

Power Efficient Framework for Resource Utilization in Fog Computing

A Thesis

Submitted for the award of the degree of

Doctor of Philosophy

in

Computer Science & Engineering Department

Submitted by

Abhishek Jain

(Reg No: 951703003)

Under the Guidance of

Dr. Amit Kumar Bhardwaj

Associate Professor



Thapar Institute of Engineering and Technology, Patiala

INDIA-147004

June, 2023

Certificate

I hereby declare that the work being presented in this thesis "**Power Efficient framework for Resource Utilization in Fog Computing**", in fulfillment of the requirements for the award of degree of **DOCTOR OF PHILOSOPHY** submitted in Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my work carried out under the supervision of Dr. Amit Kumar Bhardwaj, Associate Professor, LMTSM, Thapar Institute of Engineering & Technology and refers other researcher works which are duly listed in the reference section. The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.



(Abhishek Jain)

Registration No. 951703003

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge and belief.



(Dr. Amit Kumar Bhardwaj)

Associate Professor

LMTSM, Thapar Institute of Engineering & Technology

Supervisor

Acknowledgements

I would like to express my sincere gratitude to the Almighty for granting me the insight and determination to reach this significant milestone. I am forever thankful to God for the blessings that have accompanied me throughout this journey.

First and foremost, I extend my heartfelt gratitude to my supervisor, **Dr. Amit Kumar Bhardwaj**, for his invaluable guidance and encouragement at every stage of my PhD. program. This thesis would not have been possible without his unwavering support and belief in my abilities. During moments of difficulty, his constant motivation, vast knowledge, and scientific expertise illuminated my path. I am indebted to them for his insightful discussions and suggestions, which have shaped the direction of my research. His contribution to this thesis surpasses his roles as academic supervisor; he has also provided personal support without which I may not have completed this journey. I am truly grateful for having such an exceptional mentor. His unwavering support, encouragement, vast knowledge, and insightful perspectives has been a guiding light in my darkest times. I couldn't have asked for a better mentor for this journey. Their faith in me has always re-energized and rejuvenated my spirits. I am also grateful to **Dr. Maninder Singh**, Dean of Academic Affairs, and **Dr. Seema Bawa**, Dean Outreach, for being the guiding light during the most challenging moments.

I extend my utmost gratitude to **Dr. Shalini Batra**, Head of the Department of Computer Science and Engineering, for her continuous encouragement and motivation. I would also like to express my appreciation to **Dr. Anju Bala**, **Dr. Singara Singh Kasana**, and **Dr. Vinay Kumar** for their participation as members of my doctoral committee and their valuable advice during the preliminary stages of my research studies. Additionally, I would like to thank **Dr. Tejo Prakash**, Dean of Research and Sponsored Projects, for his academic support. **Dr. Padmakumar Nair**, Director of the Thapar Institute of Engineering and Technology, has played a crucial role in establishing a thriving research environment at the university by providing all the necessary resources and facilities.

I am grateful to the administrative and technical staff members of the Department of Computer Science and Engineering for their helpful guidance and support in their respective roles. I would also like to express my gratitude to **Dr. Vijay Prakash** and **Dr. Sahil Sharma** for the wonderful times we have shared and for making this research experience enjoyable and unforgettable.

The completion of this thesis would not have been possible without the encouragement and

support of my family. I would like to offer special recognition to my parents, who have always been an immense source of support and have inspired me to dream big.

My deepest gratitude goes to my wife, **Ms. Neha Jain**, whose unwavering support throughout the years is deeply cherished. She has instilled confidence in me and served as an inspiration in all aspects of life, enabling me to successfully complete this journey. I will forever remain indebted to her for her endless love and unwavering dedication, which continue to sustain me.

This work is dedicated to my son **Samyak Jain** as he has made me stronger, better and more fulfilled than I could have ever imagined. The day he was born, life became more charismatic and meaningful which encouraged me to achieve this goal.

I eagerly look forward to a life filled with joy and shared experiences.



Abhishek Jain

Abstract

Wireless Sensor Networks (WSNs) are essential for a number of applications, including industrial automation and environmental monitoring. Optimizing power consumption to increase the network's operational efficiency and lifespan is one of the main issues with WSNs. power optimisation techniques in WSNs are compared with an emphasis on routing protocols and energy-saving protocols.

Different approaches, including clustering algorithms, nature-inspired optimisation methods, and cross-layer optimisations, have been investigated to achieve power optimisation. The effectiveness of various algorithms, including ant lion optimisation (ALO), k-means clustering, and other conventional techniques, is investigated in this study in terms of lowering energy consumption and extending network lifetime.

The proposed study examines how these algorithms affect the various WSN layers, especially the end device layer and the fog layer. By contrasting the proposed algorithm with well-known methods like energy-efficient cross-layer sensing clustering, Distributed and Morphological Operation-based Data Collection Algorithm (DMOA), and trust-based secure routing, a thorough evaluation of the proposed algorithm is carried out.

The comparative study reveals the significant improvements made by the proposed power optimisation algorithm through extensive simulations and performance analysis. The outcomes show how effective it is at cutting energy use and improving overall network performance. The study also clarifies the trade-offs between different power optimisation strategies and their effects on network scalability, reliability, and latency.

This study adds to the growing body of information on power optimisation for WSNs and offers helpful tips for academics and industry professionals working in the area. The results emphasise the significance of choosing suitable power optimisation methods based on particular application requirements and network characteristics.

List of Figures

1.1	Fog Computing Architecture	01
1.2	Cloud & Fog Layer	02
1.3	Fog Computing Components	04
1.4	Fog Computing best practices	12
1.5	Integration of AI/ML	16
1.6	Modelling with ML Algorithms	18
1.7	Cluster formation in Wireless Network	22
3.1	Representation of Sensor Networks with varying node density	51
3.2	Representation of predicted optimized cluster members and cluster heads. Dashed lines show one hop communication between members and cluster heads	54
3.3	Examples of predicted Fog Nodes and Cluster Heads using FCH k-means++	54
3.4	Dead and fog node positions at different simulation times	56
3.5	Residual energy of nodes vs running time, Network Energy consumption over time for State of Art Methods	57
3.6	Network energy of nodes vs running time, Network Energy consumption over time for State of Art Methods	57
3.7	Network lifetime over Number of Sensor nodes for proposed algorithm compared with various State of Art Methods	58
3.8	Number of alive nodes vs running time. Proposed algorithm shows the Network stability as compared to state of Art methods	58
3.9	Number of Dead nodes vs running time. Proposed algorithm shows the Network stability as compared to state of Art methods	59
4.1	Architecture of proposed model	63
4.2	Communication Process	64
4.3	SOM clustering (a) SOM neighbor connections, (b) neighbor weights, (c) represent input weights of SOM clusters, (d) represent output weights of SOM cluster, (e) shows the number of cluster head are clustered with FOG cluster heads and (f) shows the final connected weights of cluster heads with FOG clustered heads	68

4.4	Clustering with various techniques(a) Clustering with SOM, (b) Clustering with k-means, (c) Clustering with Fuzzy and (d) Clustering with ensemble	69
4.5	Shows the Ensemble clustered FOG layer	70
4.6	Data Packet Representation	70
4.7	Nodes representation deployed in range with (a) 200 nodes, $\lambda=0.08$. (b) 400 nodes, $\lambda=0.16$ (c) 800 nodes, $\lambda=0.32$ (d) 1500 nodes, $\lambda=0.60$	71
4.8	Comparison plot of Network lifetime	73
4.9	Comparison histogram of alive nodes at various time stamps	73

List of Tables

1.1	Comparison of Cloud & Fog Computing	09
2.1	Comparison of Fog nodes	29-33
2.2	Comparison of Power based optimization	42
3.1	Proposed network simulation parameters	49
4.1	Notations and their meanings	63
4.2	Pseudo code for k-means++ algorithm	67
4.3	Parameters and Assumptions for Simulation	72
4.4	Shows total residual energy of nodes for simulation time	72-73

Contents

Certificate	i
Acknowledgements	ii
Abstract	iv
List of Figures	v
List of Tables	vii
1. Introduction.....	1
1.1 Fog Computing... ..	1
1.2 Basic components of fog computing.....	3
1.2.1 Physical and virtual nodes	3
1.2.2 Fog nodes.....	4
1.2.3 Monitoring Services	4
1.2.4 Data Processors	5
1.2.5 Resource Manager	5
1.2.6 Security Tools.....	5
1.2.7 Applications.....	5
1.3 Examples and use cases of fog computing.....	5
1.3.1 Smart homes	6
1.3.2 Smart cities.....	6
1.3.3 Video surveillance.....	6
1.3.4 Healthcare	7
1.3.5 Others.....	7
1.4 Need of fog computing over cloud computing	7
1.4.1 Reduced latency	7
1.4.2 Improved security	8
1.4.3 Better reliability	8
1.4.4 Reduced Bandwidth usage	8
1.4.5 Lower costs	8
1.5 Comparison of fog computing and cloud computing.....	8
1.6 Challenges.....	10
1.6.1 Structural issues	10
1.6.2 Service oriented challenges.....	10
1.6.3 Security aspects.....	11
1.7 Fog computing best practices to follow in 2023	11

1.7.1	Ensure provision for flexibility.....	12
1.7.2	Implement a fog console.....	12
1.7.3	Apply access control at the fog mode layer.....	12
1.7.4	Set up appropriate security tools and processes.....	13
1.7.5	Ensure small hardware and software footprint.....	13
1.7.6	Incorporate threat detection and mitigation at the fog level.....	13
1.7.7	Use relevant load balancing techniques	13
1.7.8	Choose storage options based on requirements	14
1.7.9	Consider energy efficiency	14
1.7.10	Design for uninterrupted fog services.....	14
1.8	Need of power efficient framework in fog computing.....	14
1.9	Integration of AI/ML	15
1.10	Need of power efficiency for wireless Sensor Networks.....	19
1.11	Research Motivation	24
1.12	Thesis organization	25
2.	Literature Survey	26
2.1	Background.....	26
2.2	Nodal collaboration.....	31
2.2.1	Cluster	31
2.2.2	Peer to Peer.....	31
2.2.3	Master-Slave.....	31
2.3	Resource/Service Provisioning Metrics	32
2.3.1	Time.....	32
2.3.2	Deadline.....	32
2.3.3	Data	32
2.3.4	Cost.....	33
2.3.5	Energy consumption and carbon footprints.....	33
2.4	Service level objectives.....	34
2.4.1	Latency Management.....	34
2.4.2	Cost Management	34
2.4.3	Network Management.....	34
2.4.4	Computation Management.....	35
2.4.5	Application Management.....	35
2.4.6	Data Management	36
2.4.7	Power Management.....	36
2.5	Security Concerns	40

2.5.1	Authentication.....	40
2.5.2	Privacy... ..	40
2.5.3	Encryption.....	40
2.5.4	DoS Attack.....	41
2.6	Research Gaps.....	43
2.7	Problem Statement.....	44
2.8	Objectives.....	45
2.9	Methodology... ..	46
2.10	Thesis Contribution.....	47
3.	Power efficient optimized clustering method (PEOCM) with intelligent fog computing for WSNs	48
3.1	Introduction.....	48
3.2	Material and methods.....	49
3.2.1	Evaluation parameters and assumptions	49
3.2.2	Energy calculations in data routing.....	49
3.2.3	Introduction to ALO.....	50
3.3	Proposed network model.....	52
3.3.1	Optimization in cluster formation.....	52
3.3.2	Introducing fog computing using k-means	54
3.4	Results and discussions.....	55
4.	Machine learning based power efficient optimized communication ensemble model with intelligent fog computing for WSNs	59
4.1	Introduction.....	59
4.2	Related works.....	60
4.3	Material and methods.....	61
4.3.1	Energy consumptions at various levels in WSN networks	63
4.3.2	Proposed network model	64
4.3.2.1	Optimization using ALO in cluster formation	64
4.3.2.2	Fog clustering with k-means.....	65
4.3.2.3	Fog clustering using self-organizing maps (SOM).....	66
4.3.2.5	Fog clustering with fuzzy networks.....	66
4.3.2.6	Ensemble model for fog clustering.....	68
4.4	Results and discussions.....	69
4.4.1	Parameters and assumptions for simulation environment.....	69
4.5	Results.....	71

5. Conclusion and Future Scope	74
5.1 Conclusion	74
5.2 Future scope	76
List of Publications	80
References/Bibliography	81

1. Introduction

1.1 Fog Computing

Fog computing is a decentralised architecture that places data sources, such as application users and sensors, to the edge of the cloud. This architecture places these sources near to the storage and processing components as shown in figure 1.1.

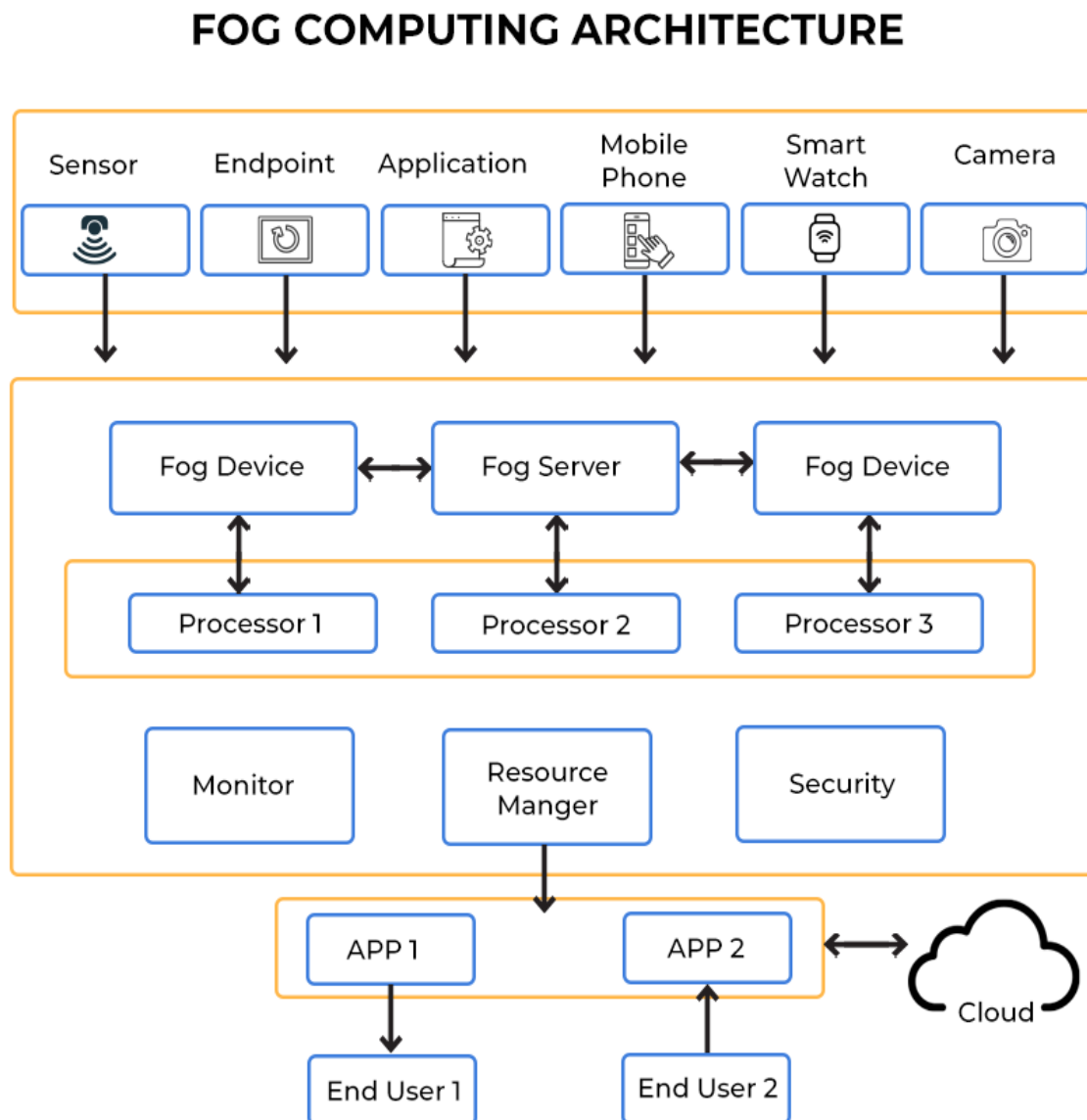


Figure 1.1: Fog Computing Architecture [1]

In 2025, 65% of the world's population, or 5.17 billion people, are expected to have internet access, according to Domo's ninth annual "Data Never Sleeps" infographic. A further

prediction made in the infographic was that by 2025, worldwide data usage, which was at 79 zettabytes in 2016, will reach 180 zettabytes. Wireless technology has advanced quickly, giving users of mobile devices access to powerful computer resources.

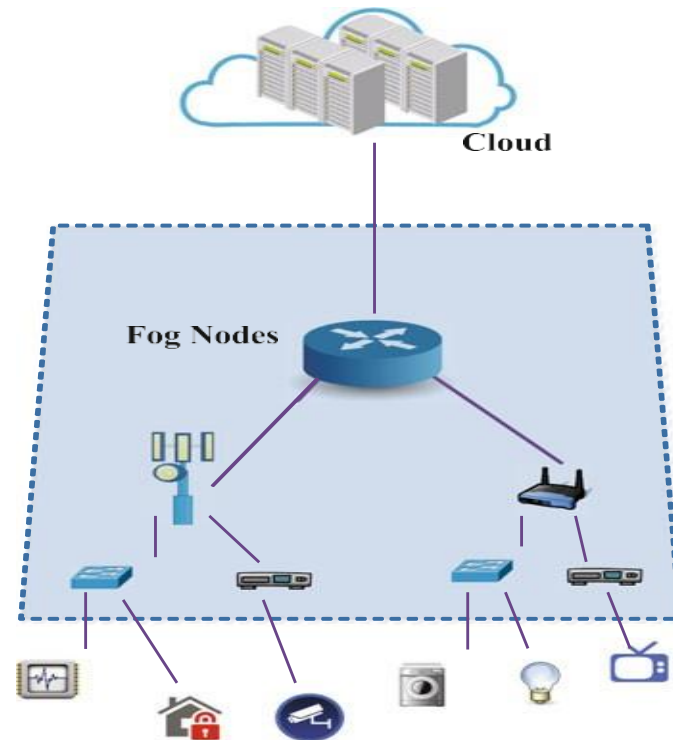


Figure 1.2: Cloud & Fog Layer [3]

Figure 1.2 shows the schematic process & transmission of data between cloud layer, fog layer & end devices. Although fog computing has several benefits, it also has some disadvantages, notably in terms of power usage and resource efficiency. There is an increase in resource demand when more devices connect to fog nodes, which might result in bottlenecks and poor performance. For battery-powered gadgets, power efficiency becomes essential as well in order to guarantee longer battery life and lower energy expenses.

Consumer data varies in volume depending on the industry. Data-intensive experiences are made possible by the Internet of Things (IoT) across a number of industries, including smart energy grids and fitness trackers. Cloud computing and artificial intelligence are essential for processing and dynamically storing enormous volumes of data. This information may be used by businesses to make wise decisions and reduce technical and financial risks.

Organizations are now evaluating the quality and volume of data they keep in the cloud as a result of the data explosion. Due to the enormous amount of data and the rising costs of cloud services, real-time reaction becomes difficult.

As an illustration, visualise a temperature sensor on a manufacturing line. It might be wiser to first check for recent temperature changes rather than sending each temperature reading directly to the cloud. The data is only transferred to the cloud for archiving when a temperature change is seen, signalling a possible problem in the production process. When dealing with data-intensive equipment like CCTV cameras that create a lot of video and audio data, this selective approach becomes very crucial.

Before sending data to the cloud, fog computing processes and stores it locally. Fog computing, which focuses on resource-intensive and comprehensive analytics, shifts some of the effort to the cloud by using less powerful processors in devices. In situations when an organization's system interacts with the outside world, such as fault warnings and alarm statuses, cloud "edge" devices manage instantaneous and time-sensitive analytics.

Fog computing includes edge computing, which processes data as it is being created. Routers, cameras, switches, embedded servers, sensors, and controllers are examples of edge devices. While using edge computing, the system puts less emphasis on exchanging data with the cloud and more emphasis on keeping and processing it locally on the device.

A layer is added by fog computing between edge devices and the cloud. This layer is made up of tiny computer servers that are placed close to edge devices and connected to other servers, centralised cloud servers, and other servers to allow intelligent data flow. These servers operate together to pre-process data, store it temporarily, and perform rule-based real-time monitoring, which improves system performance and minimises network traffic.

1.2 Basic Components of Fog Computing

The components of a fog computing system might vary depending on how it is implemented, but they usually include the following

1.2.1 Physical & virtual nodes (end devices)

Application servers, edge routers, mobile phones, wearables, and sensors are some examples of these gadgets that serve as points of communication with the outside world. They produce data and may differ in terms of their software, hardware, storage, and processing power.

BASIC COMPONENTS OF FOG COMPUTING

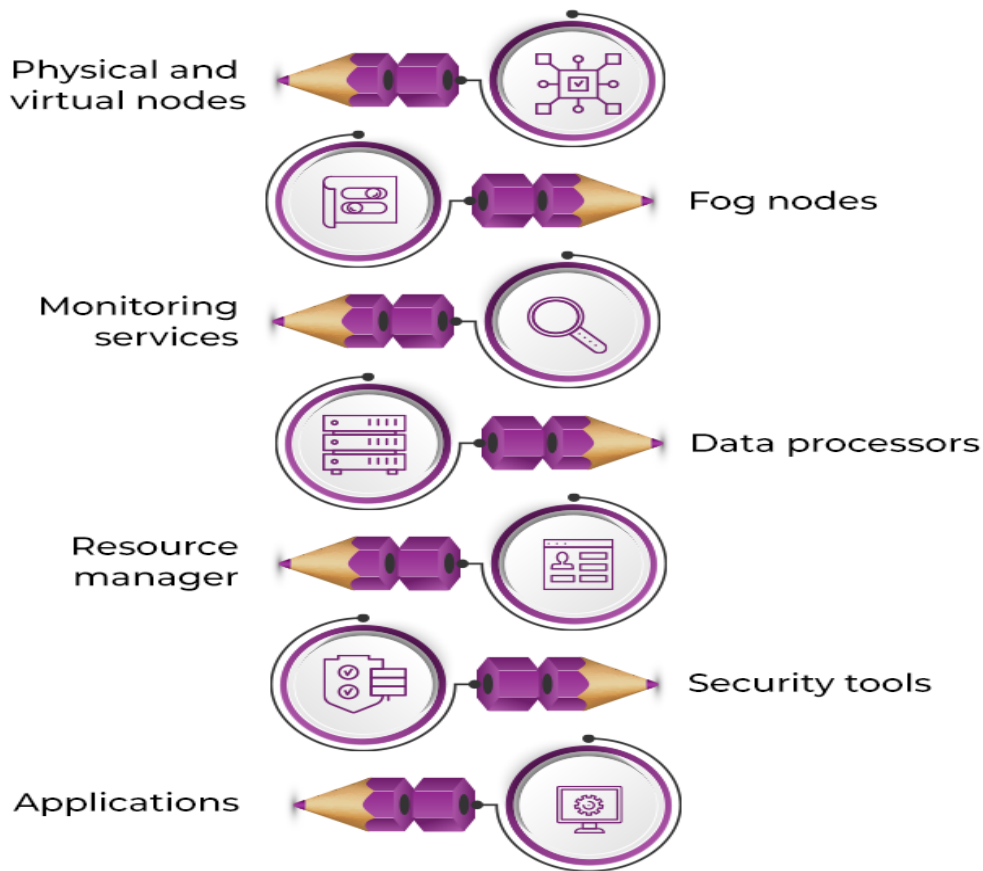


Figure 1.3: Fog Computing Components [2]

1.2.2 Fog nodes

Figure 1.3 shows all the fog computing components. The produced data is collected by fog nodes, which include fog servers, fog gateways, and fog devices. Fog gateways enable information interchange between fog devices and servers, while fog servers calculate the data to identify the best course of action. Understanding of device control, network connectivity, and hardware configurations are necessary for configuring fog nodes.

1.2.3 Monitoring services

Monitoring services include application programming interfaces (APIs) that track system performance and resource availability. These services monitor communication and guarantee the functional state of fog nodes and end points. Monitors can help optimize resource consumption by providing the option to choose between transferring data to the cloud for a fee or waiting for a node to become available.

1.2.4 Data processors

Data processors operating on fog nodes filter, reduce, and occasionally reconstruct faulty data originating from end devices. Users have the flexibility to decide whether to transfer the data to the cloud for long-term archival purposes or retain it locally on fog servers. Data processors provide standardised and configurable interfaces to other system components while homogenising information from multiple sources for simple transportation and transmission. Certain processors have the ability to fill up information gaps brought on by sensor failures using prior data, guaranteeing continuous application operation.

1.2.5 Resource manager

Independent fog nodes are synchronised by the resource management. The resource manager plays a crucial role in managing data backup to prevent data loss, efficiently allocating and reallocating resources among nodes, and coordinating data transit between nodes and the cloud. Collaborating with the monitor, the resource manager identifies areas and periods of high demand, ensuring optimal utilization of fog servers and data without duplication.

1.2.6 Security tools

When fog computing components directly interface with sources of raw data, security is a key concern. Due to the prevalence of wireless networks for communication, encryption is crucial. In order to secure data integrity and confidentiality on a fundamental level, security technologies also incorporate user and access control.

1.2.7 Applications

Applications utilize the data provided by the fog computing system in a meaningful manner to deliver user services effectively. These programmes rely on a layer of abstraction that makes available a standard user interface and communication protocols. It is typical practise to accomplish this connection via APIs and web services.

1.3 Examples and Use Cases of Fog Computing

Cloud computing has become all-pervasive, but fog computing is just now starting to emerge to address the many latency problems that plague IoT devices.

1.3.1 Smart homes

Fog computing is commonly used in smart homes to enable customised alarm systems and the automation of numerous events, including the management of lights, sprinklers, heating and cooling systems, and intercom systems. Smart homes improve convenience and energy efficiency by utilising fog computing.

