, the process is called as orchestration, and it involves assessing system monitoring data and coming up with a strategic approach for resource allocation. A strategy for how resources will be assigned to various services, as well as how task requests will be handled, is produced by orchestration.

When it comes to identifying how much energy is being utilized by each IoT node, most Internet of Things solutions run into significant difficulties. In contrast to cloud data centers, where a small number of devices with fluctuating power usage can be controlled, it is impossible to govern energy consumption in fog computing. Prior to implementing resource management applications and policies in production environments, it is necessary to assess the impact of these applications and policies on energy consumption. Proactive resource provisioning is preferable to reactive resource provisioning.

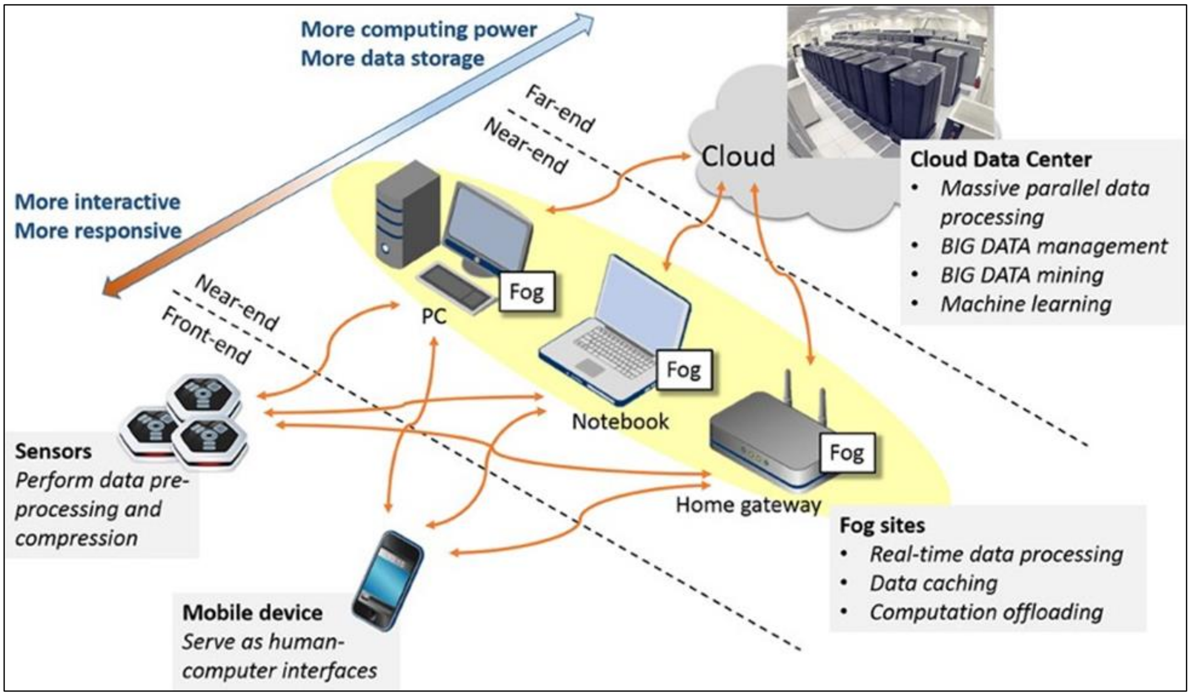

Figure 3.10: Energy Monitoring Scheme in Fog Computing [43]

A proactive scenario is one in which the state of the system is checked on a regular basis and resources are allocated in accordance with the system's needs. When predicting future resource demands, it is not uncommon for predictive mechanisms to be used to take necessary action before the system really requires it. A reactive scenario, on the other hand, deals with occurrences like as failures, alert events, and so on in a direct and immediate manner. As an example, if a monitoring component generates an event signaling a problem in the environment, the system responds by taking action to avoid the problem by consuming computational resources that are more than their stated limitations. Furthermore, depending on the purpose at hand, there are a range of approaches that can be used to improve resource supply. To maximize node utilization, service distribution may be done in a number of ways. For example, giving the highest quantity of services to the lowest number of nodes possible can maximize node use. By prioritizing service allocation in nodes that are located closest to user locations, latency optimization can be achieved. It is possible to approach the optimization problem in two ways: either via the use of exact mathematical methods or using heuristic approaches such as linear programming. Rather of seeking the perfect solution, heuristics are algorithms that approach real-world circumstances and seek near-optimal solutions rather than the optimal one. To develop a heuristic solution, it is usually required to provide a more complete description of the problem and the surrounding environment. Heuristics include techniques such as local search and greedy algorithms, to name a few.

3.7.4 Autonomic Computing

IBM has proposed the autonomic computing architecture which encompasses several management features, including self-configuration, self-healing, self-optimization, and self-protection. These characteristics require the system to know the context in which its distributed entities evolve. Enabling context awareness in a heterogeneous environment requires an appropriate architecture and knowledge management model that best abstracts the heterogeneity, distribution, and dynamics of pervasive environment entities. Different terms are related to autonomic computing, such as the term self-management), autonomic systems Furthermore, the mentioned challenges for distributed systems are also addressed by an area of research called Organic Computing Self-adaptation or self-adaptive software are other widely used terms that, have a more limited domain than autonomic computing, and are included in it. Despite different names, the concepts of the cited domains are extremely related, and, in many cases, they can be used as synonyms The case properties and control loops are the most common elements addressed by these types of works, both presented in the next subsections. One of the great motivations for the construction of this type of systems is the search to eliminate or at least reduce the need for human intervention in the software, resulting in cost reduction.

4. PROPOSED METHOD

4.1 CHAPTER OVERVIEW

This chapter presents the design aspects of the proposal, which constitutes a central contribution of this master's dissertation. In this sense, they are conceived as premises and from the conception of the autonomic architecture as designed. The figure below figure 4.1 shows the parameters defined in MATLAB

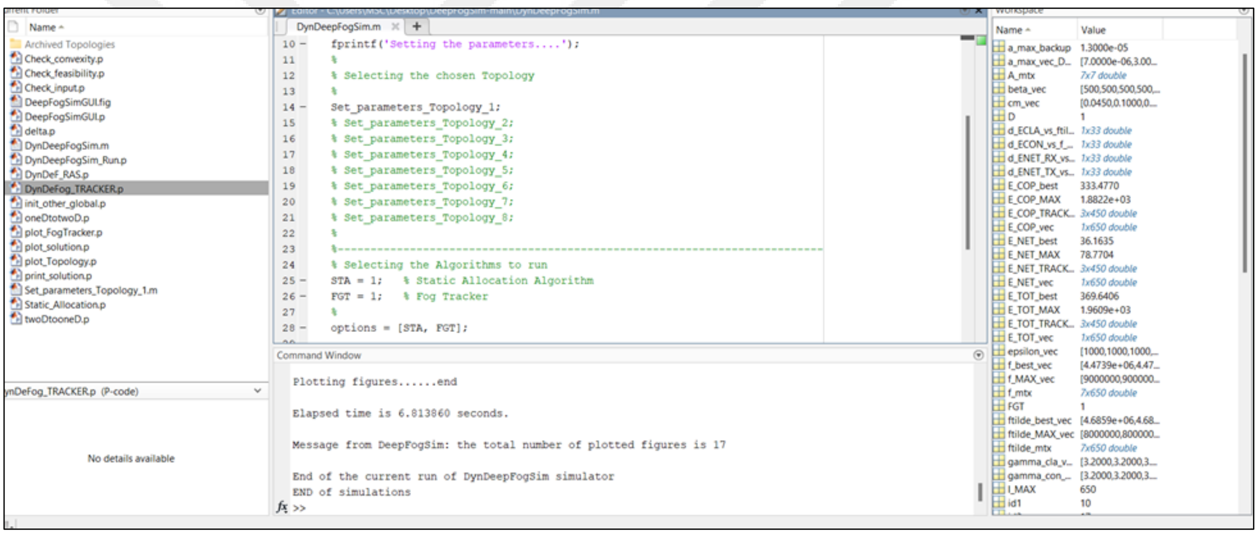

Figure 4.1: Setting parameters in MATLAB

4.2 SYSTEM MODEL

In our model's reference architecture, the cloud, fog, and Internet of Things are all represented by three stacked levels: When a cloud-based platform is used to host applications that are not supported by the fog, it is referred to as a "cloud-based platform." Network equipment that sits between the cloud and Internet of Things devices, such as routers, access points, and edge routers is referred to as fog. It is conceivable to build a management system that is completely independent of the cloud, but only if the nodes have sufficient processing power to do it. Thus, the Fog Control Nodes (FCN) and Fog Node types of devices have been developed to address this problem (FN). This form of fog node has more computing power and bandwidth than other types of fog nodes, which are frequently used as Internet of Things gateways. It provides a link between the architecture's four modules, which will be examined in further depth in Section 4.3, and the rest of the system's nodes. Additionally, the fog nodes are only capable of hosting monitoring modules, which are not included in the application modules they host. The Internet of Things (IoT) is dependent on the sensors and actuators that make up this layer. When connecting peripheral computing equipment to these, a variety of methods, including hard links, ad hoc wireless networks, and even on-board the device, are routinely employed. The deployment of Internet of Things applications is dependent on the distributed data flows created by these devices. The fog and cloud layers are the second and third tiers, respectively, of a three-tier architecture, with IoT devices at the bottom of the tier structure. The term "fog node" refers to any object that has the potential to host at least one application module. A finite collection of constants can be used to represent computing capacity when measured in terms of the amount of CPU, memory, disk, and bandwidth resources available to the system. The topics discussed in the next portion of the architecture system model representation are the application model and the resource model, respectively. It is the first of them that is concerned with the definition and mathematical expression of the properties of each application. For each application to be assigned to each fog node, the resource model provides the properties of each resource, its notation, and the constraints for each application that is specified in the resource model. These models serve as a foundation for provisioning algorithms since their attributes are linked to the provision of decision-making processes and allocation constraints.

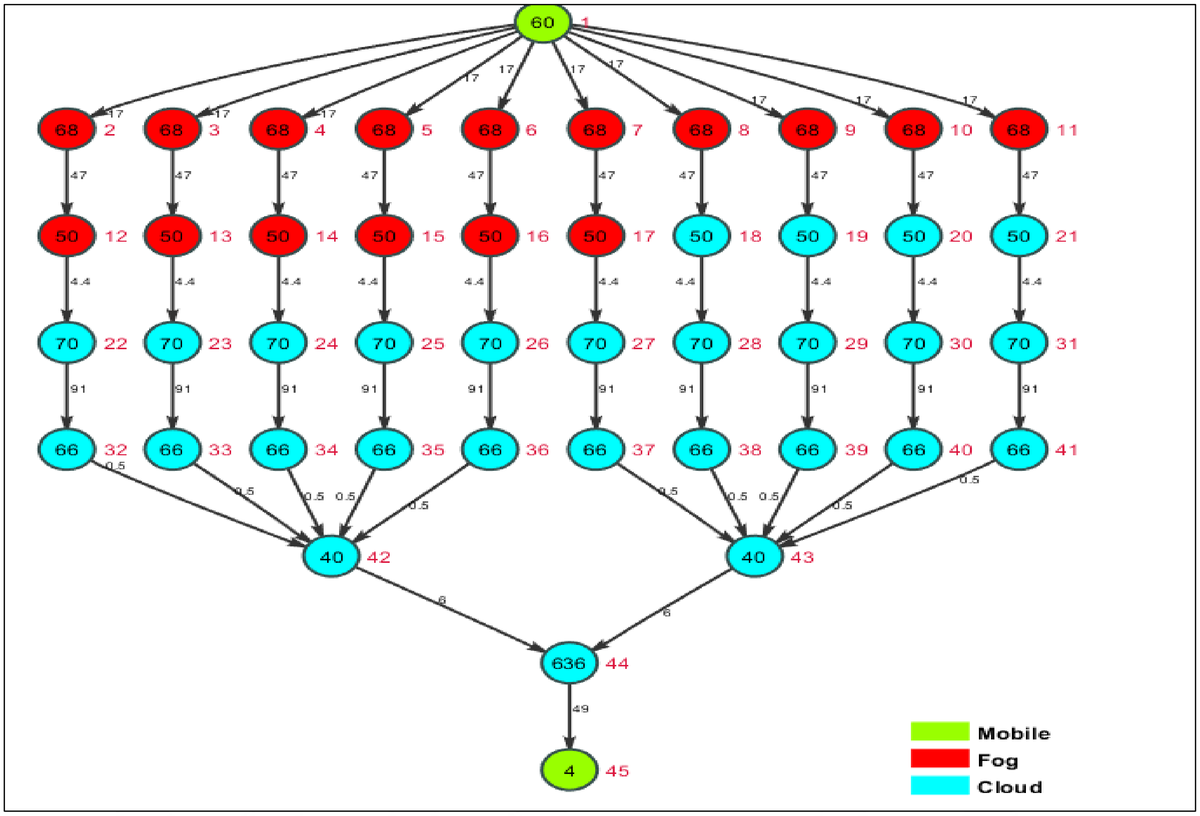

Figure 4.2: Modeling the Network Architecture

4.3 OPTIMIZATION STRATEGIES

This section describes optimization strategies in response to autonomic events in the fog computing environment. In this context, two types of events are defined in the scope of this work. The first concerns the initial request of a workload that needs to be distributed among the nodes of the network to fulfill requests from IoT sensors located at the edge of the network. The second case happens when the application module needs to be migrated to another node to meet some defined QoS objective, for example, when a node is underutilized in relation to a load limit, eg. 10%. In other words, the first objective concerns where the services will be allocated and in a second moment, "when" to migrate the service considering a better execution.

4.3.1 Service Allocation

The objective of this technique is to determine the mapping of IoT applications between network nodes according to pre-established objectives to define where the task can be performed, and which resources will be allocated for the best execution of the tasks. Figure 10 illustrates an example of allocating services between nodes in the fog environment. This example is composed of an application distributed among three modules (a1, a2 and a3), where the sensor connected to the Edge-1.1 node sends data to module a1, which sends data to modules a2 and a3, which finally sends them to an actuator connected to the Edge-1.2 node. A strategy that prioritizes the location of modules between nodes with lower latency and optimization objectives can be adapted to meet demand and bring benefits to different types of applications.