A smart home has a technology-controlled heating and cooling system, like the Nest Learning Thermostat, as well as smart alarms, programmable blinds and sprinklers, smart lighting, and intercom systems that can communicate with both individuals inside the home and those at the door. With fog computing, a customised alarm system may be created. Furthermore, fog computing can be leveraged to automate specific tasks, such as activating sprinklers based on time and temperature conditions. A common and prominent use case for fog computing is in the context of smart homes. Smart alarms, programmable blinds and sprinklers, smart lighting, intercom systems that can interact with both individuals inside the home and those at the door, as well as technology-controlled heating and cooling systems like the Nest Learning Thermostat are all features of a smart home. Fog computing can enable the creation of a personalized alarm system, offering customized features and functionalities. Additionally, it can automate various operations, such as activating sprinklers based on specific temperature and time conditions.

1.3.2 Smart cities

Fog computing is essential to smart cities in many ways, including traffic management. Pedestrians, bicycles, and automobiles are detected by sensors placed at traffic lights and road barriers, giving real-time data for effective traffic control. Traffic signals are able to automatically change based on the volume of traffic at any given time thanks to fog nodes' analysis of the data.

1.3.3 Video surveillance

In order to overcome the difficulties of transporting huge video feeds across networks, fog computing is frequently utilised in video surveillance systems. Fog nodes close to the surveillance equipment process the footage in real time, looking for anomalies or potential threats and immediately notifying the appropriate authorities.

1.3.4 Healthcare

Due to the stringent regulations like HIPAA (Health Insurance Portability and Accountability Act), the healthcare industry adheres to high standards of compliance. Leveraging fog computing, the healthcare sector can ensure real-time monitoring and prompt responses to critical situations like diagnose cancer in the early stages, and therefore, techniques that can improve accurate detection of breast cancer are always solicited [173]. This technology enables healthcare providers to maintain strict compliance while delivering efficient and timely care. Fog computing guarantees minimum latency, enabling fast identification of crisis signals and early response by leveraging data from wearables, blood glucose monitors, and health applications.

1.3.5 Others

Fog computing is also used in the retail, oil and gas, government and military, and hospitality sectors. Personal assistants like Siri and Alexa may be found on a variety of platforms and are compatible with most of them, including smartwatches. Fog computing is destined to play a big part in a variety of various market sectors due to its versatility and popularity. Fog computing also has advantages for a number of businesses, such as retail, oil and gas, government and military, and hospitality. Aides to the individual Real-time solution providers must include fog computing into their existing cloud architecture. The definition of prevention procedures and of emergency plans is one of the main task of PCR to protect and often save human life [172].

1.4 Need of Fog Computing

In the current landscape of data organization and processing, two prominent paradigms have emerged fog computing and cloud computing. Fog computing focuses on distributing the computational load to the edge of the network, nearer to the data source. In contrast, cloud computing emphasizes centralizing data processing and storage in remote data centres Cloud computing is gradually becoming accepted in different sectors and businesses are beginning to adopt the concept's shared infrastructure and applications [174].

The table 1.1 shows the comparison between fog computing & cloud computing. Hence the following are some benefits of fog computing over cloud computing.

1.4.1 Reduced Latency

Fog computing offers a significant advantage in terms of latency reduction. Unlike cloud computing, where data must travel between the data centre and the end-user, leading to processing delays, fog computing enables data processing closer to the network edge. This proximity reduces the distance that data needs to traverse, resulting in minimized latency and improved processing time.

1.4.2 Improved Security

Increased security is a benefit of fog computing as well. Data is processed and stored centrally in a cloud-based system, which increases their susceptibility to security breaches. Fog computing spreads data among a number of edge devices, making it more challenging for hackers to breach the system.

1.4.3 Better Reliability

Fog computing enhances network dependability by distributing processing power throughout the network. In the event of a device malfunction, other devices within the network can compensate, ensuring uninterrupted operations. Unlike cloud-based systems where a failure in the data centre can lead to network downtime, fog computing allows for greater resilience and prevents single points of failure, thereby maintaining the smooth functioning of the network.

1.4.4 Reduced Bandwidth Usage

The amount of bandwidth a network uses can be decreased using fog computing. Data doesn't need to be repeatedly sent to a centralised data centre since processing takes place closer to the network's edge, conserving bandwidth.

1.4.5 Lower Costs

Lastly, fog computing offers the potential for cost savings compared to cloud computing. By offloading workloads to edge devices, companies can reduce the need for expensive centralized servers, leading to lower operational expenses. This shift in resource allocation

allows for more cost-effective infrastructure, making fog computing an attractive option for organizations looking to optimize their expenses.

1.5 Comparison of Fog & Cloud Computing

Parameter	Fog computing	Cloud computing
Latency	Low	High
Response time of the system	High	Low
Architecture	Distributed	Centralized
Device Communication	From the edge directly via Various standards and protocols, One hop	from a distance via internet, multiple hops
Data processing	Close to the information source in short-term time	Far from the information source in long-term time
Computing capability	Lower (FC reduces the data amount sent to Cloud computing.)	Higher (cloud computing does not render reduction in data while transforming or sending data)
Server nodes	Very large nodes that have limited computing power and storage	Few nodes with scalable computing power and storage
Security	can be defined, More secure	Undefined, Less secured
Working	Indoor or outdoor	Warehouse-size building that have air

environment		conditioning systems
Location of server nodes	On the edge of the local network	Within the internet

Table 1.1: Comparison of Cloud & Fog Computing [49]

The Fog is a highly virtualized platform but not a replacement of Cloud computing [177]. Cloud computing is an emerging computing paradigm which allows sharing of massive, heterogeneous, elastic resources among users [184] whereas Fog computing has emerged as a potential solution for supporting Internet of Things (IoT) and Edge Computing applications. With the increasing number of IoT devices generating massive amounts of data and the latency involved in transferring this data to and from the cloud, traditional cloud-based architectures are becoming less effective. In contrast, fog computing offers a distributed computing architecture that brings compute and storage closer to the devices, resulting in reduced latency and improved response time.

To address the challenges associated with fog computing, researchers have developed various power-efficient frameworks for optimizing resource utilization. This paper aims to explore the current state of power-efficient fog computing frameworks and their potential applications. It will discuss the existing challenges and opportunities in this field and provide suggestions for future research endeavours.

1.6 Challenges

Furthermore, in the realm of fog-based services, it is essential to integrate various aspects of a controlled environment while placing a strong emphasis on security. Several challenges arise in Fog Computing (FC) concerning its structure, service orientation, and security. These challenges encompass

1.6.1 Structural Issues

- The potential FC framework includes various components in both edge and core networks, often with different types of processors specialized for specific tasks. Enabling these components for general-purpose computing in addition to their conventional activities presents significant difficulties.

- Selecting appropriate nodes within the Fog, determining deployment locations, and managing resources are crucial, depending on the operational environment and requirements.
- The distribution of computational nodes in Fog Computing (FC) throughout the edge network allows for sharing and virtualization. Identifying appropriate techniques and metrics to facilitate collaboration among internal nodes and enable efficient resource allocation is crucial.
- The structural configuration of FC is well-suited for integration with the Internet of Things (IoT). However, effectively incorporating diverse networking protocols such as vehicular networks and Content Distribution Networks (CDNs) presents significant challenges in terms of ensuring seamless connectivity and optimal performance.

1.6.2 Service-Oriented Challenges

- Developing large-scale applications on resource-constrained Fog nodes is a complex task compared to traditional data centres. The availability of a suitable programming platform for developing distributed applications in the Fog becomes crucial to address this challenge.
- Defining policies that govern the allocation of services and computational tasks across IoT devices, sensors, Fog infrastructure, and the Cloud is essential. Additionally, designing interfaces for effective data visualization poses a challenge in the context of Fog Computing.
- Service Level Agreements (SLAs) in Fog Computing are influenced by multiple factors, including application characteristics, service costs, data flow, energy usage, and network status. Defining corresponding Service Level Objectives (SLOs) and service provisioning metrics becomes a challenging task. It is also important to maintain the inherent Quality of Service provided by the Fog nodes.

1.6.3 Security Aspects

- Fog computing faces security vulnerabilities as it relies on traditional networking components as its foundation.
- Ensuring authorized access to services in a highly distributed paradigm and maintaining privacy in Fog Computing is challenging.

- The implementation of security mechanisms to ensure data-centric integrity in Fog computing can have a substantial impact on the Quality of Service, potentially resulting in performance degradation.

1.7 Fog Computing Best Practices to Follow in 2023

A fog engine's implementation presents unique difficulties. Businesses frequently opt for a centralised approach for their technical infrastructure as administration becomes simpler. A decentralised collection of heterogeneous fog devices must be installed, which raises new maintenance and compatibility problems. Here are the top 10 fog computing best practises for 2023.

1.7.1 Ensure provision for flexibility

The seamless integration of diverse hardware and software components is a key advantage of Fog computing. However, without a flexible interfacing program to facilitate this integration, the complexity of the system can escalate rapidly. Web-based services and APIs must take new physical and virtual sensors into account. Both the current cloud solution and other fog nodes must be seamlessly integrated with the fog engine. Figure 1.4 shows the best practices in fog computing.

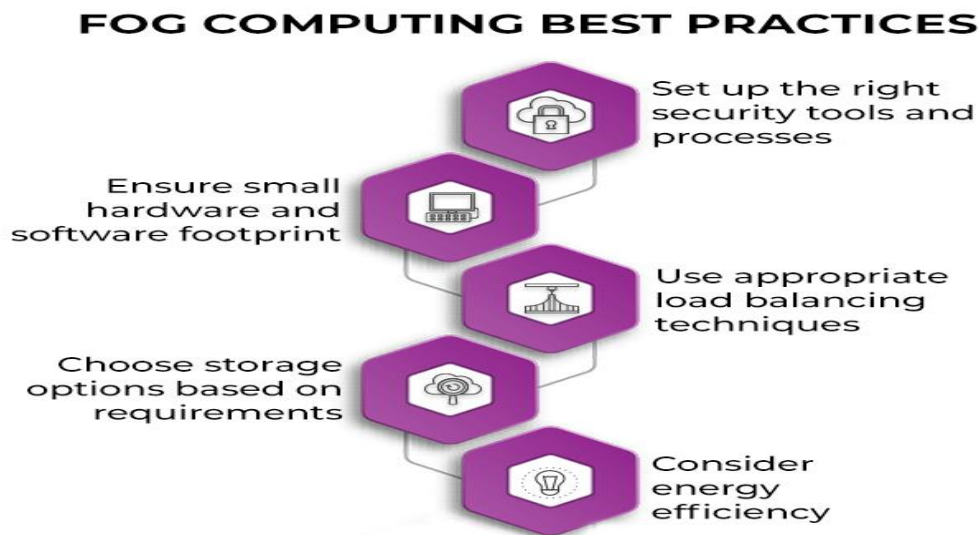


Figure 1.4: Fog Computing Best Practices [5]

1.7.2 Implement a fog console

To effectively manage the fog computing infrastructure, administrators must monitor all installed fog nodes and decommission any that require it. This decentralized architecture can be administrated from a centralized point by addressing vulnerabilities in zombie fog devices and enforcing compliance with regulations. Robust compliance checks necessitate a powerful reporting and logging engine, along with a management panel.

1.7.3 Apply access control at the fog node layer

In fog computing, users have direct access to cloud-based services, similar to traditional cloud-based configurations. Cloud providers offer specialized access management systems that can be combined with Identity and Access Management (IDAM) solutions from other vendors. Fog layers act as intermediaries between users and the cloud, following the same permission procedures and rules. The fog engine must be informed of the requester's identity for proper authentication.

1.7.4 Set up appropriate security tools & processes

Security is a significant concern in fog computing, given its local and decentralized nature. Fog security prioritizes user authentication and requires secure data transmission, especially over wireless communication channels. Application service requests also require verification of their application signatures to ensure integrity. Comprehensive security measures, including regular vulnerability assessments, patch management, and upgrade processes, must be established.

1.7.5 Ensure small hardware & software footprint

Optimal selection of hardware and software for each sensor is crucial in fog computing. The objective is to maintain a small hardware and software footprint, avoiding excessive complexity and over-engineering at the fog level. Overburdening the fog nodes with unnecessary resources can lead to increased costs and potential security risks. Careful analysis of sensor functionality and the accompanying fog node is necessary, and their lifespans can be controlled and automated through a centralized console.

1.7.6 Incorporate threat detection & mitigation at the fog level

To enhance overall security, proactive risk identification at the fog level is recommended, even before they impact the main cloud infrastructure. The security component of the fog engine should be configured to detect and address unusual application and user behaviors.

With the diverse range of hardware and software components involved, it is crucial to address vulnerabilities promptly by adhering to a predetermined process and schedule for security patches and upgrades.

1.7.7 Use relevant load balancing techniques

Efficient monitoring and load balancing of fog nodes are essential to achieve the benefits of increased network traffic and reduced latency in fog computing. Overloading or underloading the fog nodes can hinder the desired outcomes. Load-balanced fog layers can optimize various Quality of Service (QoS) metrics, including resource utilization, throughput, performance, response time, cost, and energy efficiency.

1.7.8 Choose storage options based on requirements

Different sensor levels supported by the organization require varying storage options. Local flash chips are suitable for storing security keys, log files, and tables, while rotating discs are more appropriate for large media libraries. In-memory storage on a data server should be avoided in the fog architecture. The cost of storage per GB should be considered when selecting hardware.

1.7.9 Consider energy efficiency

Additional hardware in fog computing can lead to undetected energy waste. It is crucial to adopt energy-efficient practices such as ambient cooling, low-power silicon, and selective power-down modes to optimize energy efficiency.

1.7.10 Design for uninterrupted fog services

To ensure continuous operation and prevent a single node failure from causing service disruption, independent operation between fog nodes and the main computer system is necessary. Specialized data backup strategies tailored to each fog node's purpose should be implemented and regularly repeated.

Fog computing enhances business agility and improves Quality of Service (QoS) by processing data closer to the source, resulting in faster user experiences. It offers cost savings by reducing the need for excessive cloud storage and improves network performance. As cellular networks become more reliable, businesses should consider fog computing as a

strategic component of their infrastructure, alongside cloud computing, to capitalize on its benefits and align with the evolving technological landscape.

1.8 Need of power-efficient frameworks in fog computing

Power-efficient frameworks are crucial in fog computing to optimize resource utilization and reduce energy consumption. These frameworks dynamically allocate resources based on workload and energy consumption, ensuring efficient usage and prolonging battery life. One example is the Energy-Aware Task Offloading (EATO) technique, which considers both energy consumption and network latency to determine the optimal fog node for task offloading. By reducing energy consumption and minimizing latency, this approach improves system performance. Another framework, Adaptive Resource Allocation (ARA), dynamically allocates resources based on workload and energy consumption. By optimizing resource usage, it reduces energy consumption and enhances overall efficiency. Power-efficient fog computing frameworks have applications in healthcare, transportation, and smart cities. In healthcare, real-time monitoring of patients' vital signs can reduce hospitalization and improve care quality. In transportation, optimizing traffic flow and energy consumption enhances efficiency and reduces carbon footprint. Despite the benefits, challenges remain. The heterogeneity of devices and network architectures complicates the development of standardized solutions. Real-time decision making, security, and privacy are also concerns that need to be addressed. Further research is required to develop scalable and interoperable power-efficient frameworks. This involves exploring novel resource allocation techniques, improving energy harvesting and storage, and addressing security and privacy issues. Power-efficient frameworks in fog computing have the potential to transform IoT and Edge Computing applications. By optimizing resource utilization and reducing energy consumption, fog computing improves performance, efficiency, and sustainability. Continued research and development will shape the future of computing.

1.9 Integration of AI/ML

An agreed definition of Artificial Intelligence (AI) is the study of intelligent behaviour achieved through computations [171]. Due to the advantages of fast learning speed and low computational cost, Extreme Learning Machines have become popular for solving many complex problems [173]. Additionally, the incorporation of Artificial Intelligence (AI) and Machine Learning (ML) techniques within power-efficient fog computing frameworks can significantly boost their performance and efficiency.

Figure 1.5 shows the process of integrating AI/ML which involves the data preparation, Training sets generation, algorithm training and deployment & monitoring. AI and ML techniques can be used to predict the workload and energy consumption of the fog nodes, enabling more efficient resource allocation and management. Furthermore, the utilization of AI and ML enables intelligent task offloading and scheduling, resulting in enhanced system performance and reduced energy consumption.

WSNs generate vast volumes of data from numerous sensor nodes, and ML algorithms offer a valuable solution for data fusion. Data fusion pertains to the amalgamation and analysis of data from multiple sensors to attain enhanced accuracy and reliability. ML techniques such as clustering, classification, and regression can be effectively employed to execute data fusion within WSNs.

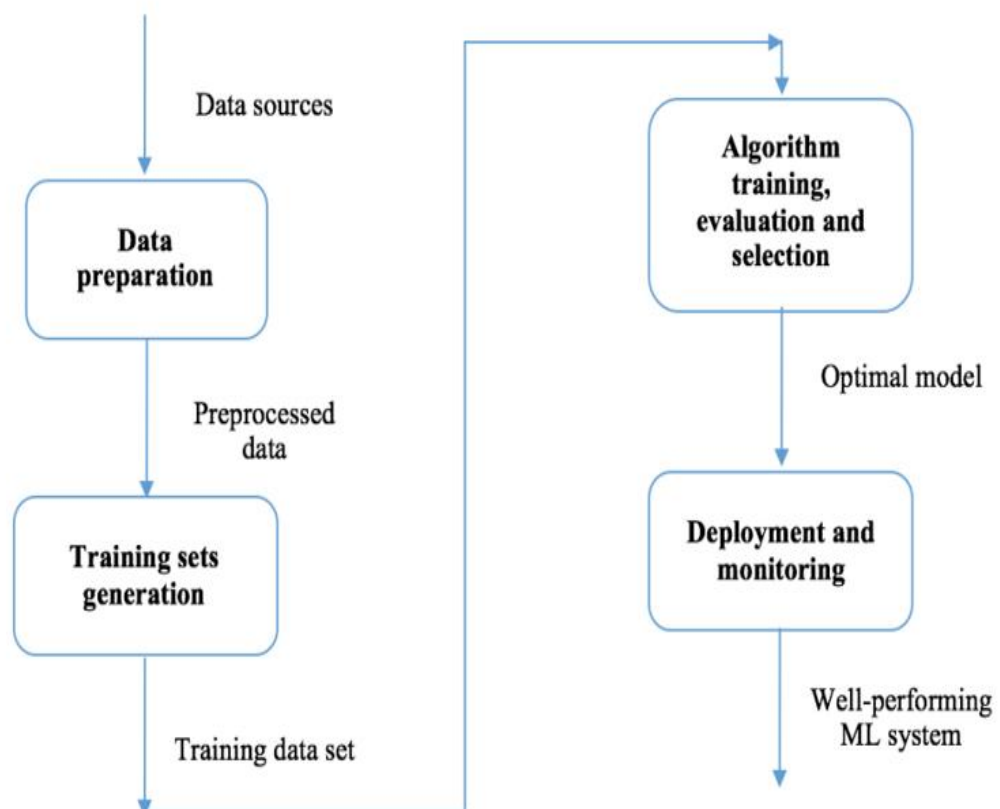


Figure 1.5: Integration of AI/ML [171]

Furthermore, the development of energy-efficient hardware and software for fog nodes can also contribute to the efficiency of power-efficient fog computing frameworks. Advances in hardware technologies such as low-power processors and sensors, and energy harvesting and storage technologies can enable fog nodes to operate with minimal energy consumption. In addition, developing energy-efficient software that optimizes the usage of resources and reduces unnecessary computation can further enhance the efficiency of fog computing systems.

Power-efficient fog computing frameworks have the potential to revolutionize the management of IoT and Edge Computing applications, bringing significant benefits to businesses, organizations, and the environment. By improving performance, efficiency, and sustainability, these frameworks address the challenges of resource utilization and power efficiency. With the integration of AI and ML techniques, advancements in energy-efficient hardware and software, and ongoing research, power-efficient fog computing frameworks will continue to shape the future of computing.

The adoption of power-efficient fog computing frameworks not only drives positive outcomes for businesses but also contributes to environmental conservation. By reducing energy consumption and enhancing computing system efficiency, fog computing plays a vital role in lowering carbon emissions and promoting resource conservation. This aligns with global efforts to combat climate change and foster sustainability.

To ensure the efficient and sustainable operation of IoT and Edge Computing applications, power-efficient frameworks for resource utilization in fog computing are crucial. Developing such frameworks requires collaboration across disciplines, encompassing computer science, engineering, and environmental science. By addressing resource utilization and power efficiency challenges, power-efficient fog computing frameworks pave the way for more efficient, sustainable, and resilient computing systems, benefiting businesses and the environment alike.

As the demand for IoT and Edge Computing applications continues to soar, the importance of power-efficient fog computing frameworks grows. These frameworks optimize resource usage, reducing energy consumption and enhancing computing system performance. Moreover, they contribute to the advancement of sustainable and resilient computing systems, effectively tackling the challenges associated with resource utilization and power efficiency.

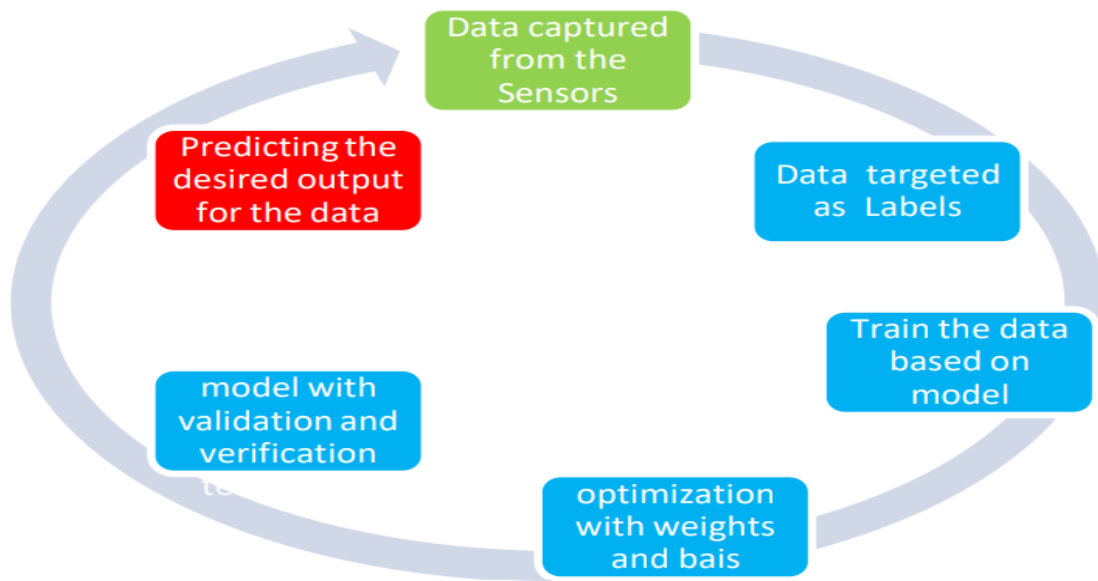


Figure 1.6: Modelling with ML Algorithms [172]

Figure 1.6 showcases the life cycle of Modelling with ML algorithm. Because The development of power-efficient fog computing frameworks requires collaboration between researchers, engineers, and businesses to develop solutions that are both effective and practical. By leveraging the latest advances in hardware and software technologies and integrating AI and ML techniques, power-efficient fog computing frameworks can achieve unprecedented levels of efficiency and performance.

The development of power-efficient frameworks for resource utilization in fog computing is crucial to overcome the challenges associated with resource utilization and power efficiency in IoT and Edge Computing applications. These frameworks play a significant role in shaping the future of computing, benefiting businesses, society, and the environment.

To fully realize the potential of power-efficient fog computing frameworks, several research directions must be explored. One such direction involves the development of efficient resource allocation and management techniques that can effectively handle the diverse devices and network architectures present in fog computing. Additionally, advancements in energy harvesting and storage techniques for fog nodes are essential to minimize energy consumption and optimize their operation.

Ensuring the security and privacy of fog computing systems is another critical aspect that demands further research. As fog nodes operate in dynamic and distributed environments,

robust security measures must be developed to protect data and devices from potential threats.

Furthermore, the development of energy-efficient hardware and software for fog nodes is paramount in enhancing the overall efficiency and performance of fog computing systems. This entails exploring low-power processors, sensors, and software algorithms that can minimize unnecessary computations.

Achieving power-efficient frameworks for resource utilization in fog computing necessitates multidisciplinary collaboration among researchers, engineers, and businesses. By focusing on research areas such as efficient resource allocation, energy harvesting and storage, security, and energy-efficient hardware and software, we can address the challenges posed by resource utilization and power efficiency in IoT and Edge Computing applications.

Moreover, integrating sustainability principles into the development of power-efficient fog computing frameworks is of utmost importance. Sustainable fog computing involves designing and implementing systems that have minimal environmental impact while meeting user requirements. This can be achieved through the use of renewable energy sources to power fog nodes, eco-friendly materials in node construction, and proper end-of-life disposal practices.

Furthermore, fog computing can contribute to sustainability in various domains such as agriculture, transportation, and smart cities. Optimizing resource usage in farming, improving traffic flow and fuel efficiency, and reducing energy consumption in cities are examples of how fog computing can promote sustainability.

By emphasizing sustainability in the design and implementation of power-efficient fog computing frameworks, we can ensure their long-term viability while positively impacting the environment.

1.10 Need of Power Efficiency for Wireless Sensor Networks

Wireless sensor networks (WSNs) play a vital role in enabling the Internet of Things (IoT) and find applications in diverse fields such as environmental monitoring, industrial automation, smart homes, and healthcare. However, WSNs encounter various challenges, including limited power, computational resources, and storage capacity. These constraints

significantly impact the lifetime and performance of WSNs, posing challenges to their deployment and long-term sustainability.

To address these challenges, fog computing has emerged as a promising paradigm that extends cloud computing to the network edge, closer to data sources and end-users. By leveraging fog computing, WSNs can offload computation and storage tasks from resource-constrained sensor nodes to fog nodes, which possess greater resources and can offer low-latency and context-aware services. This integration allows WSNs to overcome their limitations and achieve more power-efficient and scalable solutions. The performance metrics considered are spectral efficiency, throughput, transmission power & load balancing [183].

In a typical WSN, nodes are composed of sensor nodes, relay nodes, and sink nodes. Sensor nodes are responsible for sensing the environment and transmitting data to relay nodes. Relay nodes facilitate data transmission between sensor nodes and the sink node, which collects and forwards the data to the base station for further processing.