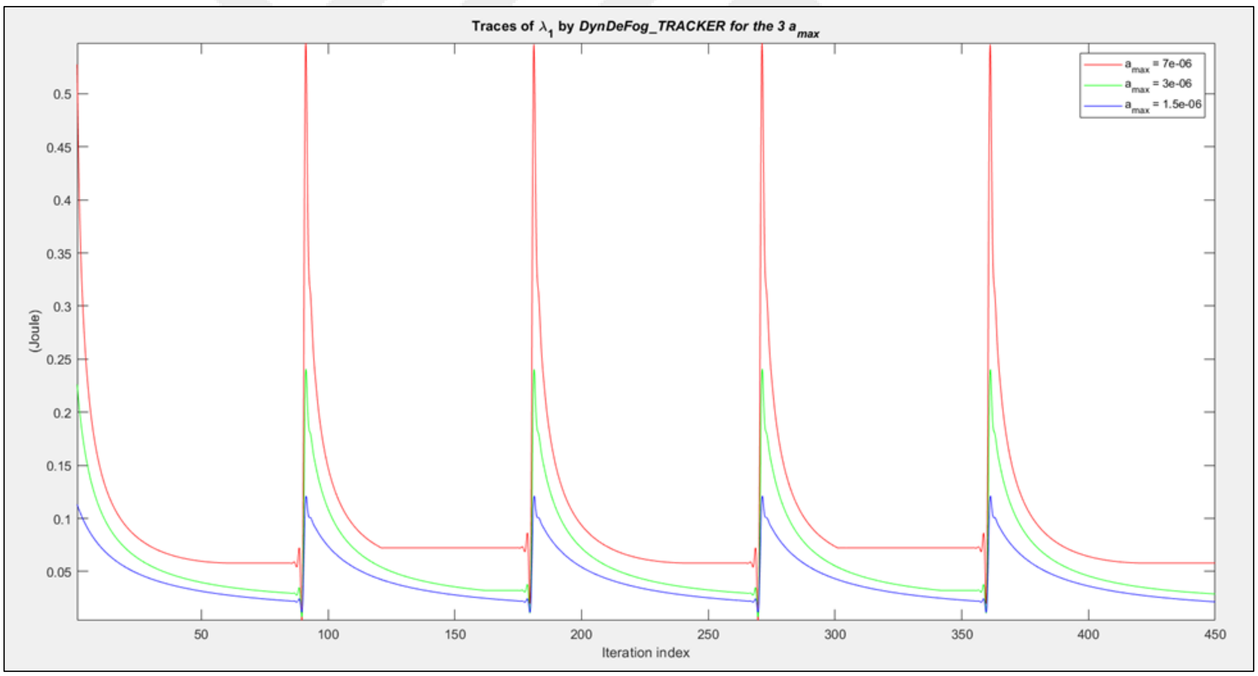

Figure 4.3: Service Allocation the Proposed System

Next, we describe two strategies with different types of algorithms, the first (Fist-Fit) that uses a greedy strategy and the second (Genetic Algorithm), with an evolutionary strategy.

A. First Fit: it is a greedy heuristic algorithm that provides a quick solution to the mapping, but not necessarily efficient, where each application module is placed in the first position that can accommodate it in the list of available nodes.

Algorithm 1 receives as input two lists, the first contains all available nodes in the fog environment and in the cloud and the second, a set of tasks to be performed. Then the lists are sorted in ascending order according to available capacity and application requirements respectively. The list of nodes is traversed starting from the lowest-capacity node in order to optimize the use of environment resources. The inner loop traverses the task list and calls a method that checks if the node is able to handle the task according to available resources and if so, includes the pair in the mapping list. Finally, the algorithm returns the list with the mapping.

B. Genetic Algorithm: This algorithm was inspired by Charles Darwin's Theory of Evolution of Species, which is based on Darwin's theory of reproduction by selection. The genetic algorithm (GA) is based on Darwin's theory of survival of the fittest. Conforming to Darwin's Theory, characteristics are passed down from parent to child, and individuals who are more adapted to their environment have a better chance of surviving and passing their characteristics on to future generations. It is made up of chromosomes (or individuals) that have a set of genes with the same length as one another that constitutes the initial population of genetic algorithms (the starting population). Everyone is assigned a score based on how close they are to achieving the best solution. On a population-wide scale, several genetic operators are applied, including mutations, selections, and cross-over operators.

Algorithm 2 represents the pseudocode described in this work where, t represents the current time, d represents the time determined to finish the execution of the algorithm and P represents the population. Chromosome representation: is a vector that represents an allocation mapping plan for services. The total number of services for a given task represents the length of the chromosome. Each gene on the chromosome represents an application module to be allocated and an index that references the node that will meet the demand.

C. Fitness Function: To determine whether candidates provide the most value for money, the fitness function is a vital statistic to consider. Fitness function values are widely used to determine how close a solution is to being found to an ideal solution, and they can be used in this manner. The allocation of resources can be decreased or maximized by identifying the fewest or most significant points for any function, which is how optimization can be regarded. To reduce latency for a given set of nodes in a solution, it is possible to adapt the objective function to other optimization targets, such as lowering the utilization of device resources or the energy consumption in fog computing, by modifying the objective function.

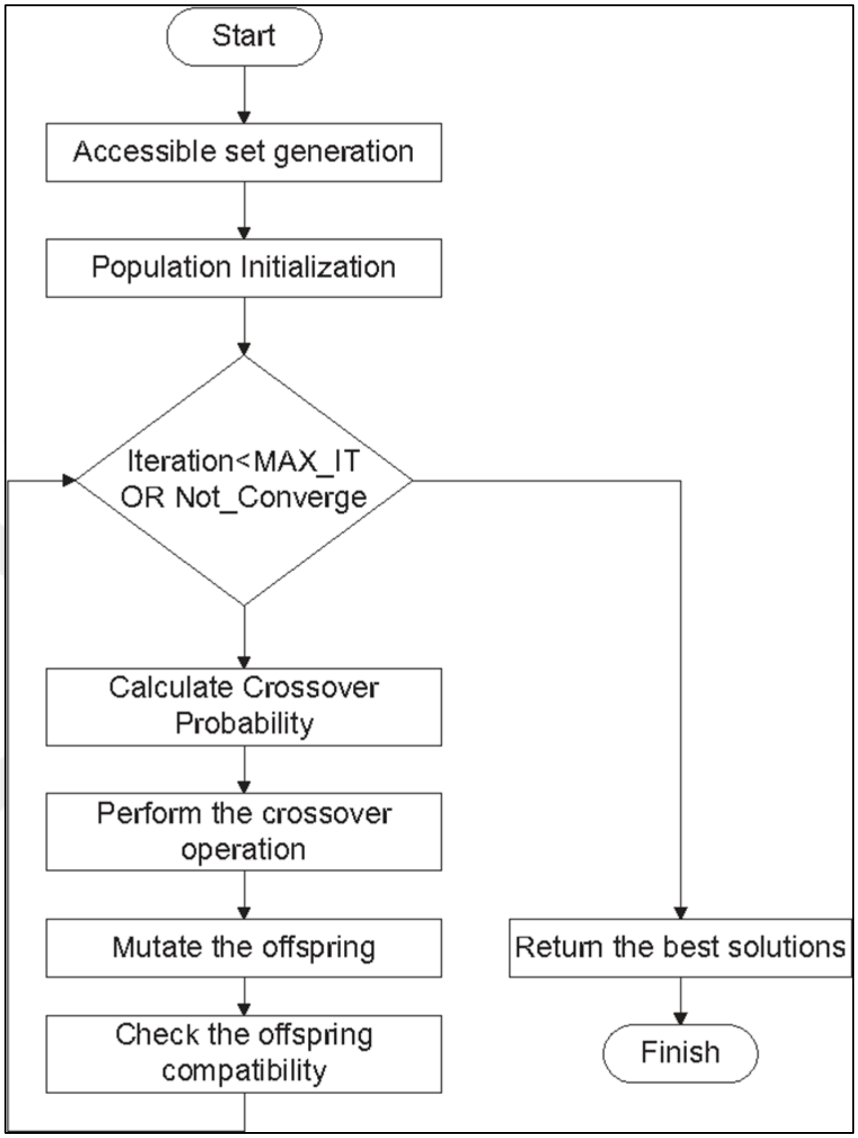

Figure 4.4: GA Algorithm Scheduling [50]

4.4 SERVICES MIGRATION AND HIGH AVAILABILITY

The migration strategy includes determining when to shift work to a new host and establishing migration procedures, among other things. Monitoring the environment generates autonomic events, the purpose of which is to decide which fog node, in terms of latency, is the most suitable for carrying out the activity in question without violating the QoS limitations. Autonomic events are generated by the environment monitoring system. The threshold values for Quality of

Service (QoS) for several types of Internets of Things applications are depicted in Figure 4.5. The delay, jitter, packet loss rate, and bandwidth of different domains are all represented by the numbers in this table.

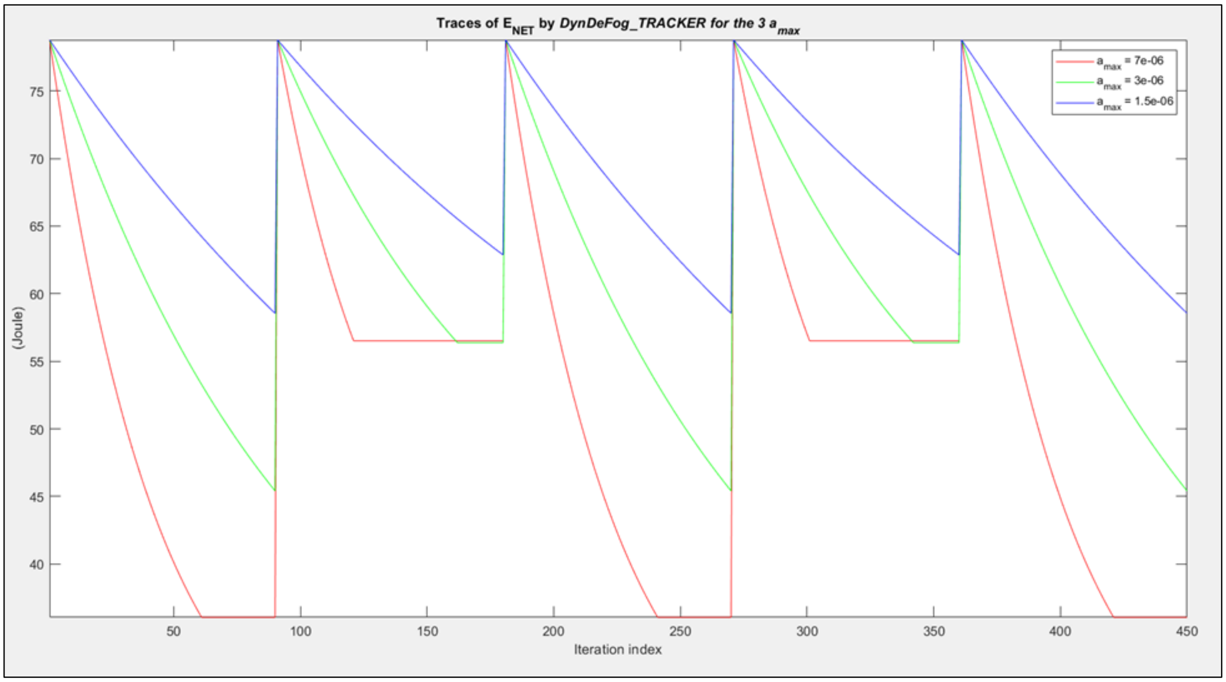

Figure 4.5: (QoS) For Several Types of Internets

During task migration in fog computing, a mechanism is required to restore a copy of the current state of the service to the destination node, which is essential to ensure that the service remains operational. A utility called CRIU (Checkpoint/Restore in User space) is used to preserve service quality while enforcing resource allocation regulations on Linux systems. CRIU (Checkpoint/Restore in User space) is a tool that implements checkpoint and restore capabilities on Linux systems. A sequence of measures is taken to attain high availability.

While the service is being supplied to customers, a sequence of actions is carried out in order to move the specified modules between two nodes in the fog environment (the "source" and "destination") as part of the migration algorithm's flow. During this time, the service is available to customers. An instruction from the control node to a target node instructs it to contact an image repository in the cloud and download a file containing the module's file. The source node then generates a checkpoint for an application module that has been specified.

The current state of an application can be represented by a checkpoint, which is then restored to the source node at the exact moment the checkpoint was made, as shown in Figure 4.6 Transferring checkpoint data between the source and destination nodes is accomplished through the usage of Rsync, a file synchronization tool. Following the successful completion of Rsync, the target node will be in possession of the module and the checkpoint will be restored. From the control node to the source node, a command is issued instructing it to terminate and delete the application module on which it is running.

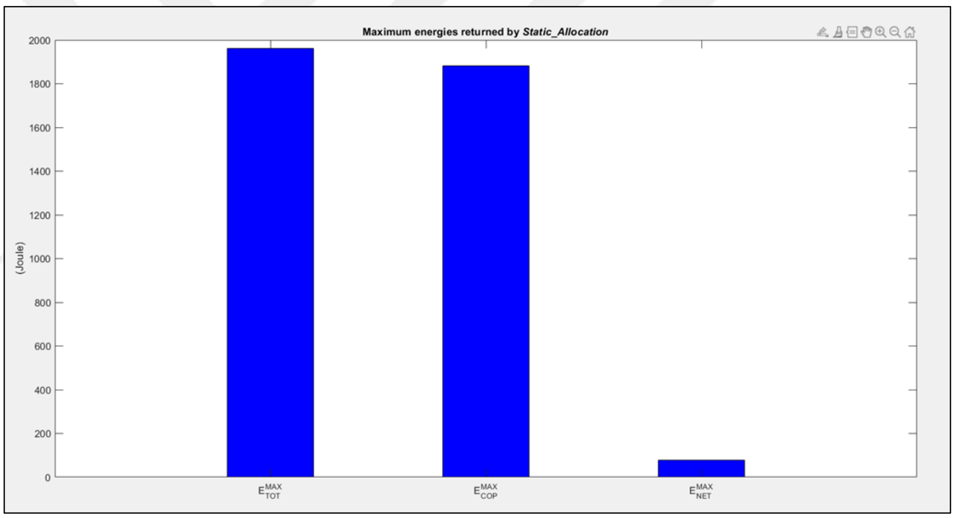

Figure 4.6: Transferring Checkpoint Data Usage of Rsync

4.5 EXPERIMENTS AND EVALUATION

It has been demonstrated that fog computing can reduce latency and resource consumption for Internet of Things (IoT) applications, as explored in more detail in Section 4 of this chapter.

4.5.1 Evaluation Methodology

It is necessary to assume that a computer model is identical to the actual manufacturing environment to perform prototyping. Because of the amount of equipment required to create a realistic environment, it may be difficult to complete the work and the experiment situation may become more complicated as a result. It is possible to add more and more scenarios to this environment using simulation and emulation tools, which makes it easier to construct.

The Internet of Things (IoT) Environment in Virtual Reality The ability to emulate fog devices using Docker containers is made possible by an emulator that was created using open-source tools. It is also possible to define the bandwidth and latency of devices and resources such as CPUs, RAM, and hard drives with a network. The ability to validate experiments with real-world applications, as well as the flexibility with which they could be implemented, were important considerations in the selection of this technology.

A total of two Fog Control Nodes/Gateways are considered, each of which is connected to four edge devices, namely four Raspberry Pi 2B and four Raspberry Pi 3B, which are in turn connected to sensors and actuators of the same type. The average number of iterations per second for each device, as determined by the Coremark tool, is used to calculate the CPU requirements for the emulator.