One of the primary challenges faced by WSNs is the constraint of limited power supply. Sensor nodes typically rely on batteries with finite lifespans, making energy consumption a critical consideration in the design and operation of WSNs. Additionally, WSNs encounter difficulties related to communication, computation, and storage due to the large volume of data generated by sensor nodes, requiring substantial processing and storage resources.

Fog computing is a paradigm that extends cloud computing to the edge of the network. Fog computing enables computation, storage, and networking services to be deployed closer to the data sources, which can significantly reduce the latency and bandwidth requirements of the network. By leveraging fog computing, WSNs can offload computation and storage tasks from the sensor nodes to the fog nodes, which have more resources and can provide low-latency and context-aware services.

The power-aware data acquisition and processing module is responsible for optimizing the data acquisition and processing process in WSNs to minimize energy consumption. The module includes several sub-components, including a duty cycling mechanism, a data compression and aggregation mechanism, and a data filtering mechanism. The duty cycling mechanism enables the sensor nodes to operate in a low-power mode when not actively sensing the environment. The data compression and aggregation mechanism reduce the amount of data generated by the sensor nodes, which reduces the energy consumption of the

transmission process. The data filtering mechanism filters out redundant or unnecessary data to further reduce the energy consumption of the network.

The fog-enabled data aggregation and filtering module is responsible for aggregating and filtering the data received from the sensor nodes. The module includes several sub-components, including a fog-enabled data aggregation mechanism, a fog-enabled data filtering mechanism, and a fog-enabled data storage mechanism. The fog-enabled data aggregation mechanism aggregates the data received from the sensor nodes at the fog nodes to reduce the amount of data transmitted to the base station. The fog-enabled data filtering mechanism filters out redundant or unnecessary data at the fog nodes to further reduce the energy consumption of the network. The fog-enabled data storage mechanism stores the data at the fog nodes, which enables the data to be processed locally and reduces the data transmission requirements of the network.

The dynamic workload distribution and resource allocation module is responsible for dynamically distributing the workload and allocating the resources in the network to optimize the performance and energy consumption of the network. The module includes several sub-components, including a dynamic workload distribution mechanism, a resource allocation mechanism, and a network topology management mechanism. The dynamic workload distribution mechanism distributes the workload among the sensor nodes and fog nodes to balance the energy consumption and performance of the network. The resource allocation mechanism allocates the resources in the network to optimize the energy consumption and performance of the network.

Finally, the network topology management mechanism is responsible for managing the network topology to ensure efficient data transmission and minimal energy consumption. This mechanism includes sub-components such as node placement and mobility management.

The proposed power-efficient framework for WSNs using fog computing has several advantages over existing approaches. First, the framework enables energy-efficient data acquisition, processing, and transmission, which extends the lifespan of the WSN nodes and reduces the need for battery replacement. Second, by leveraging fog computing, the framework provides low-latency and context-aware services to WSNs, which improves the performance and efficiency of the network. Third, the framework enables dynamic workload

distribution and resource allocation, which optimizes the energy consumption and performance of the network.

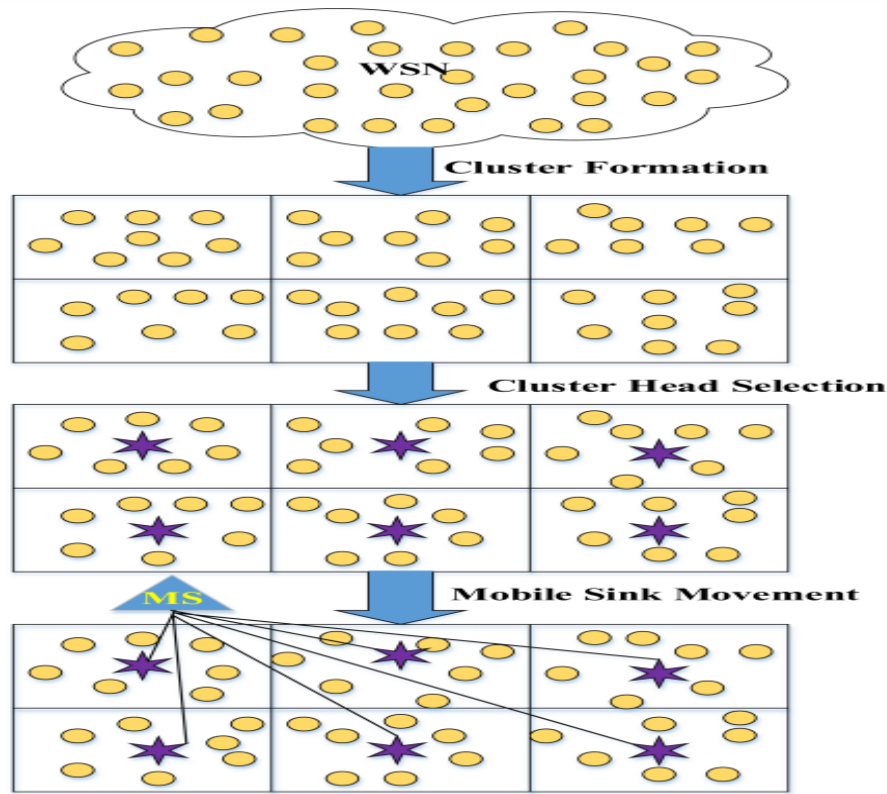


Figure 1.7: Cluster Formation in Wireless Sensor Network [6]

Figure 1.7 shows the cluster formation in Wireless Sensor Networks. The proposed framework has several potential applications in various domains, including environmental monitoring, health monitoring, and smart cities. In environmental monitoring, the framework can be used to monitor the air and water quality, weather conditions, and natural disasters. In health monitoring, the framework can be used to monitor the health and well-being of patients in hospitals and homes. In smart cities, the framework can be used to monitor the traffic flow, air quality, and energy consumption of buildings.

The contributions of this thesis are twofold. First, we propose a power-efficient framework for WSNs using fog computing, which optimizes the energy consumption and performance of the network. Second, we evaluate the proposed framework using simulation and experimental studies and demonstrate its effectiveness in reducing the energy consumption and improving the performance of WSNs.

1.11 Research Motivation

Internet of Things (IoT) is a heterogeneous ubiquitous network based upon modern computational intelligent techniques [178] and Industrial operators use the Industrial Internet of Things (IIoT) to automate the systems, deploy sensors to monitor, measure, analyse data increase revenue opportunities and improve efficiencies for manufacturing operations in wireless sensors network (WSN) has let the manufacturing industry is experiencing substantial benefits. The quantity of data from these plants can be computed in petabytes (1 million gigabytes) because of millions of connected sensors on industrial control systems (ICSs), videos surveillance cameras covering plants, millions of streaming, industrial robots, autonomous drones and so on.

Operational technology environments approached by traditional information technology can not meet the reliability security, network bandwidth difficulties, privacy and necessary volume in a controlled operational environment. Hence, Fog environment is essential in Wireless Sensors Networks (WSNs) to allow IIoT to reach its potential.

The approved data-based routing protocols for connecting Fog nodes are ordinarily vulnerable to a huge quantity of energy and power voids in Wireless Sensors Networks. There is need to create Power-efficient framework for Fog environment which can handle, migration of operators, dynamic load in the network and migration of tasks.

1.12 Thesis Organisation

Chapter 1 Introduction

This chapter discusses the foundation of important concepts need to be known for better understanding of the thesis work. General frameworks of Power efficient frameworks for WSNs using fog computing environments are studied and different techniques are discussed in detail.

Chapter 2 Literature Survey

This chapter explores various Power Efficient techniques presented in the literature. Detailed review of energy-efficient cross-layer-sensing clustering method, Distributed and Morphological Operation-based Data Collection Algorithm (DMOA), and trust-based secure routing is given. Research gaps and motivation are discussed. The research objectives are also presented.

Chapter 3 Power-Efficient Optimized Clustering Method (PEOCM) with Intelligent Fog Computing for WSNs

The proposed algorithm focuses on ALO only on the bottom layer and not on the fog layer. To evaluate the performance of the proposed algorithm, the results are compared with the findings of several traditional algorithms, such as energy-efficient cross-layer-sensing clustering method, Distributed and Morphological Operation-based Data Collection Algorithm (DMOA), and trust-based secure routing. Comparative results showed that the proposed algorithm presented significant improvement.

Chapter 4 Machine Learning Based Power Efficient Optimized Communication Ensemble Model with Intelligent Fog Computing for WSNs

This study is motivated from fog computing in which Ant lion optimization is applied in the bottom layer and an ensemble model of k-means, SOM and Fuzzy c-means techniques is used for clustering in FOG layer. To access model's performance, results are compared to the traditional methods. This model has shown significant improvement in terms of energy consumption as compared to traditional model ECCM. Proposed model has also shown improvement in terms of load balancing.

Chapter 5 Conclusion and Future Scope

In this chapter, conclusion of this thesis is presented. All the proposed Power efficient frameworks are concluded in this chapter. A direction for future scope of this research work is also discussed.

2. Literature Survey

2.1 Background

In recent years, cloud computing has gained popularity as a platform for processing user requests in cloud data centres [51]. At the same time, the demand for Internet of Things (IoT) connected devices and applications, such as smartphones, gadgets, and Google Glass, has significantly increased [52]. This surge in demand has presented challenges in optimizing Quality of Service (QoS) parameters like bandwidth, privacy, latency, storage, response time, security, and computing [51]. To address these challenges, fog computing has emerged as a promising paradigm that integrates IoT concepts and enables communication and interaction among nodes and devices [52].

Various frameworks and architectures for fog computing have been proposed in the literature [49,117], targeting different IoT applications. However, existing review papers have not provided taxonomy of fog computing frameworks, highlighting the need for a taxonomy that categorizes these frameworks and identifies research gaps. This article presents a systematic literature review of existing works, with a specific focus on frameworks and architectures related to fog computing. The findings of this review will support researchers in advancing their studies in the field of fog computing.

With the proliferation of IoT devices, mobile internet, and interconnected objects, a massive amount of data is being generated [53]. It is projected that by 2023, around 60-70 billion devices will be connected to the internet, leading to a significant increase in data transfer [152]. Traditional computing models like cloud and distributed computing face challenges in handling this scale of data [54].

Certain applications require quick response and mobility support, such as emergency response in smart healthcare, traffic light systems, and smart grids [144]. To address issues like high bandwidth, ultra-high latency, privacy-sensitive applications, and geographic dispersion, a computing paradigm is needed to assist cloud computing and process requests from connected IoT devices with minimal latency and response time [11].

The concept of fog computing was introduced by Cisco in 2012 to tackle the challenges faced by IoT applications in traditional cloud computing models [30]. It has received support from the OpenFog Consortium, which has published white papers on the subject. According to the

consortium, fog computing is defined as "A horizontal system-level architecture that allocates storage, computing, networking functions, and controls the nearest connected devices" [55]. Fog computing acts as a bridge between IoT nodes/devices and the cloud, facilitating storage, data management, networking, and computing on network nodes [59, 108, 109]. By reducing the burden on traditional cloud computing data centres [96, 120] and leveraging decentralized computing, fog computing enhances computational speed.

Currently, fog computing plays a significant role in various domains such as agriculture, healthcare, construction, industry, smart cities, smart homes, energy grids, transportation, environmental monitoring, and climate change monitoring [22, 79, 131].

Previous surveys have focused on frameworks and architectures based on fog computing. However, to the best of our knowledge, no systematic review exists in the literature that specifically focuses on the taxonomy of fog computing frameworks. Several authors have independently proposed frameworks and architectures for fog computing in various IoT application domains.

Frameworks are sets of rules and tools used to create and produce something [44]. They provide pre-built compositions or structures designed to extend specialized or general-purpose functionality [102]. Frameworks establish standard and specific guidelines to assist in the creation and extension of valuable services.

Habibi et al. [56] developed a taxonomy covering security, software, applications, systems, networking, and resource management in fog computing. They conducted a comprehensive study on architectural perspectives, describing the logical and physical components of the fog computing environment. The authors compared and evaluated proposed architectures based on various criteria, although their review was primarily focused on the architectural aspect of fog computing.

Margariti et al. [57] presented a survey of simulation tools and modeling methods to examine system costs in terms of configuration, network, and applications. They provided an overview of computing simulation tools, highlighting their characteristics, limitations, and efficiency.

Brogi et al. [58] conducted a review of methodologies for solving the service placement problem in fog computing. They provided a comprehensive summary of existing algorithms and open-source prototypes, with a focus on infrastructure characteristics and optimized metrics.

Markus and Kertesz [59] focused on the classification of software quality, analyzing source code and its features in various simulators.

Yousefpour et al. [60] conducted a survey and proposed a taxonomy for fog computing, discussing previous papers that focused on frameworks and architectures of fog computing.

Hou et al. [61] emphasized frameworks related to vehicular fog computing, including conventional fog devices, path computing, enlisted edge computing, and fog nodes as IoT hubs. They presented an overview of the resource model, generic architecture, and resource allocation framework for the fog computing model.

Fernando et al. [62] discussed the central elements of fog computing frameworks and architectures, highlighting opportunities for further research. They investigated and compared industry-related frameworks, considering parameters such as low latency, dynamicity, scalability, context-aware offloading, and heterogeneity that support the fog computing framework.

Mouradian et al. [63] conducted a comprehensive survey on fog computing, examining state-of-the-art research issues in the field.

These surveys have provided valuable insights into fog computing frameworks and architectures, but none have specifically focused on developing a taxonomy for fog computing frameworks. Table 2.1 shows the detailed survey of the existing work in the field.

Work	Fog Nodes	Nodal Collaboration	Provisioning Metrics	SLOs (Service Level Objectives)	Applicable Network
Lee et al. [16]	Servers	Master-slave	Data (flow)	Network management	IoT
Aazam et al. [17]	Servers	—	Context (user)	Resource Management	IoT
Jalali et al. [18]	Servers	Peer to Peer	Time (computing)	Power Management	CDN (content distribution network)

Zhu et al. [19]	Servers	-	Context (user)	Application Management	CDN
Zeng et al. [20]	Servers	Peer to Peer	Time (communication, computation)	Resource Management	IoT
Hong et al. [21]	Network Devices	Peer to Peer	Data (size)	Application Management	IoT
Nazmu deen [22]	Network Devices	Master Slave	Data (size)	Data Management	PLC (Power line communication)
Aazam et al. [23]	Network Devices	—	Data (size)	Network Management	IoT
Cirani et al. [24]	Network devices	—	Context (application)	Network management	IoT
Dsouza et al. [25]	Cloudlets	Peer to Peer	Context (application)	Network management	IoT
Cardellini et al. [26]	Cloudlets	Cluster	Context (application)	Application Management	IoT
Yan et al. [27]	Base Stations	Cluster	Context (user)	Application Management	RAN (Radio Access Network)
Gu et al. [28]	Base Stations	Peer to Peer	Cost (deployment,	Resource Management	IoT

			communication)		
Truong et al. [29]	Base Stations	Master-slave	Context (application)	Network management	Vehicular network
Oueis et al. [30]	Base Stations	Cluster	Time (deadline)	Latency Management	Mobile network
Hou et al. [31]	Vehicles	Cluster	Context (user)	Resource Management	Vehicular
Ye et al. [32]	Vehicles	–	Time (deadline)	Application Management	Vehicular
Oueis et al. [33]	Base Stations	Cluster	Data (flow)	Resource Management	Mobile network
Faruque et al. [34]	Network Devices	Cluster	Energy consumption	Power Management	IoT
Shi et al. [35]	Network Devices	Peer to Peer	Context (application)	Application Management	Mobile network
Giang et al. [36]	Network Devices	Peer to Peer	Data (flow)	Application Management	IoT
Inthara wijitr [37]	Servers	–	Time (communication, computation)	Latency Management	Mobile network
Peng et al.	Base Stations	Peer to Peer	Data (size)	Network management	RAN

[38]					
Hassan et al. [39]	Network Devices	Cluster (execution, communication)	Cost	Application Management	Mobile network
Zhang et al. [40]	Cloudlets	Peer to Peer	Cost (deployment)	Cost	LRPON (Long reach passive optical network)
Deng et al. [41]	Network Devices	Peer to Peer	Data (Size)	Application Management	Mobile network
Do et al. [42]	Network Devices	—	Energy consumption	$C i O_2$ management	CDN
Aazam et al. [43]	Servers	—	Context (user)	Resource management	IoT
Datta et al. [44]	Network Devices	Peer to Peer	Context (user)	Data Management	Vehicular network
Aazam et al. [45]	Servers	—	Context (user)	Resource Management	IoT
Gazis et al. [46]	Network Devices	—	Context (application)	Resource Management	IoT

Table 2.1: Comparison of Fog nodes

2.2 Nodal Collaboration

2.2.1 Cluster

Fog nodes are capable of forming cluster among themselves as a consequence they can sustain a collaborative execution environment. Their homogeneity [26] and location [31] can be a basis for the formation of Fog nodes. Higher priority can also be given to computational

load balancing [33] and functional sub-system development [34] during the formation of cluster.

The collaboration is potent in harnessing the capabilities of several Fog nodes concurrently. However, it is arduous to make static clusters scalable in run-time. Besides, the dynamic formation of clusters chiefly depends on the attainability and the current load of Fog nodes. In either of the cases, a crucial role is played by networking overhead.

2.2.2 Peer to Peer

Hierarchical [21] and flat order [35] can be a classification for Peer to Peer. The collaboration can also be categorized as home, local, non-local etc. [18] based on proximity. Processed output of one node is capable of appearing as input to some other node [36] as well as virtual computing instances also get shared between the nodes [25] through P2P collaboration.

2.2.3 Master-Slave

A master Fog node controls the processing load, functionalities, resource management, data flow, etc. [16] of underlying slave nodes generally.

Master-slave approach collectively with cluster and P2P based nodal interactions are capable of forming a hybrid collaborative network [22,29]. Regardless, the master and the slave Fog nodes require high bandwidth to communicate with each other owing to this type of functional decomposition.

2.3 Resource/Service Provisioning Metrics

Resource/service provisioning metrics are mentioned as follows.

2.3.1 Time

- **Computation time**

It can be adjusted per the existing load and substantially depends on the resource configuration where the task has been scheduled [20], or the application is running. It also helps in the identification of active and idle periods of various applications which are notable determinants of resource and power management [18].

- **Communication time**

It is defined as the networking delay to trade data elements. End device/sensors to Fog nodes [37] and Fog nodes to Fog nodes [20] can be the categorizations of computation time. The communication time reflects the network context, which assists the selection of suitable Fog nodes for executing tasks.

2.3.2 Deadline

It refers to the paramount delay that can be tolerated by a system in service delivery. Deadline satisfied task completion is reckoned an essential parameter for measuring QoS of the system [30,32]. Service delivery deadline plays a crucial part in portraying latency-sensitive and latency tolerant applications.

2.3.3 Data

- **Data size**

It refers to the measurement of data that has to be Fog computing processes. Numerous works have discussed data size in regards to computational space provisions of requests [35]. Besides, bulk data collected from distributed devices/sensors incorporate features of Big Data [22]. It can be efficacious to provision services and resources according to the data load in the above case. Furthermore, data size is crucial in deciding either local or remote processing of the computational tasks [38].

- **Data flow**

It describes the attributes of data transmission. Data flow can be event-driven [16] or real-time [36] and factors into associated service and asset provisioning significantly. Besides, dynamic load balancing is fostered by a sudden change in data flow [33].

2.3.4 Cost

- **Networking cost**

Associated expenses and bandwidth requirements are directly related to Fog computing environment. In various works, inter-nodal data sharing cost and cost of uploading from end devices/sensors are fundamentals of networking cost [28]. Whereas, in other literature, networking cost [39] is specified as network latency experienced due to bandwidth issues [39].

- **Deployment cost**

It is linked with the expenses associated with infrastructure placement in Fog computing environment. Some papers have reasoned that cost-effective infrastructure deployment is approving of productive service and resource provisioning. Placement of nodes in the network [40] and the creation of virtual instances [28] can be used to discuss infrastructure deployment cost.

- **Execution cost**

It is defined as the computational cost of Fog nodes while running applications or processing tasks. Despite wide usage of execution cost in billing and resource provisioning of other computing paradigms, the metric has displayed minute usage in Fog computing literature. These works have calculated the total execution cost regarding per unit time resource usage cost and task completion time [39].

2.3.5 Energy consumption & Carbon Footprints

The expenditure of energy in the energy-latency trade-off [41] and home-based Fog computing environment [34] in distinct phases of Fog-Cloud interaction have been emphasized prominently in literature. A paper mentions the carbon emission rate in terms of per unit energy usage in distinct nodes has been taken note of for resource provisioning purposes [42]. The energy aspects of end components, for example, energy characteristics of communication medium, residual battery lifetime can also be researched in provisioning Fog resources as end sensors/devices are energy-constrained.

2.4 Service Level Objectives

2.4.1 Latency Management

Latency management resists the ultimate service delivery time from exceeding an accepted threshold. Applications QoS requirement or latency of a service request can be an appropriate threshold. In some literature, the efficient initiation of nodal collaboration for the execution of tasks by collaborated nodes within the enforced latency constraints [30] is highlighted. In another paper, low-latency Fog network architecture has been suggested for handling latency [37]. Besides, in another paper, the computation task distribution between the client and Fog nodes have been conducted to reduce the communication latency of service requests and total

computation [20]. The essential purpose of this work is to select the node which contributes to the lowest latency in service delivery.

2.4.2 Cost Management

Operating Expenses (OPEX) and Capital Expenses (CAPEX) can be used to discuss cost management. Deployment of cost of distributed Fog nodes and their associated networks is the main contributor CAPEX. Optimized number and appropriate placement of Fog nodes are imperative in lessening the CAPEX in such cases. A Fog computing network architecture that optimizes the numbers and places of Fog node deployment has been proposed [40] in regards to this issue.

For launching VMs Fog nodes have been contemplated as virtual platforms in [28]. In these VMs execution of data processing operations is not free of cost. It is manageable to minimize OPEX in Fog computing by exploiting the cost-diversity in different Fog nodes as the cost of data processing operations can be varied from distinct providers. In turn, a solution to find an apt set of Fog nodes for hosting VMs has been proposed in the paper.

2.4.3 Network Management

- **Network congestion**

It generally arises due to escalating overhead on the network. Simultaneous interactions of end constituents with Cloud data centres and dispersion of IoT end sensor/devices across the edge can inflate overhead on the core network. This results in network congestion which may degrade the performance of the system. A layered architecture of Fog node that offers local processing of the service requests has been proposed taking cognizance of this into account. Outcome of this is that even while receiving bulk service requests, Clouds get a brief version of the requests.

- **Virtualization**

SDN is considered as one of the key backers of the virtualized network and offers support for NFV. SDN is a networking approach that enacts in software on separate servers after decoupling the control plane from networking gear. The architectural concept that virtualizes of traditional networking functions firewalling, (network address translation (NAT), caching, domain name service (DNS), intrusion detection, etc.) is called NFV. Several works [16, 29, 38] have suggested new network structures that enable SDN and NFV, this is due to the

broad range of services provided by SDN and NFV making them crucial in a Cloud-based environment.

2.4.4 Computation Management

- **Resource estimation**

Helps in the allocation of computational resources per some policies for appropriate resource allocation, achieving desired QoS and imposing accurate service price. Existing literature has developed resource estimation policies concerning experienced QoE, user characteristics, features of service access devices, etc. [17, 43, 45].

- **Workload allocation**

Work Allocation is for ensuring a balanced load on various components. To equalize the computational load on Fog nodes/client devices, scheduling based workload allocation policy has been introduced in [20]. Which results in overhead on both parts becoming economical and amending QoE. In another work [41] to balance delay and power expenditure, a workload allocation framework has been proposed.

2.4.5 Application Management

- **Programming platform**

To develop, compile and execute applications programming platforms offer essential components such as run-time environment, interfaces, libraries, etc. "Mobile Fog" [21] has been created to overcome the issue of proper programming support (due to the dynamic nature of Fog computing). It offers support for developing extensive applications over heterogeneous-distributed devices by simplifying abstraction of programming models. A programming platform based on distributed data flow approach has also been designed in [36] for application development, besides coordinating resources during applications execution.

- **Scaling**

The adjustment capability to retain service quality despite unpredictable events and the conception of application users. They are also applicable to user's service access and application scheduling. Architecture of a QoS aware self-adaptive scheduler to assist scalable scheduling of data stream applications has been suggested recently [26]. This scheduler is capable of scaling applications with the increase of resources and users while not requiring

global information about the environment. Furthermore, due to the self-adaptive capability of the scheduler, placement of applications in dispersed fashion and automatic reconfiguration of the resources become more manageable. An adaptive procedure for user's service access mode selection based on distance, location and QoS requirements of the service reaching entities has also been proposed [27].

2.4.6 Data Management

Data management is an imperative SLO necessary for efficient performance. Various research papers have discussed it from differing perspectives. In [23, 44] resource allocation for data pre-processing and initiation of proper data analytic services are focused on data management policy. Also, low-latency aggregation of data coming from distributed end devices/sensors can also be considered for efficient data management [22]. Besides, storage augmentation in Fog computing for preserving data of end entities can be highly effective as the storage capacity of end devices/sensors are poor. Accordingly, in [39] storage expansion in Fog computing for mobile devices has also been presented, besides application administration, as an elemental part of data management.

2.4.7 Power Management

Fog computing can be extremely convenient for offering power management as a service for various networking systems. A service platform that can facilitate power management in home-based IoT network with custom consumer control has been proposed in [34]. Moreover, it is also competent in managing the power usage of centralized Cloud data centres in specific situations. The type of running applications determines the power consumed by cloud datacentres. In this scenario, Cloud data centres can be complemented by providing infrastructure for hosting several energy-extensive applications. Owing to the above, proper power management will be ensured by reducing energy consumption in Cloud data-centres [18]. Further, the emanation of carbon footprint can also be regulated by power management [42].

The bottom sensing layer forms the basis of standard routing protocols. Failure to optimize and handle communication resources is due to the shortage of control nodes. Contrastingly, the self-organized coordinated algorithms are too complex for implementation. Communication is frequently accomplished at the expense of node energy in data transmission. Due to the demand for a large number of hops for transmissions, the reflection

to the dynamic network variation is consistently impeded. Despite that, the battery life is inadequate and re-charging is out of the question owing to the small volume of sensor nodes. The computation, storage and communication capabilities of the sensor nodes are all stifled. Accordingly, along with frailer data aggregation performance, it takes loftier execution time, with the likelihood of more errors [50].