To complete this experiment, a total of 193 core marks are required, which is equal to two Centos 7 virtual computers with 16 cores each, 32GB of virtualized memory, and the Ubuntu 16.04 operating system. It is possible to simplify communication between containers and nodes that are not part of the virtual environment by using Docker's overlay drivers, which provide an interface for managing data transfer. Furthermore, network properties such as latency and bandwidth, as well as the computing characteristics of nodes such as CPU, memory, and storage, can be determined. whereas the edge devices (Smartphones, tablets, and general-purpose PCs) are connected to the fog environment via two independent private networks (Edges) that are organized into sub-workgroups, Fog-1 and Fog-2 are connected to each other through a public network, and the edge devices are connected to the fog environment through a public network. depicts the virtual environment as well as the characteristics of the network that exists within it During the experimentation process, the projected network characteristics for the public and private networks setup will allow for real-time data collection due to the nature of the networks. tools, which are discussed in greater detail in the following subsections.

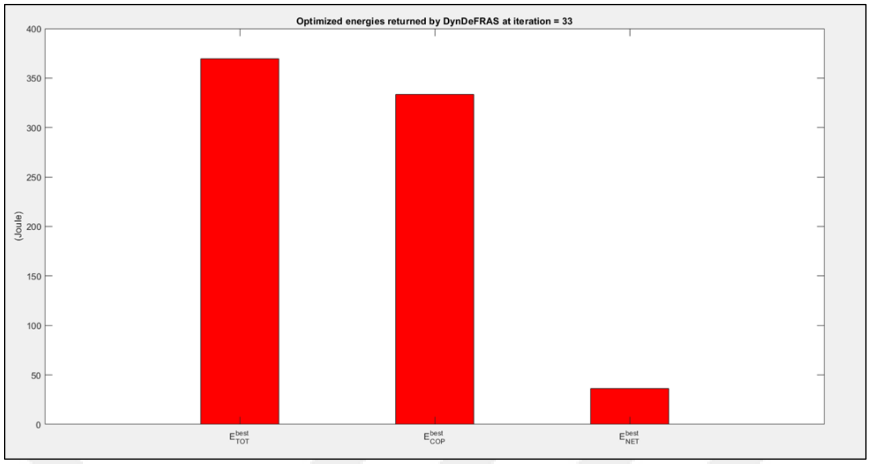

Figure 4.7: Fog-1

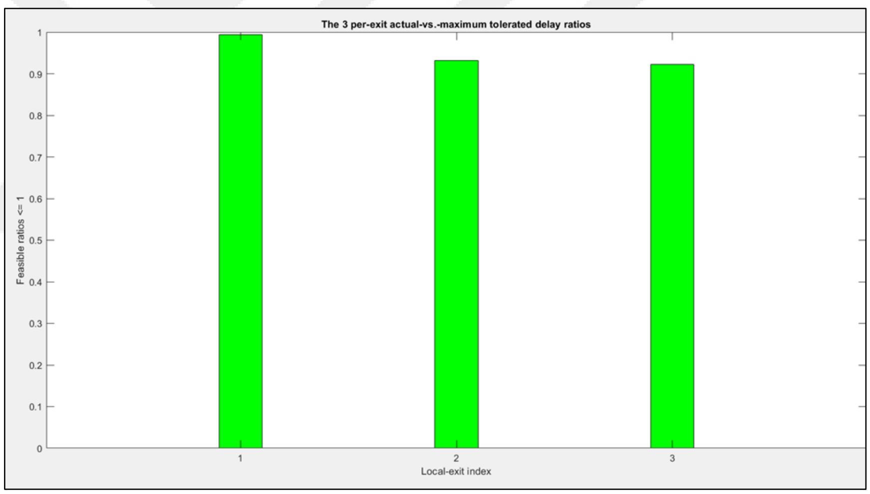

Figure 4.8: Fog-2

4.6 RNN RESULTS AND COMPARISON

The search terms "fog computing" and "machine learning" were used to locate journal articles and conference proceedings using Google Scholar. In our search, we did not include any studies published in 2017 or later. The initial selection was made based on the websites' provided abstracts and keywords. The retrieved studies were then classified according to their relevance. The second round of the selection process was conducted after the primary contributions and conclusions of the studies had been read. It was determined that only studies focusing specifically on the application of machine learning strategies in fog computing environments would be included. We identified the problems, objectives, and methodologies of the 19 completed studies by analyzing and reviewing the relevant literature. Given how ML was incorporated into FC, it was determined that the extracted data would be divided into three distinct sections. In chapter 6, a summary of the work performed in each of the reviewed studies was presented, and the summaries of these studies were analyzed and compared.

4.6.1 Recurrent Neural Network Rnn

The following are the most essential and fundamental terms used to describe deep learning: There are numerous types of software that fall under the headings of unsupervised machine learning, multiple layers of learning, and artificial intelligence. These classifications describe software that employs neural networks, pattern recognition, and other comparable techniques [32] Comparable to how the human brain organizes information and finds patterns to make better decisions [33] It employs both unsupervised and supervised methods for pattern recognition and classification tasks in order to learn multi-level features in hierarchical architectures and representations. Technological advancements in sensor networks and networking have made it possible to analyze vast quantities of data [35] DL is currently one of the most popular research avenues in the field of machine learning, with successes in a variety of fields. This is due to the fact that DL can be applied to a diverse range of problems [38] Due to the fact that it is the most general tool for modeling a problem, DL is capable of solving challenging issues [39] Not only is it capable of producing useful results in the absence of other methods, but it can also create extremely accurate models and halve the time required to develop a good concept. When other methods have failed, it is able to accomplish each of these. To put DL models into practice, however, enormous amounts of computing power are required [40] Today's Internet of Things (IoT) and fog networks require DL algorithms, which aid in the identification of meaningful trends and data, in order to function properly. Traditional deep learning algorithms utilize four types of neural networks: convolutional neural networks (CNN), deep neural networks (DNN), long short-term memory networks (LSTM), and recurrent neural networks (RNN):