Formations of clusters are important in communication in wireless sensors network. The data should travel to optimal nodes and should be transferred through those paths using which maximum efficiency can be achieved. This selection of optimal path is routing. Routing can be performed using various software programmed devices known as protocols. Thus, selection of optimal protocols is very crucial for network optimization and efficiency.

Routing algorithms have two types, that is non uniform and uniform clustering routing algorithms. In a previous work [88], a non-uniform clustering routing algorithm was proposed. The clusters, closer to the sink node, are smaller, and the number of sensor nodes is less. The clusters away from the sink node, are larger and the number of sensor nodes is also more. In this way, the energy consumption closer to the sink is greatly reduced. A method based on fog computing was presented by the authors of [89].

These researchers stored some data collected from the sensor nodes in the fog layer to avoid interruption in between communication. This method also makes the movement of sink node easy. In another work [90], an algorithm, which was used to calculate the neighboring node's location, was proposed. Filter based framework known as FERA was used. In this study, simulation was used to observe the performance of the system. In [91] the amount of data that is transmitted is decreased, which directly affects the lifetime of a network as the sensor nodes in WSN consume maximum energy during transmission and reception of data packets. Author of [92] presented a novel algorithm termed as complex alliance strategy with multi objective optimization of coverage. A proportional relationship of the energy conversion function between the working node and its neighboring nodes was also presented in this study.

A self-adaptive and self-organized protocol of routing algorithm LEACH was promoted by various authors. Many of them worked on the betterment of protocol and some improved the LEACH algorithm significantly [93]. Author of [94] exercise on forming cluster heads with balanced energy by selecting two cluster head depending upon the energies and distance

between head and base station. Author of [95] experienced that optimistic distribution of computing load shows better results as compared to the blind distribution.

In this model clustering and routing was done using Fuzzy Multiple Attribute Decision Making technique. A routing scheme known as trust-based secure routing was proposed in a previous study [96]. The source nodes send data and notification to sinks through disjoint paths, separately so that the data and notification can be verified independently to ensure their security. This prevents malicious attacks on the network and makes it more secure. Author of [87] proposed clustering method based on mobile intelligent fog computing. This method is also known as energy-efficient cross- layer-sensing clustering method (ECCM). The routing protocol is also optimized using particle swarm optimization (PSO).

All the previous proposed routing algorithms have succeeded in transmitting the data effectively by choosing optimal routes but all models also have limitations. High complexity and lack of energy efficiency are still one of the major drawbacks of the available algorithms. The proposed algorithm uses fog paradigms for optimizing the network. It introduces a fog layer for storing the data from the sensors. The method is also optimized using various algorithms like Ant lion optimization (ASO) and k - means clustering which enhances the efficiency of the network thus improving the lifetime of the network.

In the era of IOT, importance of wireless sensor networks has increased. Energy consumption in communication is dependent on the cluster formation. Selection of optimal routes and nodes should be done in order to reduce energy consumption and latency. At present there are many techniques available for cluster formation.

Over the years numerous algorithms were introduced to optimize cluster routes. Routing algorithms can be classified into uniform and non-uniform clustering algorithms. Authors of [88] proposed non uniform cluster routing algorithm. The objective of this algorithm was to balance energy consumption in the nodes. In study [89] deployment scheme was proposed which reduced long hops in cluster into multiple small hops to reduce transmission time in the wireless sensor networks. Authors of [90] proposed a model based on fog layer computing. This model stores partial information of the nodes in advance in the fog layer, this reduces the interruptions between the communication and was helpful in moving sink node. A low-cost collaborative charging in wireless sensor networks technique was presented in study [91].

In [92] a data dissemination strategy which was based on ant colony optimization techniques was proposed. Authors named this strategy as transmission with multiple load balancing schemes (TMLBS). Load decentralization, load maintenance and load diversion schemes were used in this proposed model. Researchers of [93] proposed a relative low-cost filter-based framework FERA. This model adaptively used push/pull strategies and was able to handle data request in VANET effectively. Authors of [94] proposed DMOA which was motivated by morphological erosion and dilation on binary images. This algorithm was capable of extracting event backbone nodes and estimates the event region on their basis.

A novel algorithm named as complex alliance strategy with multi-objective optimization of coverage (CASMOC) which improves the congestion in network was proposed in [95]. Authors of this study also provided a proportional relation between working node and neighbouring nodes for energy conversion.

Due to the uncertainty of data, there could be coverage blind area and redundant data. To deal with these authors of [96] proposed a novel model named as Coverage Control Algorithm for Moving Target Nodes based on Sensing Probability (CMTN-SP). Survivability aware connectivity restoration strategy was proposed in study[97].

This strategy involves the design of load equilibrium mechanism and reliability enhancement measure. In study [98] a novel algorithm called energy-based k-coverage control algorithm (EBKCCA) was proposed. Based on the calculations by this algorithm authors concluded that multi transmission consumes less energy than single transmission.

In study [99] Energy efficient method for clustering is proposed. Firstly, a strong node is constructed in fog layer then data aggregation routing is constructed and at last particle swarm optimization is used to elect cluster head for effective routing.

Above discussed algorithms have shown promising results but there is still some room for advancement. Presently available algorithms are highly complex in nature. Apart from that energy efficiency is always a reason of concern.

Article	Ideas	Improvement Criteria	Limitation
Deng et al. [41]	Nonlinear integers, Hungarian Method	Power Consumption, Delay	Suitable for centralized infrastructure, not easy to apply to fog computing
Guo et al. [51]	Markov decision problems	Power consumption, Delay	A continuous time queuing system
Queis et al. [33]	Heuristic	User satisfaction, Task Latency, Power consumption	Complexity when number of user requests increases.

Table 2.2: Comparison of Power based optimization.

2.5 Security Concerns

2.5.1 Authentication

An essential feature of invasion defence in Fog based services is user authentication. Unauthorised access in Fog services is unacceptable as they are employed in "pay as you go" basis. Secured Fog computing environment requires user authentication, device authentication, instance authentication, data migration authentication [25].

2.5.2 Privacy

Data from end devices/sensors is processed in Fog computing. In certain situations, the data closely associated with user activity and situation is detected. Hence, proper assurance of privacy is considered a vital question in Fog computing. The privacy issues concerning Fog based vehicular computing have been focussed upon for further examination, in [31].

2.5.3 Encryption

Cloud computing is complemented by Fog computing. The data processed has to be forwarded to Cloud in particular situations. It is indispensable to encrypt the data in Fog nodes to protect sensitive information. A data encryption layer has been proposed in [23] to take this into account.

2.5.4 DoS Attack

It is challenging for Fog nodes to manage an extensive number of simultaneous requests as they are resource constrained. Therefore, they can experience performance degradation. Denial-of-Service (DoS) attacks play a critical role in creating severe service disruptions. Fog nodes can be made busy for long durations of time by making a lot of unnecessary service requests concurrently. Accordingly, resources for executing useful services become inaccessible. This kind of DoS attack in Fog computing has been suitably examined in [24].

2.6 Research Gaps

Following research gaps have been identified based on the literature review done in the previous section

- 1) To obtain an extra bit of battery life for fog devices has been one of the greatest difficulties of fog computing solutions.
- 2) Researchers have used various optimization techniques for FOG Nodes but very few researchers have used Ensemble model approach followed by an optimization; proposed work uses this as a Research Gap.
- 3) Aggregation of Fog nodes by migrating tasks from one node to another node can be constructive in certain scenarios so as to manage power in a Fog environment.
- 4) The framework of Fog is distinct from Cloud so advancement of associated programming languages to enable Cloud-based services and existing standards in Fog are necessary.
- 5) Development of user interfaces and efficient networking protocols are necessary for managing a large number of connections in Fog.
- 6) fog nodes need improvement on availability, scalability, security, performance and load balancing.

2.7 Problem Statement

Industrial operators use the Industrial Internet of Things (IIoT) to automate the systems, deploy sensors to monitor, measure, analyse data increase revenue opportunities and improve efficiencies for manufacturing operations in wireless sensors network (WSN) has let the manufacturing industry is experiencing substantial benefits. The quantity of data from these plants can be computed in petabytes (1 million gigabytes) because of millions of connected sensors on industrial control systems (ICSs), videos surveillance cameras covering plants, millions of streaming, industrial robots, autonomous drones and so on.

Operational technology environments approached by traditional information technology cannot meet the reliability security, network bandwidth difficulties, privacy and necessary volume in a controlled operational environment. Hence, Fog environment is essential in Wireless Sensors Networks (WSNs) to allow IIoT to reach its potential.

The approved data-based routing protocols for connecting Fog nodes are ordinarily vulnerable to a huge quantity of energy and power voids in Wireless Sensors Networks. In the problem formulation of this PhD research proposal, research gap # 1, 2 and 3 is considered. As per research gap 1, 2 and 3, there is need to create Power-efficient framework for Fog environment which can handle, migration of operators, dynamic load in the network and migration of tasks.

2.8 Objectives

The broad objectives of this PhD research work are

- 1) To study and analyse existing Power Efficient resource utilization techniques in fog environment.
- 2) To propose novel power efficient framework for resource utilization in fog environment.
- 3) To implement and validate the proposed framework.

2.9 Methodology

The objectives are fulfilled by using the following methodology

1. The primary aim of this research work is to study the present available power efficient algorithms in fog environment.
2. Based upon Fog resources like power consumption and QoS parameters a novel resource utilization framework for Fog resources is proposed.
3. Framework is proposed after comparison with various other existing techniques.
4. Proposed framework is implemented using in MATLAB 9.4, 2020a.
5. Suitable architecture of sensor networks is used for the implementation in Fog Environment.
6. The architecture of wireless sensor network is developed for Fog nodes.
7. The developed architecture is integrated with internet through specific protocols for sharing of information and communications.
8. Further, simulations is performed to validate the suggested power efficient framework for Fog environment and on the basis of post effects of proposed framework, a sustainable solution is given.

2.10 Thesis Contributions

- 1) A comprehensive survey of various Power Efficient frameworks is presented in this work. This will help the researchers to be able to understand different Power Efficient techniques.
- 2) The proposed model *Power-Efficient Optimized Clustering Method (PEOCM) with Intelligent Fog Computing for WSNs* introduces a fog layer in WSN consisting of cluster heads in the network. Sensing layer is optimized using ant lion optimization (ALO), and the wireless sensors in fog layer are clustered using K- means clustering. The proposed algorithm focuses on ALO only on the bottom layer and not on the fog layer. To evaluate the performance of the proposed algorithm, the results are compared with the findings of several traditional algorithms, such as energy-efficient cross-layer-sensing clustering method, Distributed and Morphological Operation-based Data Collection Algorithm (DMOA), and trust-based secure routing. Comparative results showed that the proposed algorithm presented significant improvement.
- 3) The proposed model *Machine Learning Based Power Efficient Optimized Communication Ensemble Model with Intelligent Fog Computing for WSNs* is based on ant lion optimization which helps in initial cluster formation.
- 4) Further an ensemble model of k-means, SOM and fuzzy c-means is used to divide clusters in FOG Nodes using weighted means algorithm.
- 5) Researchers have used various optimization techniques for FOG Nodes but a very less researchers have used Ensemble model approach followed by an optimization; proposed work uses this as a Research Gap.

3. Power-Efficient Optimized Clustering Method (PEOCM) with Intelligent Fog Computing for WSNs

3.1 Introduction

In wireless sensor networks (WSNs), optimizing energy consumption during data packet transmission is a crucial characteristic. To achieve this, routing algorithms must provide efficient solutions that minimize energy consumption. The routing protocols are required whenever the source needs to send packets to the destination [181]. In this context, the end devices layer of WSNs involves the clustering of sensor nodes using the ant lion optimization (ALO) technique, followed by the fog layer that employs the k-means clustering method. However, the accuracy of nature-inspired optimization solutions primarily depends on the complexity and number of parameters involved. The algorithm proposed in this study focuses only on the bottom layer and does not consider the fog layer. The proposed algorithm significantly outperforms conventional algorithms used for the same purpose. To assess its performance, we compared its results with those of several traditional algorithms, such as the energy-efficient cross-layer-sensing clustering method, Distributed and Morphological Operation-based Data Collection Algorithm (DMOA), and trust-based secure routing. Our comparative analysis demonstrated that the proposed algorithm leads to a significant improvement in performance.

Parameter	Value
Number of nodes	800
ε_{fs} (short distance energy consumption in WSNs)	10 pj/bit/m ²
ε_{mp} (long distance energy consumption in WSNs)	0.0013pj/bit/m ²
k (packet size)	50 (header)+1000 (data) bits
E_e	0.1 j
E_{elec}	5 nj/bit

Table 3.1 Proposed network simulation parameters

3.2 Material and Methods

3.2.1 Evaluation Parameters and Assumptions

In this study, we propose a stimulation algorithm that employs 800 sensor nodes randomly and evenly distributed within a 500 m x 500 m range. Data is transmitted in the form of data packets consisting of two parts, namely the header and data frame. The header size is 50 bits, and the data frame size is 1000 bits. To enable comparison with previous techniques, we assume that all sensor nodes are homogeneous, have the same power of transmission, and have a unique ID. We vary the node density for different simulations, denoted by λ , which represents the number of nodes per 10 m². We also assume that the sensor nodes are static, and only the sink node can move within the simulation area regardless of the power constraints. Each node can sense events, perform data aggregation, and route the same data according to the defined protocol. Figures 3.1(a) - (b) illustrate the networks with varying λ values.

3.2.2 Energy Calculations in Data Routing

The proposed algorithm employs a data clustering-based routing approach with a driven strategy. In cluster routing protocols, clusters consist of several cluster members (CM) and a cluster head (CH). The data is communicated within the cluster from CMs to CH in a single hop. However, in cases where the distance exceeds certain threshold values, inter cluster communication is achieved using multi-hop communication. Intra-cluster routing can be performed in a single hop or multi-hop routing and is also known as cross-layer communication. In most cluster-based WSNs, the CH determines the routing strategy and controls the CMs. Sensor node energy requirements are categorized into three parts energy consumed in sensing events and processing to build the complete data frame, energy consumed in data communication, and energy consumption when a node is in an idle or sleep position. Maximum energy consumption occurs when a node communicates with other nodes using radio frequencies. Equations 1-3 describe the energy consumption in data communication, following Heintzelman's first-order communication radio model.

$$E_{Tx}(k, d) = \begin{cases} k * E_{elec} + k * \varepsilon_{fs} * d^2 & d < d_0 \\ k * E_{elec} + k * \varepsilon_{mp} * d^4 & d \geq d_0 \end{cases} \quad (1)$$

$$E_{Rx} = k * E_{elec}, \quad (2)$$

$$d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}}, \quad (3)$$

where E_{Tx} is the energy required to transmit k bits over a distance d ; E_{Rx} is the energy required to receive k bits; irrespective of the distance; and ε_{fs} and ε_{mp} represent the constant values for free space and multifading channel path, respectively.

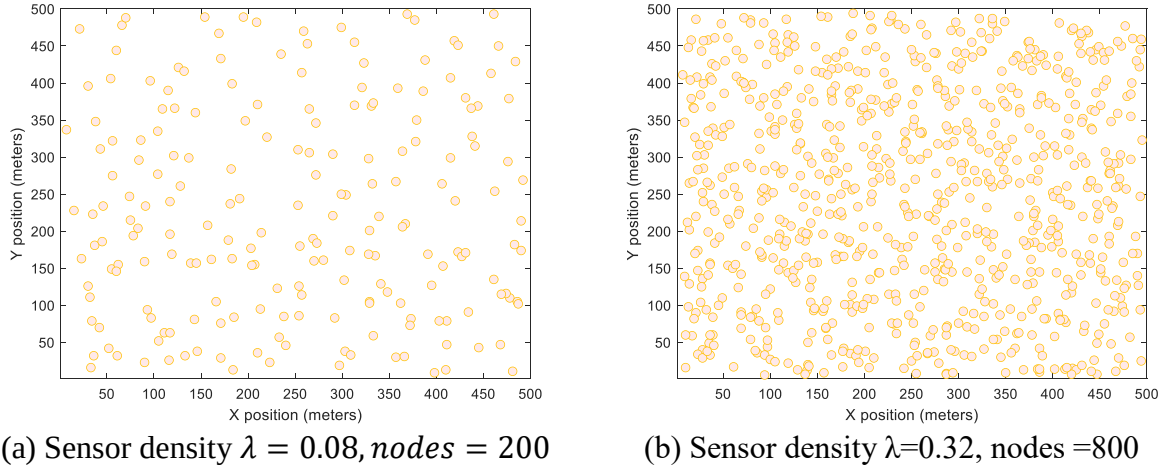


Figure 3.1: (a–b) Representation of Sensor Networks with varying node density λ

Equations 1-2 indicate that communication is more efficient when the distance is less than d_0 , otherwise, energy consumption will increase. In cases where the distance exceeds d_0 , multihop communication is preferable over single hop communication. Equation 3 defines the distance threshold value d_0 , which is 87.7m in this study. Table 3.1 presents the constant parameters utilized in calculating data transmission and reception energies.

3.2.3 Introduction to ALO

The ALO is a nature-inspired optimization model based on the hunting behaviour of ant lions. Many species of ants have foraging behavior[176]. Similar to other optimization techniques, it includes five main steps, which involve random walks of ants lions, construction of traps, entrapment of ants into the traps, catching prey, and trap rebuilding. The algorithm optimizes the cluster members (CMs) and cluster heads (CHs) using these steps. The random walks of ants are performed according to Equation 4, and the ant locations are updated with every iteration, using normalized values within the scale of [0,1].

$$X_j^{it} = \frac{(x_j^{it} - a_j) \times (d_j - c_j^{it})}{(d_j^{it} - a_j)} + d_j, \quad (4)$$

where a_j represents the minimum value of the random walk by ants of the j^{th} variable, b_j represents the maximum value of the random walk by the ants of the j^{th} variable in it^{th} iteration, it represents the it^{th} iteration, and d_j^{it} denotes the maximum value of the j^{th} variable in it iteration.

$$C_j^{it} = Antlion^{it} + C^{it}, \quad (5)$$

$$d_j^{it} = Antlion^{it} + d^{it}, \quad (6)$$

Where $Antlion^{it}$ represents j^{th} antlion in it^{th} iteration. The movement of antlion is in a hypersphere as defined by Equations 5–6.

Once ants are trapped in the ant lion pit, the ant lion continually throws sand toward the ant to make it slide, thereby keeping the ant inside the pit. This behavior of the ant lion decreases the pit's diameter. Equations 7–8 are used to update the pit area by updating variables C^{it} and d^{it} .

$$C^{it} = \frac{c^{it}}{I}, \quad (7)$$

$$d^{it} = \frac{d^{it}}{I}, \quad (8)$$

where I represents ratio, $I = 10^{\omega \frac{t}{T}}$, where t denotes the current iteration, T represents the maximum iteration taken for this study, and ω is a constant following the constraints given below

$$\begin{cases} \omega = 2 \text{ when } t > 0.10T \\ \omega = 3 \text{ when } t > 0.50T \\ \omega = 4 \text{ when } t > 0.75T \\ \omega = 5 \text{ when } t > 0.90T \\ \omega = 6 \text{ when } t > 0.95T \end{cases}$$

Pseudo code for ALO used in proposed algorithm

1. Initiates population includes Ants & Antlions randomly
 2. Fitness value evaluation for Ant lions
 3. Calculates global best (fittest Antlion and assume it elite)
 4. **while**! (termination condition)
 5. **for** $i = 1 \rightarrow$ No. of Ants
 6. Select Antlion with Roulette wheel selection
-

7. Updating variables c & d
 8. Random walk followed by normalization
 9. Updating position of Ants
 10. **end for**
 11. fitness value calculation of Antlions
 12. Evaluation process and replacement of weak solution by fit solutions
 13. **end while**
-

3.3 Proposed Network Model

3.3.1 Optimization in Cluster Formation

The ALO optimization technique is used in the cluster formation phase of the proposed algorithm. ALO is used after each round of communication to determine the optimal CMs and CHs. The percentage of clusters formed varies between 2% and 5% of the total number of sensor nodes used. The proposed algorithm uses single hop communication for intercluster data transfer between CMs and CHs, and cross-layer communication supports both single hop and multihop communication depending on the distance. ALO predicts the number of clusters based on a fitness function that considers remaining energy, node density, and distance to the nearest node. In each round of communication, candidate nodes for CH are selected from a pool of sensors. ALO is only used in the bottom layer for cluster formation.

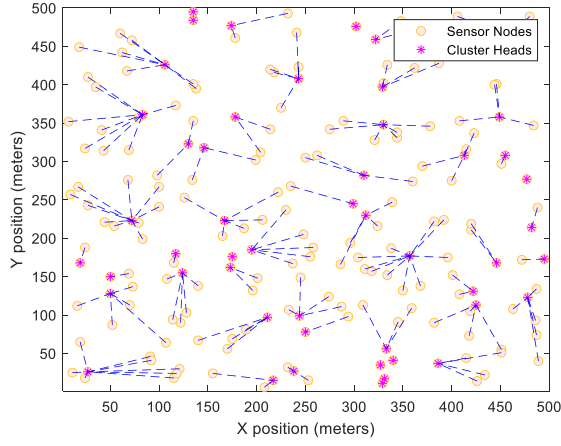
The problem of finding the optimal solution for M clusters with K CHs results in M×K possible solutions. ALO solves this problem efficiently. The proposed algorithm uses a fitness function, as shown in Equation 9, to determine the optimal CMs and CHs. Figures 3(a)–(b) illustrate examples of optimized CMs and CHs.

$$\text{fitness function} = \min\{E_{ln} + E_{for}\}, \quad (9)$$

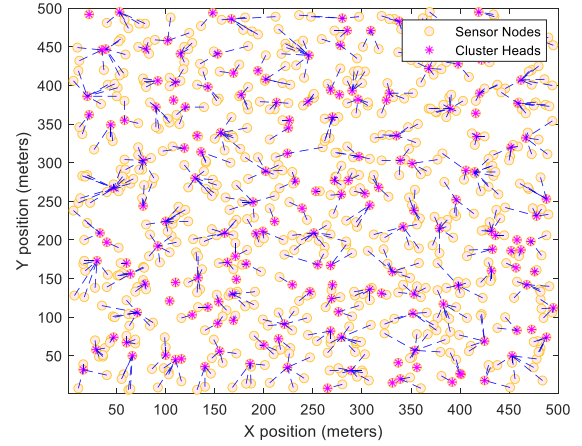
$$E_{ln} = \sum_{k=1}^{k=m} E_{Tx}(k, CH_k), \quad (10)$$

where E_{ln} denotes data communication of k^{th} node to its respective CH_k k^{th} CH with a single hopping protocol & $CFOG_k$ are fog cluster.

$$E_{for} = \sum_{k=1}^{k=M} E_{Tx}(k, CFOG_k), \quad (11)$$



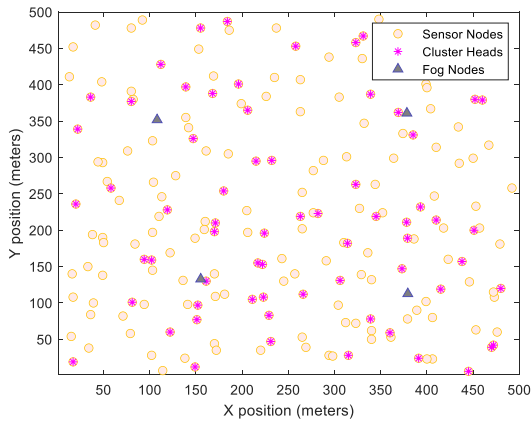
(a) Sensor nodes and cluster heads when nodes=200



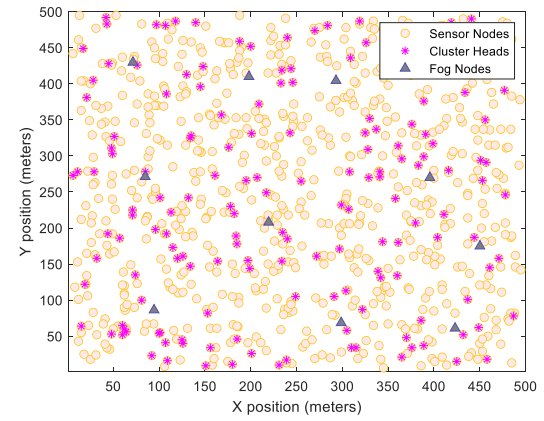
(b) Sensor nodes and cluster heads when nodes=800

Figure 3.2: (a–b) Representation of predicted optimized cluster members and cluster heads.

Dashed lines show one hop communication between members and cluster heads



(a) Sensor nodes, cluster heads, and fog nodes when nodes=200



(b) Sensor nodes, cluster heads, and fog nodes when nodes=800

Figure 3.3 (a–b) Examples of predicted Fog Nodes and Cluster Heads using FCH k-means++.

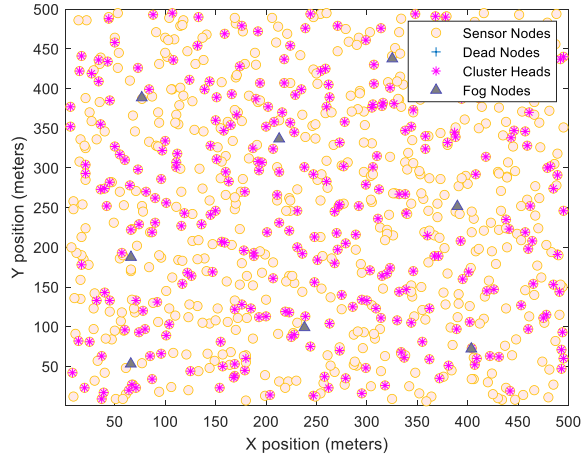
where E_{for} represents the energy used in the transmission of the aggregated data of the k^{th} fog node considering single hop or multihop communication, m denotes the number of sensor nodes $[0.02 \leq m < 0.05m]$, and M represents the number of clusters.

$$D_{sum}^j = \sum_{i=1}^{f_m} \sqrt{(Fmx_i - Fmx_j)^2 + (Fmy_i - Fmy_j)^2}, \quad (12)$$

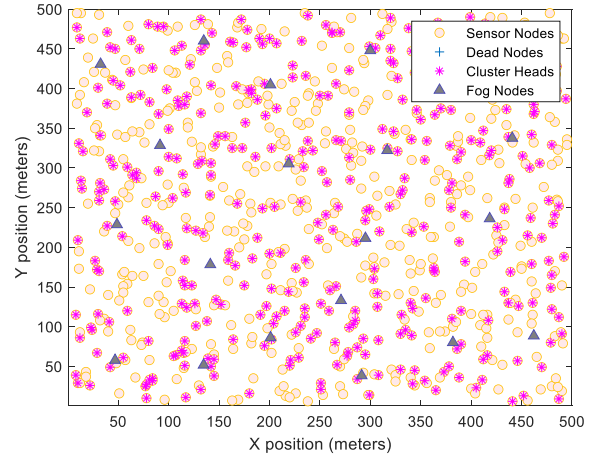
where f_m represents the number of fog members in a j^{th} fog cluster, $[Fmx_i, Fmy_i]$ denotes the position vectors of the i^{th} fog node, and D_{sum}^j represents the average distance of the fog node with its FM of the several FCs.

3.3.2 Introducing Fog Layer using k-means

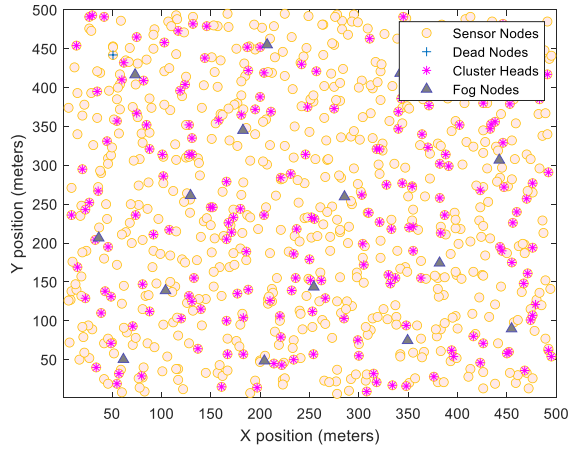
At this point, the optimized CH selections have been made, and the CHs have started communication with their respective CMs to collect data. Once a CH receives data from all its active CMs, data aggregation is performed. The next step is to transmit the data from the CHs to their corresponding fog node. To achieve this, the proposed algorithm utilizes k-means optimization. K-means is applied to determine the best possible fog clusters, represented by FCH, which includes only CHs as members. The number of FCH varies from 3% to 5% of cluster heads. Each fog cluster then selects its fog node, denoted by FN. Equation 12 shows the calculation of D_{avg} for every fog member. Fog members having minimum D_{avg} will be elected as the FN. Thus, every CH has to send its aggregated data to its nearest FN. Communication between CH and FN can have single hop or multihop transmission depending upon energy consumption. Figures 3.3 (a)–(b) show the examples of the FCH that is predicted using k-means++.



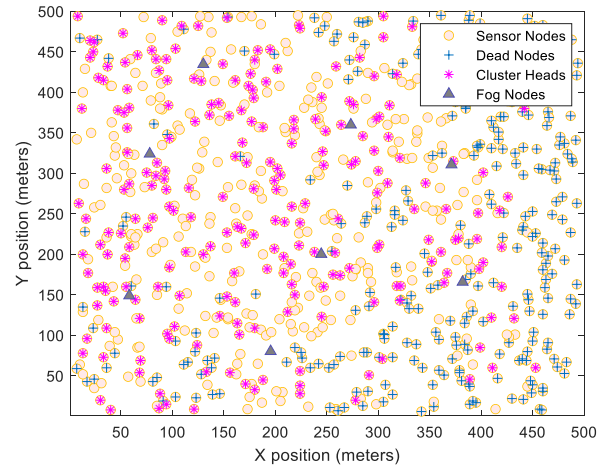
(a) Distribution of various nodes with simulation time $t=500s$.



(b) Distribution of various nodes with simulation time $t=1000s$.



(c) Distribution of various nodes with simulation time $t=1500s$.



(d) Distribution of various nodes with simulation time $t=2000s$.

Figure 3.4: (a–d) Dead and fog node positions at different simulation times

3.4 Results and Discussions

The proposed algorithm is implemented and evaluated using MATLAB9.4. To compare its performance with previous studies [29], [35], [36], several evaluation parameters are used, including the residual energy of the nodes versus running time and network lifetime versus the number of sensors. The simulation area is set to 500 m×500 m, and different numbers of sensor nodes (ranging from 100 to 800) are randomly deployed. The number of clusters, which is controlled by a random variable, varies between 2% to 5% of the total number of

sensor nodes. The data packet consists of a 50-bit header and 1000-bit information. The sensing radius is set to 10 m, and the initial energy of every sensor node is 5J.

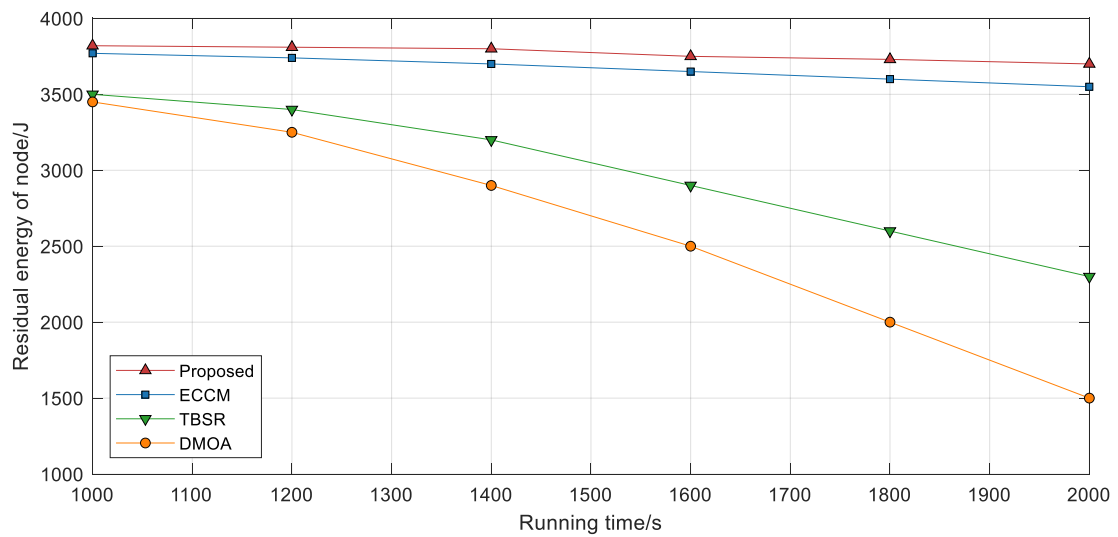


Figure 3.5: Residual energy of nodes vs running time, Network Energy consumption over time for State of Art Methods

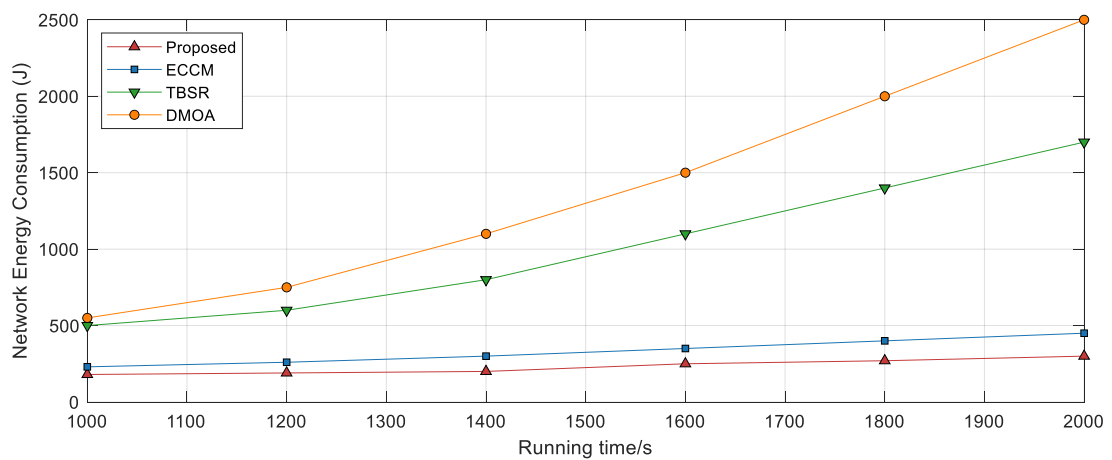


Figure 3.6: Network energy of nodes vs running time, Network Energy consumption over time for State of Art Methods

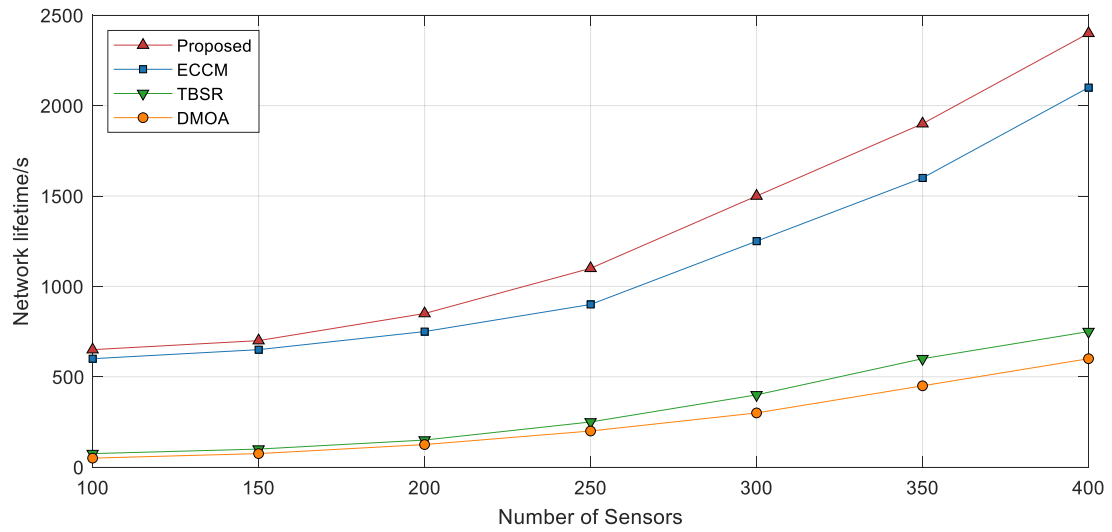


Figure 3.7: Network lifetime over Number of Sensor nodes for proposed algorithm compared with various State of Art Methods

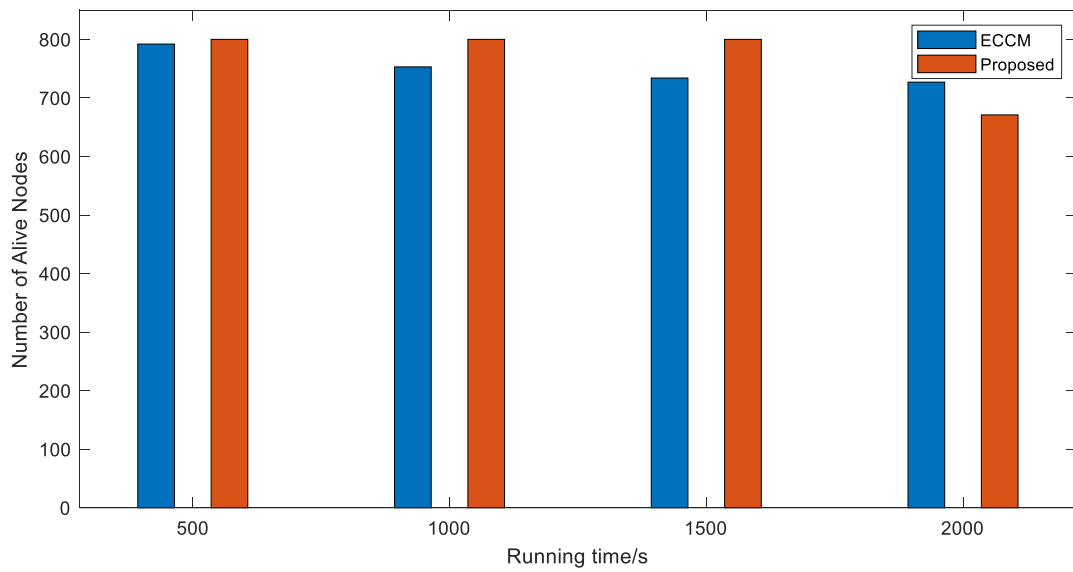


Figure 3.8: Number of alive nodes vs running time. Proposed algorithm shows the Network stability as compared to state of Art methods

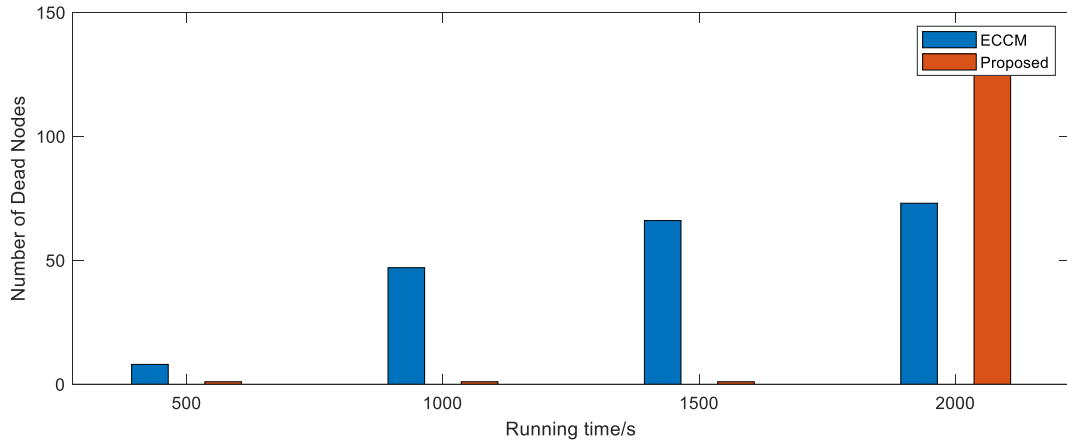


Figure 3.9: Number of Dead nodes vs running time. Proposed algorithm shows the Network stability as compared to state of Art methods

ALO optimization is utilized in the proposed algorithm with 10 iterations and 50 ant lion populations for every round of communication. The positions of dead and fog nodes at different simulation times, namely $t=100s$, $500s$, $1000s$, and $2000s$, are shown in Figures 3.4 (a)-(d). The number of dead nodes is 0, 0, 0, and 129 at 100, 500, 1000, and 2000 time/s, respectively. The proposed algorithm shows load balancing in the network, with no dead nodes up to 2200s of simulation time, in contrast to the traditional ECCM algorithm, which has several dead nodes even at a simulation time of 100s. Figure 3.5 (a)-(b) compares the residual energy of nodes and network energy consumption with running time, showing the stability of residual energy in the proposed algorithm over time compared to ECCM. Network lifetime versus the number of sensor nodes is shown in Figure 3.6, indicating a 14.29% improvement in network lifetime compared to the ECCM protocol. Another important parameter for comparison with state-of-the-art methods is the number of alive/dead nodes over time, and the proposed algorithm performs better in this regard, as shown in Figures 3.7, 3.8 & 3.9.

4. Machine Learning Based Power Efficient Optimized Communication Ensemble Model with Intelligent Fog Computing for WSNs

4.1 Introduction

In recent years, wireless sensor networks (WSNs) have advanced significantly, with energy consumption being a crucial performance parameter. This chapter proposes a new algorithm that uses Ant Lion Optimization (ALO) and three different clustering techniques to improve WSN performance. The proposed algorithm is inspired by fog computing, and it uses ALO for cluster formation in the bottom layer, while an ensemble model of k-means, SOM, and Fuzzy c-means is used for clustering in the FOG layer. The proposed algorithm is compared to traditional methods to evaluate its performance, and it has demonstrated a significant reduction in energy consumption compared to the ECCM protocol. Additionally, the proposed algorithm has shown improvement in terms of load balancing, which is an important parameter in WSNs.

Wireless sensor networks have developed recently as a result of IOT technology advancements. A dedicated network of wirelessly connected sensors is what is known as a wireless sensor network. WSNs have been introduced in a number of industries over the years, including healthcare [51], [52], automobile [53], home and office [54], and others [55]. WSNs are essential for implementing IOT. Energy consumption and load balancing are two factors that can be used to assess WSN performance. The routing mechanism affects energy consumption [56]–[58]. Currently, a variety of models are available to optimise the path that information takes from sensor nodes to the hub [59]–[63]. Data transfer from one end to the other is the essence of communication, and the routing mechanism is a phenomenon that determines the path that data takes. Routing can involve one hop or many hops, and each hop uses energy [64] through [67]. Energy management [68]–[71] is crucial because replacing or recharging a battery is typically not an option [22]–[74]. The models that are currently on the market are more complex and use more energy. This study suggests a model that balances loads more effectively and uses less energy.

- Proposed model is based on ant lion optimization which helps in initial cluster formation.

- Further an ensemble model of k-means, SOM and fuzzy c-means is used to divide clusters in FOG Nodes using weighted means algorithm.
- Researchers have used various optimization techniques for FOG Nodes but a very less researchers have used Ensemble model approach followed by an optimization, proposed work uses this as a Research Gap.
- Results shows significant difference in network life lime of sensor nodes when compared to existing techniques.

This study is further divided into following sections

- Related work
- Material and Methods
- Performance evaluation
- Conclusion

4.2 Related Works

In the age of IoT, wireless sensor networks have gained significant importance. Energy consumption in communication depends on cluster formation, and optimal selection of routes and nodes can reduce energy consumption and latency. Currently, many techniques are available for cluster formation. Various algorithms have been proposed to optimize cluster routes. Routing algorithms can be categorized as uniform and non-uniform clustering algorithms. In [88], the authors proposed a non-uniform cluster routing algorithm to balance energy consumption in the nodes. The deployment scheme proposed in [89] reduced long hops in clusters to multiple small hops, which reduced transmission time in wireless sensor networks. In [90], the authors proposed a model based on fog layer computing that stored partial information of the nodes in advance in the fog layer, reducing interruptions between communication and facilitating the movement of sink nodes. In [91], a low-cost collaborative charging technique was presented for wireless sensor networks. In [92], the authors proposed a data dissemination strategy based on ant colony optimization techniques called Transmission with Multiple Load Balancing Schemes (TMLBS), which used load decentralization, load maintenance, and load diversion schemes. The authors of [93] proposed a relatively low-cost filter-based framework called FERA, which adaptively used push/pull strategies to handle data requests in VANET effectively. In [94], the authors proposed DMOA, which was inspired by morphological erosion and dilation on binary images and capable of extracting event backbone nodes and estimating the event region. In [95], the

authors proposed a novel algorithm called Complex Alliance Strategy with Multi-Objective Optimization of Coverage (CASMOOC) to improve congestion in the network and provided a proportional relation between working node and neighbouring nodes for energy conversion. To deal with coverage blind areas and redundant data due to data uncertainty, the authors of [96] proposed a novel model called the Coverage Control Algorithm for Moving Target Nodes based on Sensing Probability (CMTN-SP). A survivability-aware connectivity restoration strategy was proposed in [97], which involved the design of a load equilibrium mechanism and reliability enhancement measure. In [98], the authors proposed the Energy-Based K-Coverage Control Algorithm (EBKCCA), which showed that multi-transmission consumes less energy than single transmission. In [99], an energy-efficient method for clustering was proposed. First, a strong node was constructed in the fog layer, then data aggregation routing was constructed, and lastly, particle swarm optimization was used to elect cluster heads for effective routing. Although the above-discussed algorithms have shown promising results, there is still room for advancement. Presently available algorithms are highly complex, and energy efficiency remains a concern.

4.3 Material and Methods

This section presents the energy consumption calculation and assumptions made for simulations. Figure 3.1 illustrates the architecture of the wireless sensor network (WSN) which is divided into three layers – cloud layer, fog layer, and sensing layer. This study focuses on the fog layer and sensing layer. The fog layer is a virtual layer that connects the sensing layer to the cloud storage layer and is responsible for clustering the sensing nodes, modulating the routes of data transfer, and selecting cluster heads. The sensing layer is the physical layer where sensors are deployed and data is collected, which is then sent to the sink node using the optimized route.

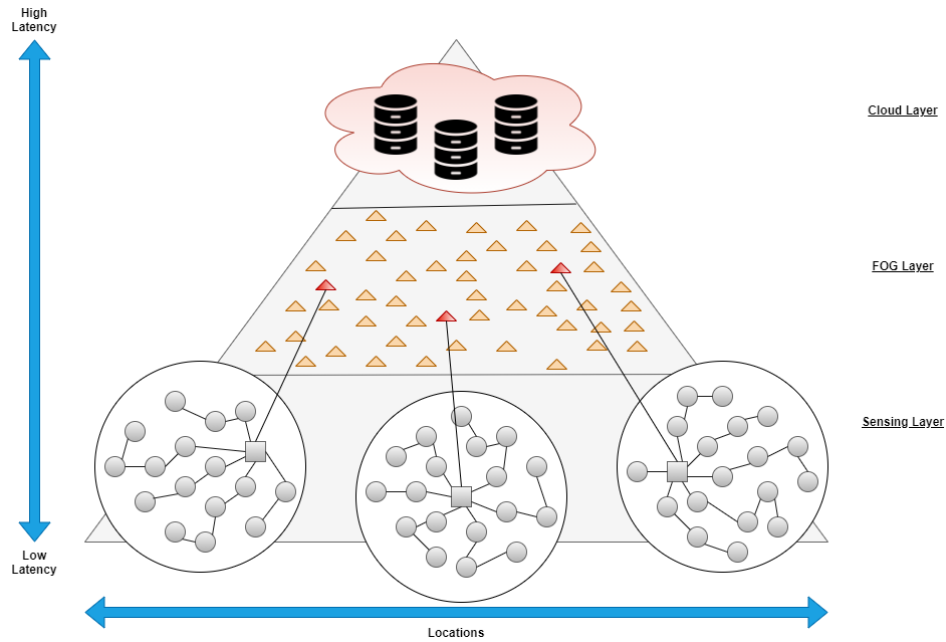


Figure 4.1: Architecture of proposed model

Proposed Table 4.1 has mathematical and scientific notations and their meaning which are used in the following sections to illustrate various calculations and methods.

Notations	Meaning	Notations	Meaning
E_T	Energy consumed for transmission	WSN	Wireless Sensor Network
d	Distance	IOT	Internet of Things
it	Iteration	ALO	Ant Lion Optimization
CH	Cluster Head	λ	Node Density
c	centroid	SOM	Self-Organizing Maps
μ	Membership degree	FCM	Fuzzy C-Means

Table 4.1: Notations and their meanings

4.3.1 Energy consumptions at Various Levels in WSN Networks

The proposed algorithm utilizes a clustering mechanism to achieve optimized routing. Each cluster comprises of cluster members and a cluster head. The data is transmitted from cluster members to cluster head in either single or multiple hops, depending on their distance from the nearest cluster head. Energy is consumed during three phases sensing events, communication, and idle state. Communication phase consumes the maximum amount of energy. The energy consumption during the communication process is illustrated in Figure 4.2.

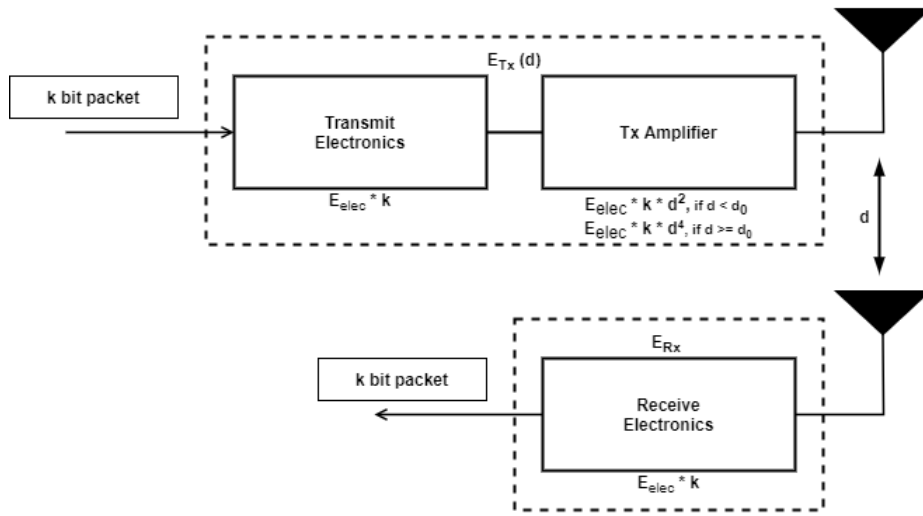


Figure 4.2: Communication Process

Equations (1-3) show the energy consumption during communication. Presented model is based on Heinzelman's [37] first-order communication radio model.

where E_{Tx} is the energy consumed for transmitting k bits over a distance d_j ; E_{Rx} is the energy consumed for receiving k bits which does not depend on distance; d_0 is the threshold distance; and ϵ_{fs} and ϵ_{mp} are the constant values for free space and multi-fading channel path, respectively.

4.3.2 Proposed Network Model

Proposed model is based on fog computing. In this study optimization technique along with various clustering techniques are used which are explained in the sub-sections below

4.3.2.1 Optimization using ALO in Cluster Formation

The ant lion optimization is a type of nature-inspired optimization technique based on the hunting method of the ant lion. The ant lion first walks randomly within a certain range and builds traps for ants. Once the ants are trapped, the ant lion performs calculations based on them, and this process is repeated for each iteration. In order to calculate the random walks of the ant lion, Equation 5 is used. After each iteration, the location of the ant lion is updated, and the locations are normalized.

Where a_j and b_j are the minimum and maximum value of random walk of j^{th} ant in it^{th} iteration and d_j^{it} denotes the maximum value of the j^{th} ant in it^{th} iteration.

The Ant Lion Optimization (ALO) algorithm is employed for cluster formation and optimization of both cluster members and cluster heads. Intra-cluster communication uses single hop while inter-cluster communication considers both single and multi-hop. A fitness function is defined to evaluate the performance of the clusters, which depends on factors such as remaining energy, node density, and nearest node distance. ALO is applied solely in the bottom layer for cluster formation and optimization. This approach has proven to be an efficient and speedy optimization technique for this specific task.

where E_{ln} is the transmission energy of k^{th} node to its respective $CH_k k^{th}$ cluster head with a single hopping protocol, m is the number of nodes and M is the number of clusters

E_{for} is the transmission energy of the k^{th} fog node considering single hop or multi-hop communication, m is the number of nodes and M is the number of clusters.