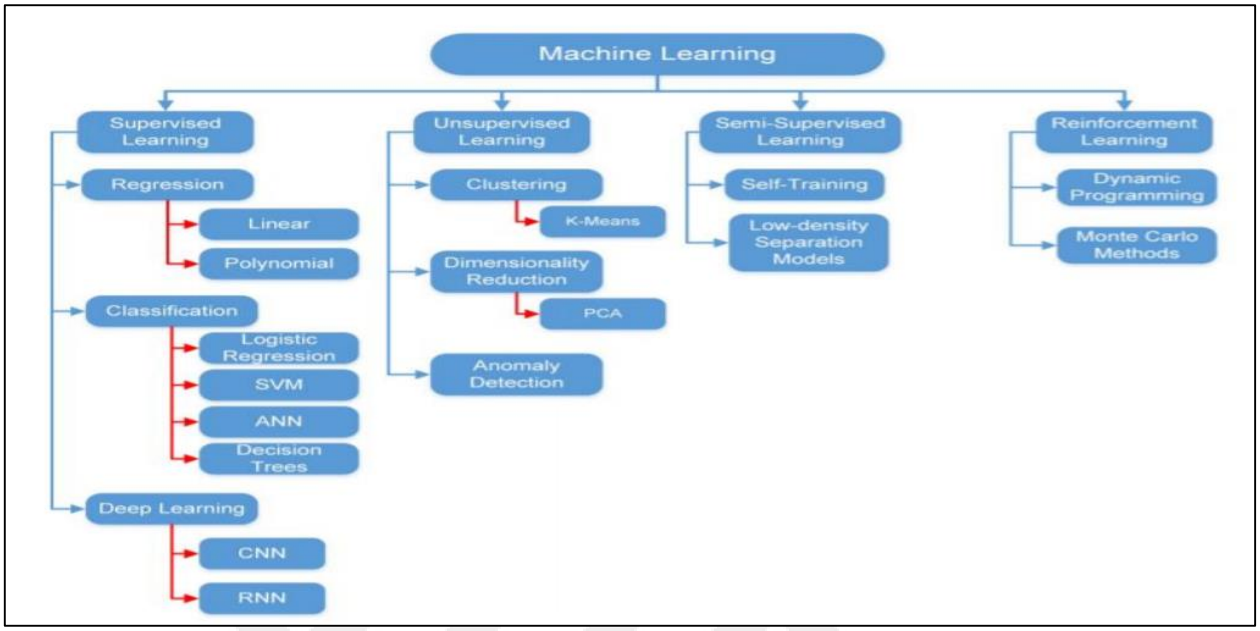

Figure 4.9: RNN Role in ML Algorithms

4.6.2 Rnn Results and Comparison with Dyndefras

Concerns about inaccurate latency, user mobility, and location awareness continue to exist despite the growing popularity of cloud computing. These issues are a direct result of cloud computing's characteristics. In lieu of the conventional method of attempting to distribute resources across the entire core network, DynDefRAS. addresses these issues by delivering elastic infrastructure and services to end users located at the network's edge. DynDefRAS systems are able to utilize data gathered from networking devices throughout their operations in order to address the challenges posed by Internet of Things applications regarding energy efficiency and delay. The DynDefRAS DL framework has evolved into a potent end-user application and a heightened service that provides in-depth analytics and smarter responses to challenges. Even though it is too early to tell whether FC intelligence will be useful in the real world, our research indicates that it has considerable potential. This review clearly demonstrates to end-users that the use of DL algorithms or their incorporation into FC can improve fog performance and efficiency, as well as provide a variety of services, such as the protection of resources and the management of traffic forecasts, as well as a reduction in latency and energy consumption.

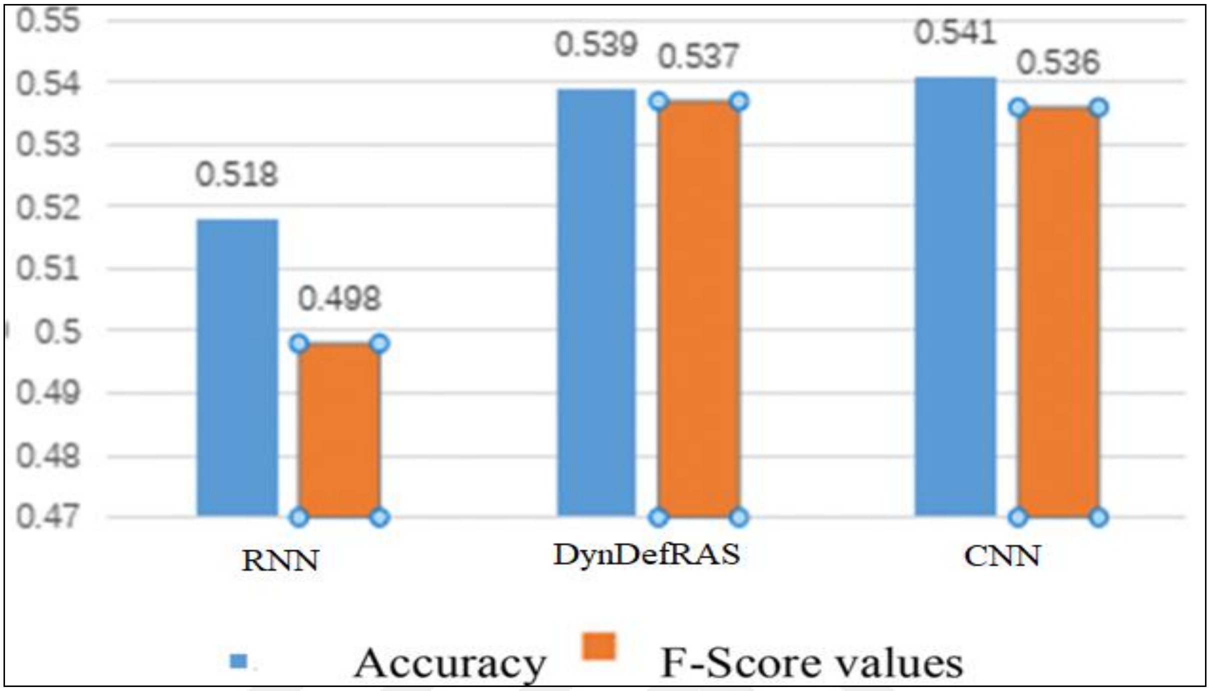

Figure 4.10: RNN Accuracy and Loss Compared to Other Algorithms

5. COCNLUSION

This chapter presents the main conclusions related to the work developed, as well as the publications carried out and future work.

5.1 COCNLUSION

As an alternative to traditional computing for latency-sensitive applications, this master's thesis presents fog computing as a viable option. The paper discusses fog computing, including its advantages and disadvantages, as well as its theoretical basis and related research findings. About the issue of allocating infrastructure resources, we offer an autonomic architecture model as an alternative to the options now available. Following the proposed paradigm, we present a technique for relocating services between fog nodes that employs Docker and CRIU (Checkpoint Live Restore) containers to assist in the implementation of optimization strategies. This method, depending on the quality-of-service requirements of the various types of applications, can assist in achieving a range of objectives, such as reducing latency or conserving energy. Testing was carried out in a large-scale Internet of Things virtual environment, which was used for the testing. In preliminary data, migration timings averaged 9.88 seconds, with a downtime of less than 2 seconds on average per migration. These findings are encouraging because they demonstrate that high availability and minimal downtime may be maintained during a service relocation project. Fog Computing was found to benefit from the migration of containers in order to run a variety of Internet of Things applications during the tests. We now have substantial evidence to support our hypothesis that the objectives of the optimization technique may be changed to take advantage of low latency and high bandwidth while still ensuring great quality of service for IoT applications, which was previously unfounded. In light of the growing popularity of IoT and latency-sensitive applications, we predict that our research will aid in the development of new strategies for efficient resource management, notably in the areas of latency, performance, and energy efficiency, among other things. To conclude things up, we'd like to provide some preliminary suggestions for further research.

5.2 FUTURE WORK

This work was designed and developed in order to maximize the utilization of resources in a fog computing environment while also providing a high level of service to latency-sensitive applications that require quick response times. Although the present research efforts mentioned in this work are encouraging, there are still several challenges to overcome. Authentication and security mechanisms can be established, enabling for the ongoing development of this work. Because its components are generally located close to devices and sensors, fog computing systems pose a substantial threat to security in a variety of ways. Despite the fact that it is theoretically possible to employ cloud-based techniques in a fog environment, research in this field is still in the early stages. Fog computing security has been investigated in the literature in terms of user authentication, privacy, safe data interchange, denial-of-service (DoS) attack, and other factors. Additionally, other concerns, such as security and privacy, are addressed throughout the duration of the project.

Reduce the amount of time it takes to migrate a service by developing new and inventive approaches. On the Internet of Things, one of the most difficult problems is determining how to provide and maintain computing and storage resources near the "things," as well as how to adjust the switching context, manage resource allocation, and migrate state to a new location in the event of a rapid location shift. Container-based virtualization techniques have the potential to enable quick service migrations by just transferring the state of the containers in the virtualization environment. A representative technology, such as Docker, is provided as an example; nonetheless, defining methods for moving services between fog devices and the cloud remains a difficult task. It is also necessary to provide ways for shortening the overall migration time.