4.3.2.2 Fog Clustering with k-means

The study proposes an optimization technique for cluster formation and data compression in a wireless sensor network using Ant Lion Optimization (ALO). The optimized cluster heads gather data from the cluster members and compress it before transmitting it to cloud storage. In recent years, researchers have promoted the use of intelligent data forwarding schemes, which utilize Fog layers to collect data from cluster heads and forward it to clouds using various algorithms, reducing energy consumption. One way to cluster the cluster heads is by selecting Fog members using K-means clustering, which is a simple and effective technique for preparing a cluster using various features. The location of the cluster heads is used as input features for preparing Fog clusters. Every Fog cluster elects its best possible Fog cluster

head, which aggregates the data sent by its nearest cluster heads. Figure 4.4(a) illustrates the clusters formed using the K-means clustering technique simulated with 800 nodes, with the number of Fog clusters set to 8% and a maximum of 100 iterations for converging the solution. To predict the initial Fog centres, the K-means++ algorithm is used, and the pseudo code for this algorithm is given below.

Pseudo Code of k-means++ algorithm

1. A random observation is selected uniformly from dataset x . Selected observation is first centroid c .
2. Distances are calculated from each point to c which is denoted as $d(X_i, c)$
3. Select the next centroid, c_2 at random from X with probability as shown in equation 8

$$\frac{d^2(X_m, c_1)}{\sum_{j=1}^n d^2(X_j, c_1)} \quad (12)$$

4. To choose centre j
 - a. Calculate the distances from each point to each value of c , and assign each observation to its closest centroid.
 - b. For $m = 1, \dots, n$ and $p = 1, \dots, j-1$, select centroid j at random from X with probability

$$\frac{d^2(X_m, c_p)}{\sum_{h, x_h \in C_p} d^2(X_h, c_p)} \quad (13)$$

where C_p is the set of all observations closest to centroid c_p and x_m belongs to C_p .

5. Repeat above step 4 until k centroids are selected.

Minimum variance [38] is then calculated using equation 10

$$\left(\frac{1}{n}\right) * \sum \left(\min_j d^2(X_i, m_j)\right), \text{ for } i = 1 \text{ to } n \quad (14)$$

where $d(X_i, m_j)$ is the Euclidean distance between X_i and m_j . Cluster centroids are referred as $\{j\}_{k=1}$.

Table 4.2: Pseudo code for k-means++ algorithm

4.3.2.3 Fog Clustering using Self Organizing Maps (SOM)

The study proposes the use of Self Organizing Maps (SOM) for clustering in the Fog layer. SOM is a clustering technique based on similarity in samples, inspired by mammalian brain and motor mappings. The SOM model uses x and y coordinates of cluster head positions as

input features, and 8% of output clusters with 16 layers. The weights and bias updates are generated after every epoch while clustering process, and training stops when the maximum number of epochs is reached, set at 1000 in this study. Figure 4.3(a) and 4.3(b) illustrates the SOM neighbour connections and neighbour weights, while Figure 4.3(c) and 4.3(d) show the relation of input and output weights when connected with SOM clustering. Figure 4.3(e) and 4.3(f) demonstrates the number of cluster heads clustered with FOG cluster heads and their connected weights. The SOM clustering balances the cluster heads in each FOG cluster effectively, as shown in Figure 4.4(b), which displays the final clusters when the maximum number of epochs is reached.

4.3.2.4 Fog Clustering with Fuzzy Networks

Fuzzy c-means (FCM) is a clustering algorithm that can be used to group data based on certain features. FCM aims to minimize the objective function given in equation 11, and allows data points to belong to different clusters with varying degrees of membership. Figure 4.4(c) illustrates the cluster formation in the FOG layer for the first iteration, using a fuzzy overlapping value of 2. The objective function reaches a value of 3.01e+5 at the 100th iteration.

$$J_m = \sum_{i=1}^P \sum_{j=1}^N \mu_{ij}^m \|x_i - c_j\|^2 \quad (15)$$

Where P and N are the number of data points and number of clusters respectively. m is the exponent of overlap control for fuzzy. x_i and c_j are the i th data point and centre of the j th cluster respectively. μ_{ij} is the membership degree of x_i in the j th cluster. For a given data point, x_i , the sum of the membership values for all clusters is one.

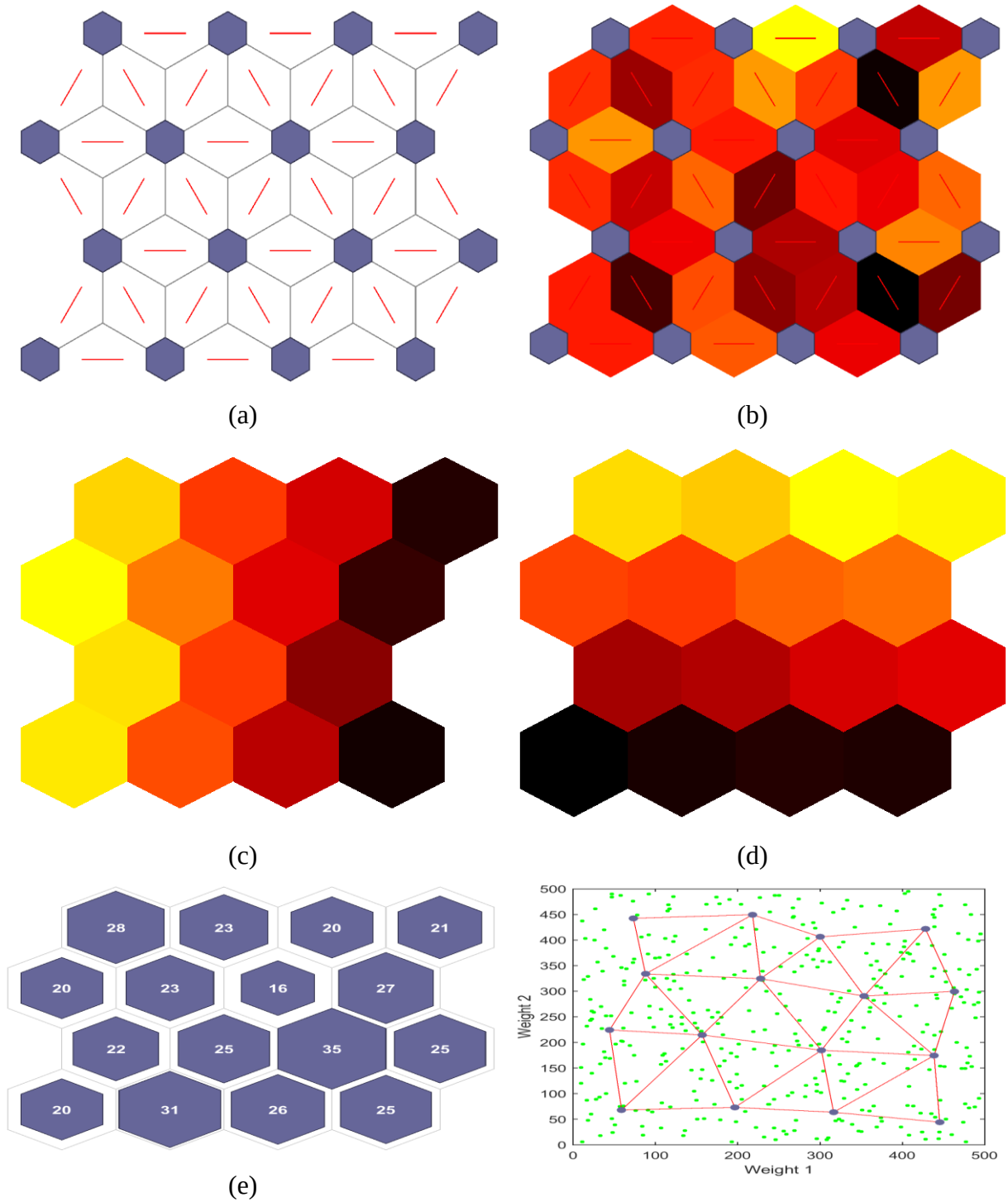


Figure 4.3: SOM clustering (a) SOM neighbour connections, (b) neighbour weights, (c) represent input weights of SOM clusters, (d) represent output weights of SOM cluster, (e) shows the number of cluster head are clustered with FOG cluster heads and (f) shows the final connected weights of cluster heads with FOG clustered heads.

4.3.2.5 Ensemble Model for Fog Clustering

This study focuses on clustering the FOG layer using three different methods and combining them to create an ensemble model to predict final FOG clusters and FOG cluster heads. To predict ensemble clusters, a weighted sum criterion is used, which utilizes information from all three clustering algorithms. The resulting FOG layer is shown in Figure 4.5, which facilitates communication between cluster heads and the cloud. Figure 4.4(d) shows the cluster created by the ensemble model.

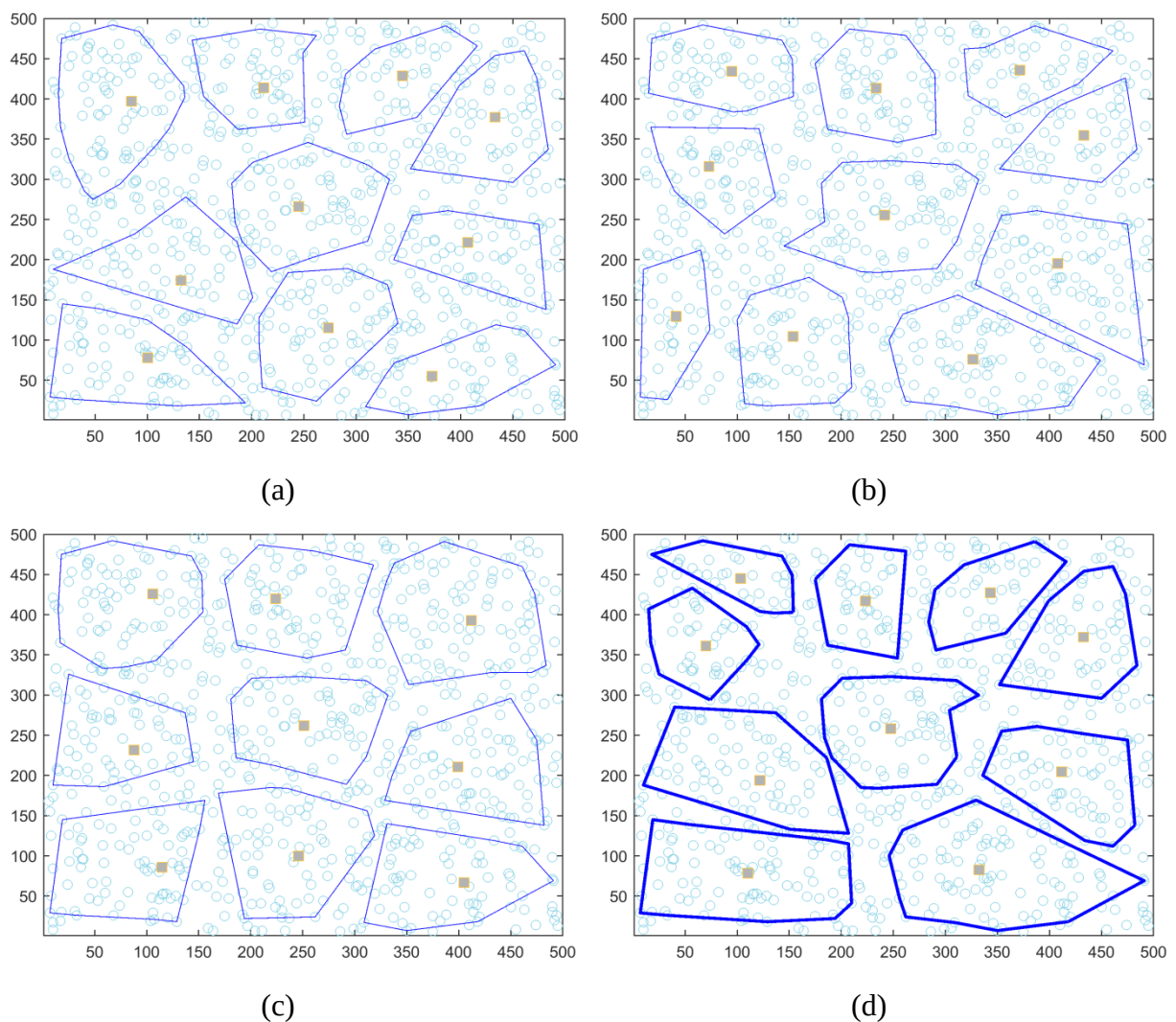


Figure 4.4: Clustering with various techniques(a) Clustering with SOM, (b) Clustering with k-means, (c) Clustering with Fuzzy and (d) Clustering with ensemble

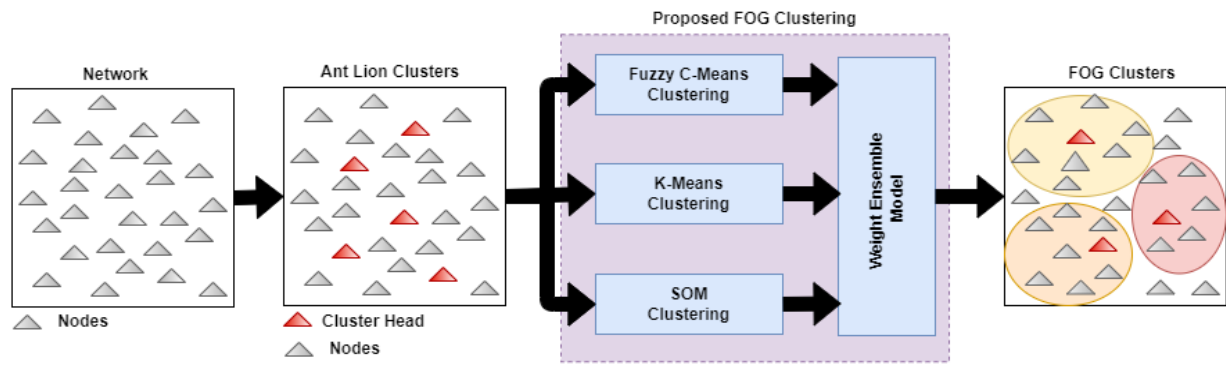


Figure 4.5: Shows the Ensemble clustered FOG layer

4.4 Results & Discussion

To evaluate the results from the proposed model a simulation environment is created which discussed in the below subsection

4.4.1 Parameters and Assumptions for Simulation Environment

This section provides an overview of the parameters and assumptions used in the simulations. The study considered 200 to 800 sensor nodes deployed in an area of 500 x 500 m². The length of the data packet was set to 1000 bits, which is equivalent to 125 bytes. The data packet is composed of a 10-byte header and the rest is payload. Figure 4.6 illustrates the packet structure.

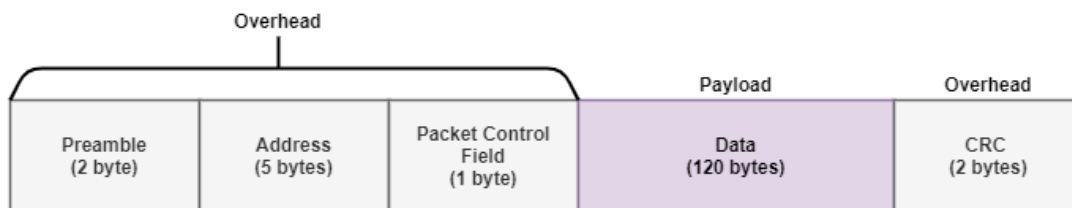
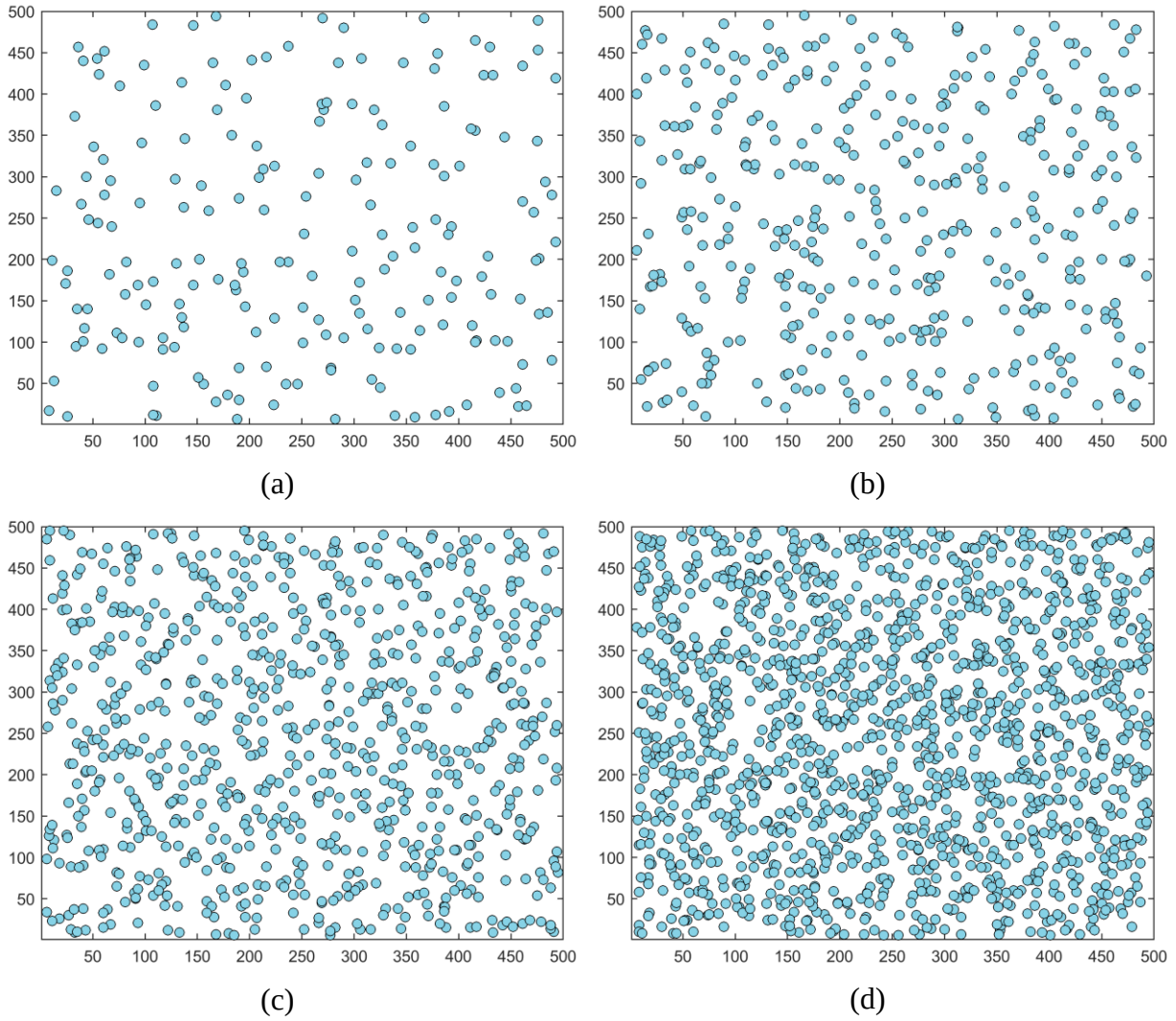


Figure 4.6: Data Packet Representation

The cluster range was set between 2% and 5% of the total number of nodes. Each sensor node was assigned a unique ID, and nodes were randomly deployed uniformly in the simulation area. Initially, all nodes had the same energy level, and each node was capable of sensing, computing, and aggregating data. Power consumption was dependent on the transmission



distance, and all nodes were assumed to be stationary, while the sink node was moving. Figure 4.7: Nodes representation deployed in range with (a) 200 nodes, $\lambda=0.08$. (b) 400 nodes, $\lambda=0.16$ (c) 800 nodes, $\lambda=0.32$ (d) 1500 nodes, $\lambda=0.60$

Node density is represented by λ (number of nodes per 10m^2). Figure 4.2 (a-d) shows the variation in representation with different value of λ and different number of nodes. Table 4.3 shows the parameters and assumptions made for simulations.

Simulation Parameter	Value
Area	$500 \times 500 \text{ m}^2$
Range for number of nodes	200 – 800
Length of data packet	1000 bits (125 bytes)
Node Density	0.08 – 0.60
E_T	50 nJ
E_R	50 nJ

Table 4.3: Parameters and Assumptions for Simulation

4.5 Results

Simulation results were obtained in this study using MATLAB 9.4 tool. To evaluate the network performance, parameters such as residual energy of sensor nodes, network lifetime, and number of alive/dead nodes were used and compared with previous studies [31], [40], [41]. The simulation area of $500 \times 500 \text{ m}^2$ was used, and the sensor nodes were randomly uniformly distributed with density varying from 0.08 to 0.60. The initial energy of sensor nodes was assumed to be 5J with a sensing radius of 10m.

Table 4.4 presents the total residual energy of nodes for simulation time ranging from 1000 to 2000 seconds. The results showed an improvement in residual energy by 1.35% at 1000 seconds compared to ECCM, while the improvement increased to 4.23% at 2000 seconds, indicating a trend of network energy saving over the network runtime. The proposed work also showed a significant improvement in residual energy of network compared to algorithms such as TBSR and DMOA. Figure 4.8 demonstrated the performance variation in terms of network lifetime of the proposed method with existing techniques.

Algorithms	Total Networks					
	1000 (seconds)	1200 (seconds)	1400 (seconds)	1600 (seconds)	1800 (seconds)	2000 (seconds)
DMOA	3450	3250	2900	2500	2000	1500

TBSK	3500	3400	3200	2900	2600	2300
ECCM	3770	3740	3700	3650	3600	3550
Proposed	3820	3810	3800	3750	3730	3700

Table 4.4: shows total residual energy of nodes for simulation time

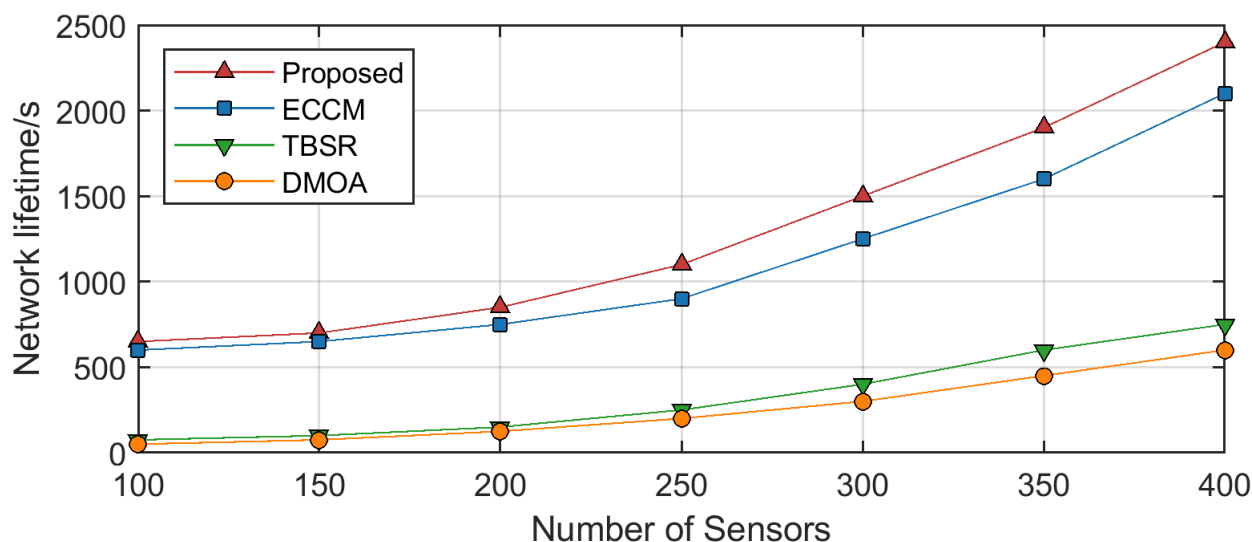


Figure 4.8: Comparison plot of Network lifetime

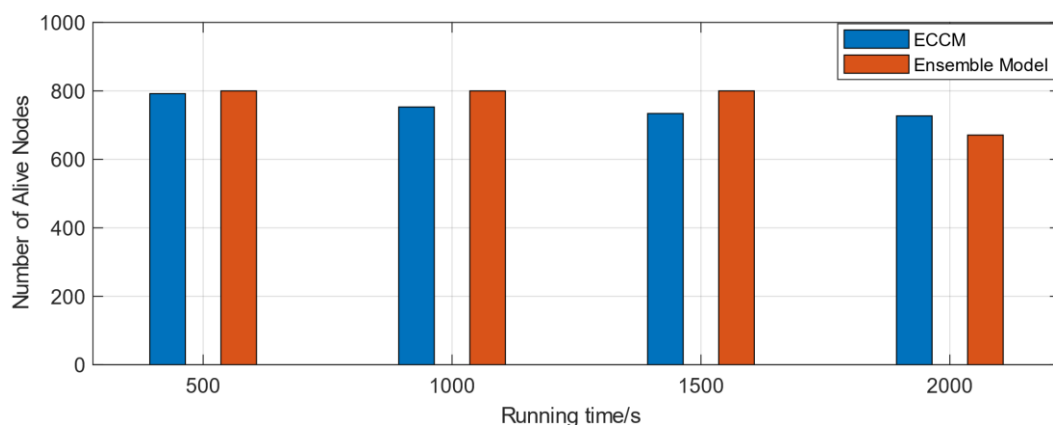


Figure 4.9: Comparison histogram of alive nodes at various time stamps

This study evaluated the network performance using varying numbers of sensor nodes, ranging from 100 to 400 with a resolution of 50. The results showed that the overall network lifetime improved slightly with fewer nodes, but the improvement increased as the number of nodes increased. Specifically, the proposed ensemble model showed a significant improvement in network lifetime, by 14.29%, compared to the ECCM approach when 400

nodes were used, as shown in Figure 4.8. Additionally, the number of alive nodes at various time stamps was measured to evaluate the load balancing of nodes, as shown in Figure 4.9. The results indicated that no nodes died until the simulation time of 1500 seconds using the proposed technique, while 66 nodes died using the ECCM approach for the FOG layer. Moreover, at the 2000-second simulation time, the proposed work had a greater number of alive nodes, by 7.4%, compared to the ECCM approach.