Evaluate and contrast various virtualization techniques and approaches. Another challenge to overcome is the service virtualization model provided by fog computing nodes, which rely on Software Defined Networking to perform their functions (SDN). Fundamentally, software-defined networking (SDN) is an architecture that enables network control directly programmable and allows infrastructure to be abstracted into apps and provided on a variety of servers throughout a network. SDN is essential in supporting Network Functions Virtualization (NFV), which is one of the most important components of the technology. For example, network function virtualization can be used to virtualize firewalls, intrusion detection systems, domain name systems, and other network-related systems and services (NFV).

Reduce power consumption in a real-world situation by utilizing DVFS techniques. Fog nodes must deal with an overwhelming number of service requests coming in from Internet of Things devices. The use of fog nodes in the environment as needed is a simple method that may be implemented quickly. As a result, the total energy consumption of the system will increase because of the increased activity of all of the fog nodes in the system. Although it is well-established in cloud computing environments, the application of DVFS in fog computing is an area that is currently being investigated in depth.

Make advantage of evolutionary algorithms to compare various resource allocation strategies. In most resource optimization studies, just one variable is taken into consideration. In contrast, many real-world challenges demand the consideration of several different elements. As a result, current research has concentrated on multi-objective instances to better understand their dynamics.

Evaluate the performance of the fog environment by utilizing the benchmark tools. It is vital to establish criteria for analyzing IoT settings through benchmark tools, especially when examining management practices, to ensure that they are comparable. We're not aware of any studies in this topic that are particularly noteworthy. As an example, micro-benchmarks for distributed stream processing systems for Internet of Things (IoT) applications can be used to evaluate distributed stream processing systems.

REFERENCES

[1]. Sarkar, S., Misra, S., “Theoretical modelling of fog computing: a green computing paradigm to support iot applications”, IET Networks 5(2) (2016) 23–29

[2]. Peter, Nisha, "Fog computing and its real time applications." International Journal of Emerging Technology and Advanced Engineering (IJETAE) 5, no. 6 (2015): 266-269.

[3]. Wang, Yifan, TetsutaroUehara, and Ryoichi Sasaki, "Fog computing: issues and challenges in security and forensics." In Computer Software and Applications Conference (COMPSAC), 2015 IEEE 39th Annual, vol. 3, pp. 53-59. IEEE, 2015.

[4]. Pearson, Siani, and AzzedineBenameur. "Privacy, security and trust issues arising from cloud computing." In Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, pp. 693-702. IEEE, 2010.

[5]. Dlamini, M. T., et al. "Security of Cloud Computing: Seeing Through the Fog." computing 2 (2011): 3.

[6]. Stolfo, Salvatore J., Malek Ben Salem, and Angelos D. Keromytis. "Fog computing: Mitigating insider data theft attacks in the cloud." In Security and Privacy Workshops (SPW), 2012 IEEE Symposium on, pp. 125-128. IEEE, 2012.

[7]. Madsen, Henrik, Bernard Burtschy, G. Albeanu, and F. L. Popentiu-Vladicescu. "Reliability in the utility computing era: Towards reliable fog computing." In Systems, Signals and Image Processing (IWSSIP), 2013 20th International Conference on, pp. 43-46. IEEE, 2013.

[8]. Hong, Kirak, David Lillethun, UmakishoreRamachandran, BeateOttenwälder, and Boris Koldehofe. "Mobile fog: A programming model for large-scale applications on the internet of things." In Proceedings of the second ACM SIGCOMM workshop on Mobile cloud computing, pp. 15-20. ACM, 2013.

[9]. Chou, Te-Shun. "Security threats on cloud computing vulnerabilities." International Journal of Computer Science & Information Technology 5, no. 3 (2013): 79.

[10]. Modi, Chirag, Dhiren Patel, BhaveshBorisaniya, Avi Patel, and MuttukrishnanRajarajan. "A survey on security issues and solutions at different layers of Cloud computing." The Journal of Supercomputing 63, no. 2 (2013): 561-592.

[11]. Stojmenovic, Ivan, and Sheng Wen. "The fog computing paradigm: Scenarios and security issues." In Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on, pp. 1-8. IEEE, 2014.

[12]. Shankarwar, Mahesh U., and Ambika V. Pawar. "Security and Privacy in Cloud Computing: A Survey." In FICTA (2), pp. 1-11. 2014.

[13]. Firdhous, Mohamed, Osman Ghazali, and Suhaidi Hassan. "Fog computing: Will it be the future of cloud computing?." In Third International Conference on Informatics & Applications, Kuala Terengganu, Malaysia, pp. 8-15. 2014.

[14]. Loke, Seng W. "The internet of flying-things: Opportunities and challenges with airborne fog computing and mobile cloud in the clouds." arXiv preprint arXiv:1507.04492 (2015).

[15]. Bitam, Salim, AbdelhamidMellouk, and SheraliZeadally. "VANET-cloud: a generic cloud computing model for vehicular Ad Hoc networks." IEEE Wireless Communications 22, no. 1 (2015): 96-102.

[16]. Roman, Rodrigo, Javier Lopez, and Masahiro Mambo. "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges." Future Generation Computer Systems (2016).

[16] Noth, A. Design of Solar Powered Airplanes for Continuous Flight. Ph.D. Thesis, ETH Zurich, Zürich, Switzerland, 2008. Available online: http://http://www.skysailor.ethz.ch/docs/Conceptual_Design_of_ Solar_Powered_Airplanes_for_continuous_flight.pdf (accessed on 20 August 2018).

[17] Bruscoli, S. Airfoil optimization for a solar powered aircraft. Master's Thesis, Universita Degli Studi di Pisa, Pisa, Italy, 2011.

[18] Noth, A.; Engel, W.; Siegwart, R. Flying Solo and Solar to Mars. IEEE Robot. Autom. Mag. 2006, 13, 44–52. [CrossRef]

[19] Böswald, M.; Govers, Y.; Vollan, A.; Basien, M. Solar Impulse—How to validate the numerical model of a superlight aircraft with A340 dimensions. In Proceedings of the International Conference on Noise & Vibration Engineering, Leuven, Belgium, 20–22 October 2010.

[20] Shiau, J.-K.; Wei, Y.-C.; Lee, M.-Y. Fuzzy controller for a voltage-regulated solarpowered mppt system for hybrid power system applications. Energies 2015, 8, 3292–3312. [CrossRef]

[21] Zhang, H.; Cheng, S. A New MPPT Algorithm Based on ANN in Solar PV Systems. In Lecture Notes in Electrical Engineering; Springer: Berlin, Germany, 2011.

[22] Commercial Radio-Controlled Aircraft Arrow. Available online: http://www2.arnes.si/~cetim6/plans/ arrowlayout.pdf (accessed on 20 August 2018).

[23] Rakesh, C.; Sunil Kumar, N. Design and Analysis of Solar Powered RC Aircraft. IJES 2015, 4, 29–41. 25.

[24] Buchireddy, K.R.; Aneesh, P.; Bhanu, K.; Natarjan, M. Design Analysis of Solar-Powered Unmanned Aerial Vehicle. J. Aerosp. Technol. Manag. 2016, 8, 397–407. 26.

[25] Espinoza-Fraire, T.; Dzul, A.; Cortes-Martinez, F.; Giernacki, W. Real-time Implementation and Flight Tests using Linear and Nonlinear Controllers for Fixed-wing Miniature Aerial Vehicle (MAV). Int. J. Control Autom. Syst. 2018, 16, 392–396. [CrossRef]

[26] Zhilenkov, A.A.; Epifantsev, I.R. System of autonomous navigation of the drone in difficult conditions of the forest trails. In Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering, ElConRus 2018, Moscow and St. Petersburg, Russia, 29 January–1 February 2018; Volume 2018, pp. 1036–1039. [CrossRef]

[27] Lee, S.; Shim, T.; Kim, S.; Park, J.; Hong, K.; Bang, H. Vision-Based Autonomous Landing of a Multi-Copter Unmanned Aerial Vehicle using Reinforcement Learning. In Proceedings of the 2018 International Conference on Unmanned Aircraft Systems, ICUAS 2018, Dallas, TX, USA, 12–15 June 2018; pp. 108–114. [CrossRef]

[28] Dionisio-Ortega, S.; Rojas-Perez, L.O.; Martinez-Carranza, J.; Cruz-Vega, I. A deep learning approach towards autonomous flight in forest environments. In Proceedings of the 2018

International Conference on Electronics, Communications and Computers (CONIELECOMP), Cholula, Mexico, 21–23 February 2018; pp. 139–144. [CrossRef

[29] Gandhi, D.; Pinto, L.; Gupta, A. Learning to fly by crashing. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; Volume 2017, pp. 3948–3955. [CrossRef]

[30] Falanga, D.; Mueggler, E.; Faessler, M.; Scaramuzza, D. Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision. In Proceedings of the IEEE International Conference on Robotics and Automation, Singapore, 29 May–3 June 2017. [CrossRef]

[31] McGuire, K.; de Croon, G.; De Wagter, C.; Tuyls, K.; Kappen, H. Efficient Optical Flow and Stereo Vision for Velocity Estimation and Obstacle Avoidance on an Autonomous Pocket Drone. IEEE Robot. Autom. Lett. 2017, 2, 1070–1076. [CrossRef]

[32] Zeggada, A.; Melgani, F.; Bazi, Y. A Deep Learning Approach to UAV Image Multilabeling. IEEE Geosci. Remote Sens. Lett. 2017, 14, 694–698. [CrossRef]

[33] Zhao, Y.; Zheng, Z.; Zhang, X.; Liu, Y. Q learning algorithm based UAV path learning and obstacle avoidence approach. In Proceedings of the Chinese Control Conference, CCC, Dalian, China, 26–28 July 2017; pp. 3397–3402. [CrossRef]

[34] Von Stumberg, L.; Usenko, V.; Engel, J.; Stuckler, J.; Cremers, D. From monocular SLAM to autonomous drone exploration. In Proceedings of the 2017 European Conference on Mobile Robots, ECMR 2017, Paris, France, 6–8 September 2017. [CrossRef]

[35] Moriarty, P.; Sheehy, R.; Doody, P. Neural networks to aid the autonomous landing of a UAV on a ship. In Proceedings of the 2017 28th Irish Signals and Systems Conference, ISSC 2017, Killarney, Ireland, 20–21 June 2017; pp. 6–9. [CrossRef]

[36] Giusti, A.; Guzzi, J.; Ciresan, D.C.; He, F.L.; Rodriguez, J.P.; Fontana, F.; Faessler, M.; Forster, C.; Schmidhuber, J.; Caro, G.D.; et al. A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots. IEEE Robot. Autom. Lett. 2016, 1, 661–667. [CrossRef]

[37] Zhang, T.; Kahn, G.; Levine, S.; Abbeel, P. Learning deep control policies for autonomous aerial vehicles with MPC-guided policy search. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; Volume 2016. [CrossRef]

[38] Daftry, S.; Zeng, S.; Khan, A.; Dey, D.; Melik-Barkhudarov, N.; Bagnell, J.A.; Hebert, M. Robust Monocular Flight in Cluttered Outdoor Environments. arXiv 2016, arXiv:1604.04779.

[39] Antonio-Toledo, M.E.; Sanchez, E.N.; Alanis, A.Y. Robust neural decentralized control for a quadrotor UAV. In Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada, 24–29 July 2016; Volume 2016, pp. 714–719. [CrossRef]

[40] Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015—Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015; pp. 1–14.

[41] Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. [CrossRef]

[42] Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. arXiv 2018, arXiv:1804.02767.

[43] He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef

[44] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. Commun. ACM 2017, 60, 84–90. [CrossRef]

[45] Rodriguez-Ramos, A.; Sampedro, C.; Bavle, H.; Moreno, I.G.; Campoy, P. A Deep Reinforcement Learning Technique for Vision-Based Autonomous Multirotor Landing on a Moving Platform. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1010–1017. [CrossRef]

[46] Carrio, A.; Vemprala, S.; Ripoll, A.; Saripalli, S.; Campoy, P. Drone Detection Using Depth Maps. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1034–1037. [CrossRef]

[47] Irfan, M.; Dalai, S.; Kishore, K.; Singh, S.; Akbar, S.A. Vision-based Guidance and Navigation for Autonomous MAV in Indoor Environment. In Proceedings of the 2020 11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020, Kharagpur, India, 1–3 July 2020. [CrossRef]

[48] Roldan, I.; Del-Blanco, C.R.; De Quevedo, D.; Urzaiz, F.I.; Menoyo, J.G.; López, A.A.; Berjón, D.; Jaureguizar, F.; García, N. DopplerNet: A convolutional neural network for recognising targets in real scenarios using a persistent range-Doppler radar. IET Radar Sonar Navig. 2020, 14, 593–600. [CrossRef]

[49] Liao, Y.; Mohammadi, M.E.; Wood, R.L. Deep Learning Classification of 2D Orthomosaic Images and 3D Point Clouds for Post-Event Structural Damage Assessment. Drones 2020, 4, 24. [CrossRef]

[50] Wang, Y.; Wang, H.; Wen, J.; Lun, Y.; Wu, J. Obstacle Avoidance of UAV Based on Neural Networks and Interfered Fluid Dynamical System. In Proceedings of the 2020 3rd International Conference on Unmanned Systems (ICUS), Harbin, China, 27–28 November 2020; pp. 1066–1071. [CrossRef]

[51] Bozcan, I.; Kayacan, E. UAV-AdNet: Unsupervised Anomaly Detection using Deep Neural Networks for Aerial Surveillance. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), Harbin, China, 27–28 November 2020; pp. 1158–1164. [CrossRef]

[52] Messina, L.; Mazzaro, S.; Fiorilla, A.E.; Massa, A.; Matta, W. Industrial Implementation and Performance Evaluation of LSD-SLAM and Map Filtering Algorithms for Obstacles Avoidance in a Cooperative Fleet of Unmanned Aerial Vehicles. In Proceedings of the IRCE 2020—2020 3rd International Conference on Intelligent Robotics and Control Engineering, Oxford, UK, 10–12 August 2020; pp. 117–122. [CrossRef]

[53] Li, B.; Wu, J.; Tan, X.; Wang, B. ArUco Marker Detection under Occlusion Using Convolutional Neural Network. In Proceedings of the 2020 5th International Conference on Automation, Control and Robotics Engineering (CACRE), Dalian, China, 19–20 September 2020; Volume 8, pp. 706–711. [CrossRef]

[54] Tan, J.; Zhao, H. UAV Localization with Multipath Fingerprints and Machine Learning in Urban NLOS Scenario. In Proceedings of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Chengdu, China, 11–14 December 2020; pp. 1494–1499. [CrossRef]