5. Conclusion & Future Score

5.1 Conclusion

The outcome presented here presents a proposed algorithm for enhancing the energy efficiency and stability of Wireless Sensor Networks that is motivated by fog computing (WSNs). To find the best Cluster Heads (CHs) and Cluster Members, the algorithm uses Ant Lion Optimization (ALO) for clustering in the bottom layer (CMs). The fog layer is the following layer, where k-means predictions are made for the fog clusters and nodes. The use of k-means in the fog layer is a significant contribution of this work because it simplifies and accelerates the computations in the bottom layer.

The results show that the proposed algorithm outperforms ECCM in terms of load balancing, dead node reduction, and network lifetime by 14.29%, 57, and 14.29%, respectively, when compared to traditional algorithms in general. These findings imply that the proposed algorithm outperforms conventional algorithms in terms of stability and energy efficiency.

To create more stable and energy-efficient networks, the authors propose combining the suggested algorithm with other optimisation techniques or hybrid optimisation techniques in future work. This might enhance the algorithm's functionality and make it even more appropriate for WSNs.

Overall, the study's findings are encouraging and show how the proposed algorithm may be used to increase the stability and energy efficiency of WSNs. Fog computing, clustering, and the optimisation offered by ALO and k-means together have the potential to significantly lower energy consumption and extend network lifetime, which could have a significant impact on the effectiveness of WSNs in a variety of applications.

A proposed algorithm is presented in the result that aims to reduce energy consumption in Wireless Sensor Networks (WSNs) by promoting fog computing. The bottom layer of the algorithm is based on Ant Lion optimisation for clustering, which identifies the best cluster heads and members. The calculations are then performed using an ensemble model of three clustering techniques—k-means, SOM, and fuzzy c-means—in a virtual layer also referred to as the fog layer (FCM).

The use of clustering algorithms like k-means, SOM, and FCM, which have less complexity than other hybrid optimisation algorithms, is one of the work's major contributions. The proposed algorithm is more effective as a result of reduced runtime computations and physical memory usage. Another significant finding was that 7.4% more nodes were still alive after 2000 seconds of simulation time compared to ECCM, proving the effectiveness of the suggested algorithm in lengthening network lifetime.

When compared to conventional techniques, the proposed work also demonstrated better load balancing. To improve the ensemble model's clustering accuracy and thus the network lifetime, the authors propose investigating additional clustering techniques as future work.

Overall, the study's findings are encouraging and show how the proposed algorithm may be used to reduce energy consumption in WSNs. The effectiveness of WSNs in various applications can be significantly impacted by the use of fog computing and clustering techniques, which can significantly decrease energy consumption and increase network lifetime.

5.2 Future Scope

With many new trends and applications, wireless sensor networks (WSNs) have a wide range of future prospects. WSNs are anticipated to have a significant impact in the following key areas in the future

Integration with the Internet of Things (IoT)

WSNs are a crucial part of the IoT architecture. WSNs will continue to offer the essential infrastructure for wirelessly collecting data from various sensors as more devices become interconnected. This integration will make it possible to seamlessly incorporate sensors into commonplace items, turning them into "smart" gadgets with effective data collection and analysis capabilities.

Smart Cities

When it comes to creating smart cities, where a variety of sensors are used to monitor and manage urban infrastructure, WSNs will be crucial. They can be used for a variety of things, including waste management, traffic management, environmental monitoring, energy optimisation, and public safety. WSNs will make it possible to collect data, analyse it, and make decisions in real-time, which will enhance urban areas' overall quality of life.

Industrial Automation

WSNs are anticipated to have a wide range of uses in industrial automation, making it possible to monitor machinery, processes, and environmental factors in real-time. They can raise output, reduce downtime, make the best use of available resources, and increase worker security. WSNs will help with the implementation of Industry 4.0 ideas like remote monitoring and control, asset tracking, and predictive maintenance.

Healthcare and Telemedicine

By enabling wearable technology, telemedicine, and remote patient monitoring, WSNs are poised to transform the healthcare industry. These networks can monitor vital signs, spot anomalies, and wirelessly transmit medical data to healthcare professionals, enabling prompt intervention and individualised care. WSNs will be crucial in enhancing patient outcomes, lowering healthcare expenses, and facilitating access to healthcare in rural areas.

Environmental Monitoring

WSNs are expected to play a significant role in environmental monitoring and conservation efforts. These networks can set up sensors to keep an eye on wildlife activity, soil conditions, and the quality of the air and water. WSNs will aid in the early detection of environmental hazards, support sustainable resource management, and contribute to conservation efforts

Agriculture

WSNs will significantly affect agriculture by facilitating precision farming methods. WSNs can keep an eye on factors like soil moisture, temperature, humidity, and crop health by placing sensors in fields. This useful information can be used to reduce environmental impact, increase crop yields, and optimise resource usage through effective irrigation, fertilisation, and pest control.

Disaster Management

By providing early warning systems, monitoring environmental conditions, and facilitating communication in urgent circumstances, WSNs can prove essential in disaster management scenarios. These networks can improve emergency response efforts and save lives by assisting in the preparation for and response to natural disasters like earthquakes, floods, and wildfires.

Fog computing holds great promise for wireless sensor networks (WSNs) power optimisation. The following are some crucial areas where incorporating power-optimization methods into WSNs and fog computing can make a big difference

Energy-Efficient Data Processing

By using fog computing, it is possible to transfer computationally demanding tasks from sensors with limited resources to fog nodes located closer to the network edge. Power consumption in WSNs can be decreased by dividing data processing and analysis tasks among different workers. By only transmitting the information that is absolutely necessary, fog nodes can perform tasks like data aggregation, filtering, and preliminary analysis, allowing sensors to save energy.

Adaptive Sleep Scheduling

According to the needs of the sensors' data collection, power optimisation techniques can be used to dynamically change the sleep schedules of the sensors. WSNs can save energy without sacrificing the timeliness or accuracy of their data by intelligently managing their sleep and wake-up cycles. In order to reduce overall power consumption, fog nodes can help coordinate sleep schedules and wake up specific sensors when required.

Efficient Routing Protocols

Routing protocols that consider power consumption are essential for reducing energy use in WSNs. Routing decisions can be optimised to use less energy by taking into account things like energy levels, communication distance, and transmission power. By adding more processing power and enabling localised decision-making, fog computing can improve routing protocols and result in more energy-efficient data routing.

Context-Aware Data Processing

Context-aware data processing is made possible by fog computing, where data is processed and analysed based on contextual data that is available at the fog nodes. Fog nodes can intelligently filter, aggregate, and process sensor data, reducing unnecessary transmissions and saving power, by utilising contextual cues like location, time, and event triggers.

Energy Harvesting and Power Management

The incorporation of energy harvesting techniques within WSNs is made easier by fog computing. Sensors can be powered or have their batteries recharged using energy harvesting resources like solar panels or kinetic energy harvesters. The WSN can operate sustainably thanks to fog nodes' ability to monitor energy levels, anticipate power needs, and optimise energy allocation across the network.

Dynamic Resource Allocation

Fog computing makes it possible for sensors to receive efficient dynamic resource allocation based on their energy needs and data processing requirements. Utilizing the processing power of fog nodes allows computational tasks to be transferred from sensors to the fog layer, thereby lowering the power requirements of individual sensors and extending the lifespan of the network.

Collaborative Edge Analytics

Collaborative edge analytics are made possible by fog computing, where several fog nodes work together to process and analyse data. This collaboration may lead to decreased data transmission, distributed data processing, and increased energy efficiency. It is possible to reduce power consumption in WSNs by distributing computation and utilising local resources.

In conclusion, a wide range of fields, such as IoT integration, smart cities, industrial automation, healthcare, environmental monitoring, agriculture, and disaster management, fall under the future purview of wireless sensor networks. WSNs will continue to be essential in creating a connected and intelligent world as technology develops. Future power optimisation in WSNs using fog computing will focus on increasing energy efficiency, extending network lifetime, and enhancing performance in general. WSNs can achieve effective data processing, adaptive sleep scheduling, energy harvesting integration, and collaborative edge analytics by utilising the capabilities of fog computing, ultimately leading to sustainable and power-efficient operation.

List of Publication

Paper Published

1. Jain, A., & Bhardwaj, A. K. (2022). Power-efficient optimized clustering method with intelligent fog computing for wireless sensor networks. *Concurrency and Computation Practice and Experience*, 34(15), [1.831]. SCI.
2. Jain, A., Bhardwaj, A.K. Machine Learning Based Power Efficient Optimized Communication Ensemble Model with Intelligent Fog Computing for WSNs. *Wireless Pers Commun* (2023). [2.017]. SCI.

References/Bibliography:

- [1] Dastjerdi, A., H. Gupta, R. Calheiros, S. Ghosh, and R. Buyya. 2016. Chapter 4—fog computing: Principles, architectures, and applications. In *Internet of Things: Principles and Paradigms*, ed. R. Buyya, and A.V. Dastjerdi, 61–75. New York: Morgan Kaufmann.
- [2] Sarkar, S., and S. Misra. 2016. Theoretical modelling of fog computing: A green computing paradigm to support Iot applications. *ET Networks* 5(2): 23–29.
- [3] Bonomi, F., R. Milito, J. Zhu, and S. Addepalli. 2012. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, ACM, 13–16.
- [4] Sarkar, S., S. Chatterjee, and S. Misra. 2015. Assessment of the suitability of fog computing in the context of internet of things. *IEEE Transactions on Cloud Computing* PP(99): 1–1.
- [5] Garcia Lopez, P., A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere. 2015. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review* 45(5): 37–42.
- [6] Varghese, B., N. Wang, S. Barbhuiya, P. Kilpatrick, and D.S. Nikolopoulos. 2016. Challenges and opportunities in edge computing. In *Proceedings of the IEEE International Conference on Smart Cloud*, 20–26.
- [7] Shi, W., J. Cao, Q. Zhang, Y. Li, and L. Xu. 2016. Edge computing: vision and challenges. *IEEE Internet of Things Journal* 3(5): 637–646.
- [8] Hu, Y.C., M. Patel, D. Sabella, N. Sprecher, and V. Young. 2015. Mobile edge computing a key technology towards 5g. *ETSI White Paper* 11: 1–16.
- [9] Klas, G.I. 2015. Fog Computing and Mobile Edge Cloud Gain Momentum Open Fog Consortium, ETSI MEC and Cloudlets.
- [10] Cau, E., M. Corici, P. Bellavista, L. Foschini, G. Carella, A. Edmonds, and T.M. Bohnert. 2016. Efficient exploitation of mobile edge computing for virtualized 5g in epc architectures. In *4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, (March 2016), 100–109.

- [11] Ahmed, A., and E. Ahmed. 2016. A survey on mobile edge computing. In *Proceedings of the 10th IEEE International Conference on Intelligent Systems and Control (ISCO 2016)*, Coimbatore, India.
- [12] Mahmud, M.R., M. Afrin, M.A. Razzaque, M.M. Hassan, A. Alelaiwi, and M. Alrubaian. 2016. Maximizing quality of experience through context-aware mobile application scheduling in cloudlet infrastructure. *Software: Practice and Experience* 46(11): 1525–1545. spe.2392.
- [13] Sanaei, Z., S. Abolfazli, A. Gani, and R. Buyya. 2014. Heterogeneity in mobile cloud computing: Taxonomy and open challenges. *IEEE Communications Surveys and Tutorials* 16(1): 369–392.
- [14] Bahl, P., R.Y. Han, L.E. Li, and M. Satyanarayanan. 2012. Advancing the state of mobile cloud computing. In *Proceedings of the third ACM workshop on Mobile cloud computing and services*, ACM, 21–28.
- [15] Satyanarayanan, M., G. Lewis, E. Morris, S. Simanta, J. Boleng, and K. Ha. 2013. The role of cloudlets in hostile environments. *IEEE Pervasive Computing* 12(4): 40–49.
- [16] Lee, W., K. Nam, H.G. Roh, S.H. Kim. 2016. A gateway based fog computing architecture for wireless sensors and actuator networks. In *18th International Conference on Advanced Communication Technology (ICACT)*, IEEE, 210–213.
- [17] Aazam, M., and E.N. Huh. 2015. Fog computing micro datacenter based dynamic resource estimation and pricing model for iot. In *IEEE 29th International Conference on Advanced Information Networking and Applications*. (March 2015), 687–694.
- [18] Jalali, F., K. Hinton, R. Ayre, T. Alpcan, and R.S. Tucker. 2016. Fog computing may help to save energy in cloud computing. *IEEE Journal on Selected Areas in Communications* 34(5): 1728–1739.
- [19] Zhu, J., D.S. Chan, M.S. Prabhu, P. Natarajan, H. Hu, F. Bonomi. 2013. Improving web sites performance using edge servers in fog computing architecture. In *Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on*, (March 2013), 320–323.
- [20] Zeng, D., L. Gu, S. Guo, Z. Cheng, and S. Yu. 2016. Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. *IEEE Transactions on Computers* PP(99): 1–1.

- [21] Hong, K., D. Lillethun, U. Ramachandran, B. IoTtenwälder, and B. Koldehofe. 2013. 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*, ACM, 15–20.
- [22] Nazmudeen, M.S.H., A.T. Wan, and S.M. Buhari. 2016. Improved throughput for power line communication (plc) for smart meters using fog computing based data aggregation approach. In *IEEE International Smart Cities Conference (ISC2)*, (Sept 2016), 1–4.
- [23] Aazam, M., and E.N. Huh. 2014. Fog computing and smart gateway based communication for cloud of things. In *Future Internet of Things and Cloud (FiCloud), International Conference on IEEE (2014)*, 464–470.
- [24] Cirani, S., G. Ferrari, N. Iotti, and M. Picone. 2015. The iot hub: A fog node for seamless management of heterogeneous connected smart objects. In *12th Annual IEEE International Conference on Sensing, Communication, and Networking-Workshops (SECON Workshops), IEEE (2015)*, 1–6.
- [25] Dsouza, C., G.J. Ahn, and M. Taguinod. 2014. Policy-driven security management for fog computing: Preliminary framework and a case study. In: *IEEE 15th International Conference on Information Reuse and Integration (IRI)*, (Aug 2014), 16–23.
- [26] Cardellini, V., V. Grassi, F.L. Presti, and M. Nardelli. 2015. On qos-aware scheduling of data stream applications over fog computing infrastructures. In *IEEE Symposium on Computers and Communication (ISCC)*, (July 2015), 271–276.
- [27] Yan, S., M. Peng, and W. Wang. 2016. User access mode selection in fog computing based radio access networks. In *IEEE International Conference on Communications (ICC)*, (May 2016), 1–6.
- [28] Gu, L., D. Zeng, S. Guo, A. Barnawi, and Y. Xiang. 2015. Cost-efficient resource management in fog computing supported medical cps. *IEEE Transactions on Emerging Topics in Computing* PP(99): 1–1.
- [29] Truong, N.B., G.M. Lee, and Y. Ghamri-Doudane. 2015. Software defined networking-based vehicular adhoc network with fog computing. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, (May 2015), 1202–1207.

- [30] Oueis, J., E.C. Strinati, S. Sardellitti, and S.Barbarossa. 2015. Small cell clustering for efficient distributed fog computing: A multi-user case. In *Vehicular Technology Conference (VTC Fall)*, IEEE 82nd. (Sept 2015), 1–5.
- [31] Hou, X., Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen. 2016. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology* 65(6): 3860–3873.
- [32] Ye, D., M. Wu, S. Tang, and R. Yu. 2016. Scalable fog computing with service offloading in bus networks. In *IEEE 3rd international Conference on Cyber Security and Cloud Computing (CSCloud)*, (June 2016), 247–251.
- [33] Oueis, J., E.C. Strinati, and S. Barbarossa. 2015. The fog balancing: Load distribution for small cell cloud computing. n *EEE 81st Vehicular Technology Conference (VTC spring)*, (May 2015), 1–6.
- [34] Faruque, M.A.A., and K. Vatanparvar. 2016. Energy management-as-a-service over fog computing platform. *EEE Internet of Things Journal* 3(2): 161–169.
- [35] Shi, H., N. Chen, and R. Deters. 2015. Combining mobile and fog computing: Using coap to link mobile device clouds with fog computing. n *EEE International Conference on DataScience and Data Intensive Systems*, (Dec 2015), 564–571.
- [36] Giang, N.K., M. Blackstock, R. Lea, and V.C.M.Leung. 2015. DevelopingIoT applications n the fog: A distributed dataflow approach. n *5thinternational Conference on the Internet of Things (IOT)*, (Oct 2015), 155–162.
- [37] Intharawijitr, K., K. ida, and H. Koga. 2016. Analysis of fogmodel considering computing and communication latency n 5g cellular networks. n *EEEinternational Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, (March 2016), 1–4.
- [38] Peng, M., S. Yan, K. Zhang, and C.Wang. 2016. Fog-computing-based radio access networks: ssues and challenges. *EEE Network* 30(4): 46–53.
- [39] Hassan, M.A., M. Xiao, Q. Wei, and S.Chen. 2015. Help your mobile applications with fog computing. n *12th Annual EEEinternational Conference on Sensing, Communication, and Networking - Workshops (SECON Workshops)*, (June 2015), 1–6.

- [40] Zhang, W., B. Lin, Q. Yin, and T. Zhao. 2016. Infrastructure deployment and optimization of fog network based on microdc and lrp on integration. *Peer-to-Peer Networking and Applications* 1–13.
- [41] Deng, R., R. Lu, C. Lai, T.H. Luan, and H. Liang. 2016. Optimal workload allocation in fogcloud computing towards balanced delay and power consumption. *IEEE Internet of Things Journal* PP(99): 1–1.
- [42] Do, C.T., N.H. Tran, C. Pham, M.G.R. Alam, J.H. Son, and C.S. Hong. 2015. A proximal algorithm for joint resource allocation and minimizing carbon footprint in geo-distributed fog computing. in *International Conference on Information Networking (ICOIN)*, (Jan 2015), 324–329.
- [43] Aazam, M., M. St-Hilaire, C.H. Lung, and . Lambadaris. 2016. Pre-fog:IoT trace based probabilistic resource estimation at fog. in *13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, (Jan 2016), 12–17.
- [44] Datta, S.K., C. Bonnet, and J. Haerri. 2015. Fog computing architecture to enable consumer centric internet of things services. in *International Symposium on Consumer Electronics (ISCE)*, (June 2015), 1–2.
- [45] Aazam, M., M. St-Hilaire, C.H. Lung, and . Lambadaris. 2016. Mefore: Qoe based resource estimation at fog to enhance qos in IoT. in *23rd International Conference on Telecommunications (ICT)*, (May 2016), 1–5.
- [46] Gazis, V., A. Leonardi, K. Mathioudakis, K. Sasloglou, P. Kikiras, and R. Sudhaakar. 2015. Components of fog computing in an Industrial Internet of things context. in *12th Annual IEEE International Conference on Sensing, Communication, and Networking - Workshops (SECON Workshops)*, (June 2015) 1–6.
- [47] Gupta, H., A.V. Dastjerdi, S.K. Ghosh, and R. Buyya. 2016. fogsim: A toolkit for modeling and simulation of resource management techniques in Internet of things, edge and fog computing environments.
- [48] Redowan Mahmud, Ramamohanarao Kotagiri and Rajkumar Buyya. 2018 Fog Computing: A Taxonomy, Survey and Future Directions
- [49] Binh Mih Nguyen, Huynh Thi Thanh Binh, Tran The Anh, Do Bao Son. Evolutionary algorithms to optimize task scheduling problem for the IoT based Bag-of-Tasks application in cloud-Fog Computing Environment. *Appl. Sci.* 2019, 9, 1730.

- [50] Zeyu Sun, Lili Wei, Chen Xu, Tian Wang, Yalin Nie, Xiaofei Xing, Jianfeng Lu. An Energy Efficient Cross Layer Sensing Clustering method based on Intelligent Fog Computing n WSNs". *Special section on Green Internet of Things* (Volume -7, September, 2019).
- [51] M. Chiang, T. Zhang, Fog and IoT: an overview of research opportunities, *IEEE Int. Things J.* 3 (6) (2016) 854–864.
- [52] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, J.P. Jue, All one needs to know about fog computing and related edge computing paradigms: a complete survey, *J. Syst. Archit.* 98 (2019) 289–330.
- [53] M. Dabbaghjamanesh, A. Moeini, A. Kavousi-Fard, A. Jolfaei, Real-time monitoring and operation of microgrid using distributed cloud–fog architecture, *J. Parallel Distrib. Comput.* 146 (2020) 15–24.
- [54] S.S. Gill, S. Tuli, M. Xu, I. Singh, K.V. Singh, D. Lindsay, S. Tuli, D. Smirnova, M. Singh, U. Jain, et al., Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: evolution, vision, trends and open challenges, *IEEE Int. Things J.* 8 (2019) 100118.
- [55] OpenFog Consortium Architecture Working Group, OpenFog Reference Architecture for Fog Computing, Tech. Rep, February 2017 [Online]. Available: https://www.iiconsortium.org/pdf/OpenFog_Reference_Architecture_2_09_17.pdf.
- [56] P. Habibi, M. Farhodi, S. Kazemian, S. Khorsandi, A. Leon-Garcia, Fog computing: a comprehensive architectural survey, *IEEE Access* 8 (2020) 105–69 133.
- [57] S.V. Margariti, V.V. Dimakopoulos, G. Tsoumanis, Modeling and simulation tools for fog computing—a comprehensive survey from a cost perspective, *Future Internet* 12 (5) (2020) 89.
- [58] A. Brogi, S. Forti, C. Guerrero, I. Lera, How to place your apps in the fog: state of the art and open challenges, *Softw. Pract. Exp.* 50 (5) (2020) 719–740.
- [59] A. Markus, A. Kertesz, A survey and taxonomy of simulation environments modelling fog computing, *Simul. Model. Pract. Theory* 101 (2020) 102042.
- [60] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, J.P. Jue, All one needs to know about fog computing and related edge computing paradigms: a complete survey, *J. Syst. Archit.* 98 (2019) 289–330.

- [61] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, S. Chen, Vehicular fog computing: a viewpoint of vehicles as the infrastructures, *IEEE Trans. Veh. Technol.* 65 (6) (2016) 3860–3873.
- [62] N. Fernando, S.W. Loke, I. Avazpour, F.F. Chen, A.B. Abkenar, A. Ibrahim, Opportunistic fog for IoT: challenges and opportunities, *IEEE Int. Things J.* 6 (5) (2019) 8897–8910.
- [63] C. Mouradian, D. Naboulsi, S. Yangui, R.H. Glitho, M.J. Morrow, P.A. Polakos, A comprehensive survey on fog computing: state-of-the-art and research challenges, *IEEE Commun. Surv. Tutor.* 20 (1) (2018) 416–464
- [64] J. Wu, M. Dong, K.IoTa, J. Li, W. Yang, and M. Wang, “Fog-computing-enabled cognitive network function virtualization for an information-centric future internet,” *IEEE Commun. Mag.*, vol. 57, no. 7, pp. 48–54, 2019, doi: 10.1109/MCOM.2019.1800778.
- [65] T. Wang, Y. Liang, W. Jia, M. Arif, A. Liu, and M. Xie, “Coupling resource management based on fog computing in smart city systems,” *J. Netw. Comput. Appl.*, vol. 135, no. September 2018, pp. 11–19, 2019, doi: 10.1016/j.jnca.2019.02.021.
- [66] S. Yi, C. Li, and Q. Li, “A survey of fog computing: Concepts, applications and issues,” *Proc. Int. Symp. Mob. Ad Hoc Netw. Comput.*, vol. 2015-June, no. August, pp. 37–42, 2015, doi: 10.1145/2757384.2757397.
- [67] X. Liu, “Routing Protocols Based on Ant Colony Optimization in Wireless Sensor Networks: A Survey,” *IEEE Access*, vol. 5, no. c, pp. 26303–26317, 2017, doi: 10.1109/ACCESS.2017.2769663.
- [68] Sun, X. Xing, T. Wang, Z. Lv, and B. Yan, “An optimized clustering communication protocol based on intelligent computing in information-centric internet of things,” *IEEE Access*, vol. 7, pp. 28238–28249, 2019, doi: 10.1109/ACCESS.2019.2896250.
- [69] T. Wang, H. Luo, X. Zheng, and M. Xie, “Crowdsourcing mechanism for trust evaluation in CPCS based on intelligent mobile edge computing,” *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 6, 2019, doi: 10.1145/3324926.
- [70] H. Teng *et al.*, “A novel code data dissemination scheme for Internet of Things through mobile vehicle of smart cities,” *Futur. Gener. Comput. Syst.*, vol. 94, pp. 351–367, 2019, doi: 10.1016/j.future.2018.11.039.
- [71] X. Liu and P. Zhang, “Data Drainage: A Novel Load Balancing Strategy for Wireless Sensor Networks,” *IEEE Commun. Lett.*, vol. 22, no. 1, pp. 125–128, 2018, doi: 10.1109/LCOMM.2017.2751601.
- [72] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, “Making Knowledge Tradable in Edge-AI

- Enabled IoT: A Consortium Blockchain-Based Efficient and Incentive Approach,” *IEEE Trans. Ind. Informatics*, vol. 15, no. 12, pp. 6367–6378, 2019, doi: 10.1109/TII.2019.2917307.
- [73] L. Adhianto *et al.*, “HPCTOOLKIT: Tools for performance analysis of optimized parallel programs,” *Concurr. Comput. Pract. Exp.*, vol. 22, no. 6, pp. 685–701, 2010, doi: 10.1002/cpe.
- [74] T. Wang, J. Zhou, A. Liu, M. Z. A. Bhuiyan, G. Wang, and W. Jia, “Fog-based computing and storage offloading for data synchronization in IoT,” *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4272–4282, 2019, doi: 10.1109/JIOT.2018.2875915.
- [75] G. Yang, T. Liang, X. He, and N. Xiong, “Global and local reliability-based routing protocol for wireless sensor networks,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3620–3632, 2019, doi: 10.1109/JIOT.2018.2889379.
- [76] J. Jiang, S. Wen, S. Yu, Y. Xiang, and W. Zhou, “Identifying Propagation Sources in Networks: State-of-the-Art and Comparative Studies,” *IEEE Commun. Surv. Tutorials*, vol. 19, no. 1, pp. 465–481, 2017, doi: 10.1109/COMST.2016.2615098.
- [77] J. Wu, M. Dong, K. IoTa, J. Li, and Z. Guan, “Big Data Analysis-Based Secure Cluster Management for Optimized Control Plane in Software-Defined Networks,” *IEEE Trans. Netw. Serv. Manag.*, vol. 15, no. 1, pp. 27–38, 2018, doi: 10.1109/TNSM.2018.2799000.
- [78] T. Wang *et al.*, “Data collection from WSNs to the cloud based on mobile Fog elements,” *Futur. Gener. Comput. Syst.*, vol. 105, pp. 864–872, 2020, doi: 10.1016/j.future.2017.07.031.
- [79] G. Li, J. Wu, J. Li, K. Wang, and T. Ye, “Service Popularity-Based Smart Resources Partitioning for Fog Computing-Enabled Industrial Internet of Things,” *IEEE Trans. Ind. Informatics*, vol. 14, no. 10, pp. 4702–4711, 2018, doi: 10.1109/TII.2018.2845844.
- [80] Y. K. Wu, H. Huang, Q. Wu, A. Liu, and T. Wang, “A risk defense method based on microscopic state prediction with partial information observations in social networks,” *J. Parallel Distrib. Comput.*, vol. 131, pp. 189–199, 2019, doi: 10.1016/j.jpdc.2019.04.007.
- [81] Chen *et al.*, “Android HIV: A Study of Repackaging Malware for Evading Machine-Learning Detection,” *IEEE Trans. Inf. Forensics Secur.*, vol. 15, no. 8, pp. 987–1001, 2020, doi: 10.1109/TIFS.2019.2932228.
- [82] X. Liu, T. Qiu, X. Zhou, T. Wang, L. Yang, and V. Chang, “Latency-aware path planning for disconnected sensor networks with mobile sinks,” *IEEE Trans. Ind. Informatics*, vol. 16, no. 1, pp. 350–361, 2020, doi: 10.1109/TII.2019.2916300.
- [83] Lu, Y. Xin, Z. Zhang, X. Liu, and K. Li, “Game-Theoretic Design of Optimal Two-Sided

- Rating Protocols for Service Exchange Dilemma in Crowdsourcing,” *IEEE Trans. Inf. Forensics Secur.*, vol. 13, no. 11, pp. 2801–2815, 2018, doi: 10.1109/TIFS.2018.2834318.
- [84] J. Lu, Y. Xin, Z. Zhang, H. Peng, and J. Han, “Supporting user authorization queries in RBAC systems by role–permission reassignment,” *Futur. Gener. Comput. Syst.*, vol. 88, pp. 707–717, 2018, doi: 10.1016/j.future.2018.01.010.
- [85] Z Sun, R. Tao, L. Li, and X. Xing, “A new energy-efficient multi-target coverage control protocol using event-driven-mechanism in wireless sensor networks,” *Int. J. Online Eng.*, vol. 13, no. 2, pp. 53–67, 2017, doi: 10.3991/ijoe.v13i02.6465.
- [86] Zhang, T. Wang, G. Wang, A. Liu, and W. Jia, “Detection of hidden data attacks combined fog computing and trust evaluation method in sensor-cloud system,” *Concurr. Comput.*, no. November, pp. 1–13, 2018, doi: 10.1002/cpe.5109.
- [87] Z. Sun *et al.*, “An Energy-Efficient Cross-Layer-Sensing Clustering Method Based on Intelligent Fog Computing in WSNs,” *IEEE Access*, vol. 7, pp. 144165–144177, 2019, doi: 10.1109/ACCESS.2019.2944858.
- [88] T. Wang, H. Luo, W. Jia, A. Liu, and M. Xie, “MTES: An Intelligent Trust Evaluation Scheme in Sensor-Cloud-Enabled Industrial Internet of Things,” *IEEE Trans. Ind. Informatics*, vol. 16, no. 3, pp. 2054–2062, 2020, doi: 10.1109/TII.2019.2930286.
- [89] T. Wang *et al.*, “Trajectory Privacy Preservation Based on a Fog Structure for Cloud Location Services,” *IEEE Access*, vol. 5, no. 8, pp. 7692–7701, 2017, doi: 10.1109/ACCESS.2017.2698078.
- [90] Y. Lai, H. Lin, F. Yang, and T. Wang, “Efficient data request answering in vehicular Ad-hoc networks based on fog nodes and filters,” *Futur. Gener. Comput. Syst.*, vol. 93, pp. 130–142, 2019, doi: 10.1016/j.future.2018.09.065.
- [91] Y. Nie, H. Wang, Y. Qin, and Z. Sun, “Distributed and morphological operation-based data collection algorithm,” *Int. J. Distrib. Sens. Networks*, vol. 13, no. 7, 2017, doi: 10.1177/1550147717717593.
- [92] Z Sun, Y. Zhang, Y. Nie, W. Wei, J. Lloret, and H. Song, “CASMOCC: a novel complex alliance strategy with multi-objective optimization of coverage in wireless sensor networks,” *Wirel. Networks*, vol. 23, no. 4, pp. 1201–1222, 2017, doi: 10.1007/s11276-016-1213-3.
- [93] V. Potdar, A. Sharif, and E. Chang, “Wireless sensor networks: A survey,” *Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA*, pp. 636–641, 2009, doi: 10.1109/WAINA.2009.192.
- [94] C. Fu, Z. Jiang, W. Wei, and A. Wei, “An Energy Balanced Algorithm of LEACH Protocol in WSN,” *Int. J. Comput. Sci. Issues*, vol. 10, no. 1, pp. 354–359, 2013.

- [95] A. Varmaghani, A. Matin Nazar, M. Ahmadi, A. Sharifi, S. Jafarzadeh Ghouschi, and Y. Pourasad, "DMTC: Optimize Energy Consumption in Dynamic Wireless Sensor Network Based on Fog Computing and Fuzzy Multiple Attribute Decision-Making," *Wirel. Commun. Mob. Comput.*, vol. 2021, 2021, doi: 10.1155/2021/9953416.
- [96] J. Tang, A. Liu, J. Zhang, N. N. Xiong, Z. Zeng, and T. Wang, "A trust-based secure routing scheme using the traceback approach for energy-harvesting wireless sensor networks," *Sensors (Switzerland)*, vol. 18, no. 3, pp. 4–9, 2018, doi: 10.3390/s18030751.
- [97] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wirel. Commun.*, vol. 1, no. 4, pp. 660–670, 2002, doi: 10.1109/TWC.2002.804190.
- [98] X. Liu, "Node deployment based on extra path creation for wireless sensor networks on mountain roads," *IEEE Commun. Lett.*, vol. 21, no. 11, pp. 2376–2379, 2017, doi: 10.1109/LCOMM.2017.2739727.
- [99] Z. Sun, W. Wu, H. Wang, H. Chen, and X. Xing, "A novel coverage algorithm based on event-probability-driven mechanism in wireless sensor network," *Tijdschr. voor Urol.*, vol. 2014, no. 1, pp. 1–17, 2014, doi: 10.1186/1687-1499-2014-58.
- [100] P. Shroff, A. Bandyopadhyay, A novel matching framework for one-sided markets in fog computing, *Int. J. Comput. Digit. Syst.* (2020) 1–10.
- [101] A. Bonadio, F. Chiti, R. Fantacci, An integrated framework for fog communications and computing in Internet of vehicles, *EPic Ser. Comput.* 56 (2018) 84–92.
- [102] F. Bonomi, R. Milito, J. Zhu, S. Addepalli, Fog computing and its role in the Internet of things, in: *MCC'12 - Proceedings of the 1st ACM Mobile Cloud Computing Workshop*, August, 2012, 2012, pp. 13–15.
- [103] D. Borthakur, H. Dubey, N. Constant, L. Mahler, K. Mankodiya, Smart fog: fog computing framework for unsupervised clustering analytics in wearable Internet of things, in: *2017 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2017 - Proceedings*, 2018-Janua, 2018, pp. 472–479.
- [104] P. Brereton, B.A. Kitchenham, D. Budgen, M. Turner, M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain, *J. Syst. Softw.* 80 (4) (2007) 571–583. [
- [105] A. Brogi, S. Forti, C. Guerrero, I. Lera, How to place your apps in the fog: state of the art and open challenges, *Softw. Pract. Exp.* 50 (5) (2020) 719–740.

- [106] R.N. Calheiros, R. Ranjan, A. Beloglazov, C.A. De Rose, R. Buyya, Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, *Softw. Pract. Exp.* 41 (1) (2011) 23–50.
- [107] B. Cheng, G. Solmaz, F. Cirillo, E. Kovacs, K. Terasawa, A. Kitazawa, FogFlow: easy programming of IoT services over cloud and edges for smart cities, *IEEE Int. Things J.* 5 (2) (2018) 696–707.
- [108] M. Chiang, T. Zhang, Fog and IoT: an overview of research opportunities, *IEEE Int. Things J.* 3 (6) (2016) 854–864.
- [109] F. Chiti, R. Fantacci, B. Picano, A matching theory framework for tasks offloading in fog computing for IoT systems, *IEEE Int. Things J.* 5 (6) (2018) 5089–5096.
- [110] M. Dabbaghjamesh, A. Moeini, A. Kavousi-Fard, A. Jolfaei, Real-time monitoring and operation of microgrid using distributed cloud–fog architecture, *J. Parallel Distrib. Comput.* 146 (2020) 15–24.
- [111] F. Davami, S. Adabi, A. Rezaee, A.M. Rahmani, Fog-based architecture for scheduling multiple workflows with high availability requirement, *Computing* (2021) 1–40.
- [112] R. De Fátima Pereira Marquesone, É.A. Da Silva, N.M. Gonzalez, K. Langona, W.A. Goya, F.F. Redígolo, T.C.M. De Brito Carvalho, J.E. Mangs, A. Sefidcon, Towards bandwidth optimization in fog computing using FACE framework, in: *CLOSER 2017 - Proceedings of the 7th International Conference on Cloud Computing and Services Science*, 2017, pp. 463–470.
- [113] Deepali, K. Bhushan, DDoS attack defense framework for cloud using fog computing, in: *RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology*, Proceedings, 2018-Janua, 2017, pp. 534–538.
- [114] Q. Deng, M. Goudarzi, R. Buyya, Fogbus2: a lightweight and distributed container-based framework for integration of IoT-enabled systems with edge and cloud computing, in: *SIGMOD 2021 International Workshop on Big Data in Emergent Distributed Environments*, BiDEDE 2021, June 20-25, 2021.
- [115] S. Deokar, M. Mangla, R. Akhare, A secure fog computing architecture for continuous health monitoring, in: *Fog Computing for Healthcare 4.0 Environments*, Springer, 2021, pp. 269–290.
- [116] A. Dhillon, A. Singh, H. Vohra, C. Ellis, B. Varghese, S.S. Gill, IoTpulse: machine learning-based enterprise health information system to predict alcohol addiction in Punjab (India) using IoT and fog computing, *Enterp. Inf. Syst.* (2020) 1–33.

- [117] J. Dizdarevic, ´ F. Carpio, A. Jukan, X. Masip-Bruin, A survey of communication protocols for Internet of things and related challenges of fog and cloud computing integration, *ACM Comput. Surv.* 51 (6) (2019) 1–29.
- [118] C. Dsouza, G.J. Ahn, M. Taguinod, Policy-driven security management for fog computing: preliminary framework and a case study, in: *Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration, IEEE IRI 2014, 2014*, pp. 16–23.
- [119] H. Du, S. Leng, F. Wu, X. Chen, S. Mao, A new vehicular fog computing architecture for cooperative sensing of autonomous driving, *IEEE Access* 8 (2020) 997–11 006.
- [120] M. Etemadi, M. Ghobaei-Arani, A. Shahidinejad, Resource provisioning for IoT services in the fog computing environment: an autonomic approach, *Comput. Commun.* 161 (2020) 109–131.
- [121] N. Fernando, S.W. Loke, I. Avazpour, F.F. Chen, A.B. Abkenar, A. Ibrahim, Opportunistic fog for IoT: challenges and opportunities, *IEEE Int. Things J.* 6 (5) (2019) 8897–8910.
- [122] C. Fiandrino, F. Anjomshoa, B. Kantarci, D. Kliazovich, P. Bouvry, J.N. Matthews, Sociability-driven framework for data acquisition in mobile crowdsensing over fog computing platforms for smart cities, *IEEE Trans. Sustain. Comput.* 2 (4) (2017) 345–358.
- [123] C. Fricker, F. Guillemin, P. Robert, G. Thompson, Analysis of an offloading scheme for data centers in the framework of fog computing, *ACM Trans. Model. Perform. Eval. Comput. Syst.* 1 (4) (2016) 1–18.
- [124] A.G. Ganek, T.A. Corbi, The dawning of the autonomic computing era, *IBM Syst. J.* 42 (1) (2003) 5–18
- [125] Garcia Lopez, P., A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere. 2015. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review* 45(5): 37–42.
- [126] Varghese, B., N. Wang, S. Barbhuiya, P. Kilpatrick, and D.S. Nikolopoulos. 2016. Challenges and opportunities in edge computing. In *Proceedings of the IEEE Conference on Smart Cloud*, 20–26.
- [127] Shi, W., J. Cao, Q. Zhang, Y. Li, and L. Xu. 2016. Edge computing: vision and challenges. *EEE Internet of Things Journal* 3(5): 637–646.

- [128] Hu, Y.C., M. Patel, D. Sabella, N. Sprecher, and V. Young. 2015. Mobile edge computing a key technology towards 5g. *ETSI White Paper* 11: 1–16.
- [129] Klas, G.I. 2015. Fog Computing and Mobile Edge Cloud Gain Momentum Open Fog Consortium, ETSI MEC and Cloudlets.
- [130] Cau, E., M. Corici, P. Bellavista, L. Foschini, G. Carella, A. Edmonds, and T.M. Bohnert. 2016. Efficient exploitation of mobile edge computing for virtualized 5g in epc architectures. In *4th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, (March 2016), 100–109.
- [131] Ahmed, A., and E. Ahmed. 2016. A survey on mobile edge computing. In *Proceedings of the 10th IEEE International Conference on Intelligent Systems and Control (ISCO 2016)*, Coimbatore, India.
- [132] Mahmud, M.R., M. Afrin, M.A. Razzaque, M.M. Hassan, A. Alelaiwi, and M. Alrubaian. 2016. Maximizing quality of experience through context-aware mobile application scheduling in cloudlet infrastructure. *Software: Practice and Experience* 46(11): 1525–1545. spe.2392.
- [133] Sanaei, Z., S. Abolfazli, A. Gani, and R. Buyya. 2014. Heterogeneity in mobile cloud computing: Taxonomy and open challenges. *IEEE Communications Surveys and Tutorials* 16(1): 369–392.
- [134] Bahl, P., R.Y. Han, L.E. Li, and M. Satyanarayanan. 2012. Advancing the state of mobile cloud computing. In *Proceedings of the third ACM workshop on Mobile cloud computing and services*, ACM, 21–28.
- [135] Satyanarayanan, M., G. Lewis, E. Morris, S. Simanta, J. Boleng, and K. Ha. 2013. The role of cloudlets in hostile environments. *IEEE Pervasive Computing* 12(4): 40–49.
- [136] Lee, W., K. Nam, H.G. Roh, S.H. Kim. 2016. A gateway based fog computing architecture for wireless sensors and actuator networks. In *18th International Conference on Advanced Communication Technology (ICACT)*, IEEE, 210–213.
- [137] Aazam, M., and E.N. Huh. 2015. Fog computing micro datacenter based dynamic resource estimation and pricing model for iot. In *IEEE 29th International Conference on Advanced Information Networking and Applications*. (March 2015), 687–694.

- [138] Jalali, F., K. Hinton, R. Ayre, T. Alpcan, and R.S. Tucker. 2016. Fog computing may help to save energy in cloud computing. *IEEE Journal on Selected Areas in Communications* 34(5): 1728–1739.
- [139] Zhu, J., D.S. Chan, M.S. Prabhu, P. Natarajan, H. Hu, F. Bonomi. 2013. Improving web sites performance using edge servers in fog computing architecture. In *Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on*, (March 2013), 320–323.
- [140] Zeng, D., L. Gu, S. Guo, Z. Cheng, and S. Yu. 2016. Joint optimization of task scheduling and image placement in fog computing supported software-defined embedded system. *IEEE Transactions on Computers* PP(99): 1–1.
- [141] Hong, K., D. Lillethun, U. Ramachandran, B. tenwälder, and B. Koldehofe. 2013. 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*, ACM, 15–20.
- [142] Nazmudeen, M.S.H., A.T. Wan, and S.M. Buhari. 2016. Improved throughput for power line communication (plc) for smart meters using fog computing based data aggregation approach. In *IEEE International Smart Cities Conference (ISC2)*, (Sept 2016), 1–4.
- [143] Aazam, M., and E.N. Huh. 2014. Fog computing and smart gateway based communication for cloud of things. In *Future Internet of Things and Cloud (FiCloud), International Conference on IEEE (2014)*, 464–470.
- [144] Cirani, S., G. Ferrari, N. Iotti, and M. Picone. 2015. The iot hub: A fog node for seamless management of heterogeneous connected smart objects. In *12th Annual IEEE International Conference on Sensing, Communication, and Networking-Workshops (SECON Workshops), IEEE (2015)*, 1–6.
- [145] Dsouza, C., G.J. Ahn, and M. Taguinod. 2014. Policy-driven security management for fog computing: Preliminary framework and a case study. In: *IEEE 15th International Conference on Information Reuse and Integration (IRI)*, (Aug 2014), 16–23.
- [146] Cardellini, V., V. Grassi, F.L. Presti, and M. Nardelli. 2015. On qos-aware scheduling of data stream applications over fog computing infrastructures. In *IEEE Symposium on Computers and Communication (ISCC)*, (July 2015), 271–276.

- [147] Yan, S., M. Peng, and W. Wang. 2016. User access mode selection in fog computing based radio access networks. In *IEEE International Conference on Communications (ICC)*, (May 2016), 1–6.
- [148] Gu, L., D. Zeng, S. Guo, A. Barnawi, and Y. Xiang. 2015. Cost-efficient resource management in fog computing supported medical cps. *IEEE Transactions on Emerging Topics in Computing* PP(99): 1–1.
- [149] Truong, N.B., G.M. Lee, and Y. Ghamri-Doudane. 2015. Software defined networking-based vehicular adhoc network with fog computing. In *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, (May 2015), 1202–1207.
- [150] Oueis, J., E.C. Strinati, S. Sardellitti, and S. Barbarossa. 2015. Small cell clustering for efficient distributed fog computing: A multi-user case. In *Vehicular Technology Conference (VTC Fall)*, IEEE 82nd. (Sept 2015), 1–5.
- [151] Hou, X., Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen. 2016. Vehicular fog computing: A viewpoint of vehicles as the infrastructures. *IEEE Transactions on Vehicular Technology* 65(6): 3860–3873.
- [152] Ye, D., M. Wu, S. Tang, and R. Yu. 2016. Scalable fog computing with service offloading in bus networks. In *IEEE 3rd international Conference on Cyber Security and Cloud Computing (CSCloud)*, (June 2016), 247–251.
- [153] Oueis, J., E.C. Strinati, and S. Barbarossa. 2015. The fog balancing: Load distribution for small cell cloud computing. In *IEEE 81st Vehicular Technology Conference (VTC spring)*, (May 2015), 1–6.
- [154] Faruque, M.A.A., and K. Vatanparvar. 2016. Energy management-as-a-service over fog computing platform. *IEEE Internet of Things Journal* 3(2): 161–169.
- [155] Shi, H., N. Chen, and R. Deters. 2015. Combining mobile and fog computing: Using coap to link mobile device clouds with fog computing. In *IEEE International Conference on Data Science and Data Intensive Systems*, (Dec 2015), 564–571.
- [156] Giang, N.K., M. Blackstock, R. Lea, and V.C.M. Leung. 2015. Developing IoT applications in the fog: A distributed dataflow approach. In *5th International Conference on the Internet of Things (IOT)*, (Oct 2015), 155–162.
- [157] Intharawijitr, K., K. Ida, and H. Koga. 2016. Analysis of fog model considering computing and communication latency in 5g cellular networks. In *IEEE International Conference on*

Pervasive Computing and Communication Workshops (PerCom Workshops), (March 2016), 1–4.

- [158] Peng, M., S. Yan, K. Zhang, and C.Wang. 2016. Fog-computing-based radio access networks: ssues and challenges. *EEE Network* 30(4): 46–53.
- [159] Hassan, M.A., M. Xiao, Q. Wei, and S.Chen. 2015. Help your mobile applications with fog computing. n *12th Annual EEEinternational Conference on Sensing, Communication, and Networking - Workshops (SECON Workshops)*, (June 2015), 1–6.
- [160] Zhang, W., B. Lin, Q. Yin, and T. Zhao. 2016. nfrastructure deployment and optimization of fog network based onmicrodc and lrp on ntegration. *Peer-to-Peer Networking and Applications* 1–13.
- [161] Deng, R., R. Lu, C. Lai, T.H. Luan, and H. Liang. 2016. Optimal workload allocation n fogcloud computing towards balanced delay and power consumption. *EEE nternet of Things Journal* PP(99): 1–1.
- [162] Do, C.T., N.H. Tran, C. Pham, M.G.R. Alam, J.H. Son, and C.S. Hong. 2015. A proximal algorithm for joint resource allocation and minimizing carbon footprint n geo-distributed fog computing. n *international Conference on nformation Networking (ICOIN)*, (Jan 2015), 324–329.
- [163] Aazam,M., M. St-Hilaire, C.H. Lung, and . Lambadaris. 2016. Pre-fog:IoT trace based probabilistic resource estimation at fog. n *13th EEE Annual Consumer Communications Networking Conference (CCNC)*, (Jan 2016), 12–17.
- [164] Datta, S.K., C. Bonnet, and J.Haerri. 2015. Fog computing architecture to enable consumer centric nternet of things services. n *international Symposium on Consumer Electronics (ISCE)*, (June 2015), 1–2.
- [165] Aazam, M., M. St-Hilaire, C.H. Lung, and . Lambadaris. 2016. Mefore: Qoe based resource estimation at fog to enhance qos nIoT. n *23rdinternational Conference on Telecommunications (ICT)*, (May 2016), 1–5.
- [166] Gazis, V., A. Leonardi, K. Mathioudakis, K. Sasloglou, P. Kikiras, and R. Sudhaakar. 2015.Components of fog computing n an ndustrial nternet of things context. n *12th Annual EEEinternational Conference on Sensing, Communication, and Networking - Workshops (SECONWorkshops)*, (June 2015) 1–6.

- [167] Gupta, H.,A.V.Dastjerdi, S.K.Ghosh, and R. Buyya. 2016. fogsim:Atoolkit for modeling and simulation of resource management techniques n nternet of things, edge and fog computing environments.
- [168] Redowan Mahmud, Ramamohanarao Kotagiri and Rajkumar Buyya. 2018 Fog Computing: A Taxonomy, Survey and Future Directions
- [169] Binh Mih Nguyen, Huynh Thi Thanh Binh, Tran The Anh, Do Bao Son. Evolutionary algorithms to optimize task scheduling problem for theIoT based Bag-of-Tasks application n cloud-Fog Computing Environment. Appl. Sci. 2019, 9, 1730.
- [170] Zeyu Sun, Lili Wei, Chen Xu, Tian Wang, Yalin Nie, Xiaofei Xing, Jianfeg Lu. An Energy Efficient Cross Layer Sensing Clustering method based on ntelligent Fog Computing n WSNs". *Special section on Green nternet of Things* (Volume -7, September, 2019).
- [171] M. Cannataro, P. H. Guzzi, G. Agapito, C. Zucco, and M. Milano, *Artificial intelligence in bioinformatics: from omics analysis to deep learning and network mining*. Elsevier, 2022.
- [172] G. Tradigo *et al.*, "A cloud based framework for Emergency Management: experiences in Calabria Region.," in *SEBD*, 2017, p. 275.
- [173] V. Lahoura *et al.*, "Cloud computing-based framework for breast cancer diagnosis using extreme learning machine," *Diagnostics*, vol. 11, no. 2, p. 241, 2021.
- [174] I. U. Oduh, S. Misra, R. Damaševičius, and R. Maskeli\=unas, "Cloud based simple employee management information system: a model for african small and medium enterprises," in *Proceedings of the International Conference on Information Technology \& Systems (ICITS 2018)*, 2018, pp. 115–128.
- [175] R. Li, X. Ye, F. Yang, and K.-L. Du, "ConvLSTM-Att: An Attention-Based Composite Deep Neural Network for Tool Wear Prediction," *Machines*, vol. 11, no. 2, p. 297, 2023.
- [176] K.-L. Du, M. N. S. Swamy, K.-L. Du, and M. N. S. Swamy, *Ant colony optimization*. Springer, 2016.
- [177] P. Zhang, M. Zhou, and G. Fortino, "Security and trust issues in fog computing: A survey," *Futur. Gener. Comput. Syst.*, vol. 88, pp. 16–27, 2018.

- [178] C. Singla, N. Mahajan, S. Kaushal, A. Verma, and A. K. Sangaiah, “Modelling and analysis of multi-objective service selection scheme in IoT-cloud environment,” *Cogn. Comput. Big Data Syst. Over IoT Fram. Tools Appl.*, pp. 63–77, 2018.
- [179] A. Verma and S. Kaushal, “Deadline constraint heuristic-based genetic algorithm for workflow scheduling in cloud,” *Int. J. Grid Util. Comput.*, vol. 5, no. 2, pp. 96–106, 2014.
- [180] D. S. Sisodia, “Ensemble Learning Approach for Clickbait Detection Using Article Headline Features,” *Informing Sci. Int. J. an Emerg. Transdiscipl.*, vol. 22, pp. 31–44, 2019.
- [181] D. S. Sisodia, R. Singhal, and V. Khandal, “A performance review of intra and inter-group MANET routing protocols under varying speed of nodes,” *Int. Jo urnal Electr. Comput. Eng.*, vol. 7, no. 5, pp. 2721–2730, 2017.
- [182] K. V. Arya and R. Gore, “Internet of things using software-defined network and cognitive radio network,” in *Sensing Techniques for Next Generation Cognitive Radio Networks*, IGI Global, 2019, pp. 312–328.
- [183] K. V. Arya, R. S. Bhadoria, and N. S. Chaudhari, *Emerging wireless communication and network technologies*. Springer, 2018.
- [184] P. Bedi, H. Kaur, and B. Gupta, “Trustworthy service provider selection in cloud computing environment,” in *2012 International Conference on Communication Systems and Network Technologies*, 2012, pp. 714–719.