ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

Power-Delay Trade-off for Optimum Data Storage in a Cloud-Fog-Mist Architecture

Niemah Izzeldin Osman[1], Mahmoud Al-Siddig Eisa[2] and Sabah Mohammed Imam[3]

ABSTRACT

The high growth in Internet-of-Things (IoT) data has resulted in the typical Cloud architecture creating a bottleneck that degrades the quality of Cloud-based services. Fog and Mist computing are proposed to offload the Cloud by storing and processing data locally and distributing tasks between network layers. The work in this paper investigates the distribution of data in a Cloud-Fog-Mist architecture and studies the influence on power consumption and network delay. A Mixed Integer Linear Programming (MILP) Model is proposed to minimize the power consumption of a Cloud-Fog-Mist architecture by optimizing the location of data files generated at the Mist layer. A second MILP model aims at minimizing network delay by storing data at its optimum location. In addition, we propose a Joint Power-Delay Optimization model to minimize network power consumption while guaranteeing not exceeding an acceptable maximum network delay. The results demonstrate that storing data at the Mist is the most power-efficient solution with maximum savings of 92.3% in power consumption. However, limited processing capabilities result in high processing delay even though data transmission delay within the Mist is negligible. On the other hand, Cloud computing remains the most efficient solution for delay-sensitive applications, regardless of the high energy consumed in the Cloud architecture. A power-delay trade-off can be achieved where power is minimized without exceeding acceptable delay.

Article information:

Keywords: Cloud Computing, Fog Computing, Mist Computing, Mixed Integer Linear Programming, Power Consumption

Article history:
Received: December 28, 2021
Revised: December 24, 2022
Accepted: January 21, 2023
Published: April 8, 2023 (Online)

DOI: 10.37936/ecti-cit.2023172.247297

1. INTRODUCTION

Cloud computing provides applications and services with high-speed servers and data centers with unlimited capacity. This enhances system performance in terms of bandwidth, latency, and cost. In addition, it provides platforms with high flexibility and scalability suitable for accommodating data-intensive applications including Internet-of-Things (IoT). Nevertheless, challenges in Cloud computing are becoming more evident [1]. Machine-to-Machine (M2M) connections are expected to reach 14.7 billion in 2023 [2]. This increase, coupled by the excellent technical advancements introduced by Fifth Generation (5G) mobile communication systems, paved the way for massive growth in fixed and mobile Cloud computing applications. Handling data volumes generated by these applications has resulted in increasing delay, congestion, and energy consumption in the centralized Cloud. Consequently, services suffer degradation of quality due to the delay-energy bottleneck. Furthermore, IoT consists of heterogeneous devices communicating using different protocols. Direct communication with these devices is impractical as it requires complex processing or involves unknown protocols. Additionally, recent and next-generation applications heavily rely on real-time sensitive data. The Cloud is susceptible to malfunction and technical issues, which implies that real-time data availability is not always guaranteed. Also, the Cloud relies on internet connection for communication with end users. The possibility of downtime reduces network reliability [3].

Fog computing is a paradigm that distributes data storage and processing of IoT-generated data toward the edge of the network [4]. It consists of many

[1,2,3] The authors are with Department of Computer Systems and Networks, Sudan University of Science and Technology, Khartoum, Sudan, E-mail: niema.osman@gmail.com, mahmoudssidig@gmail.com and sabzaid22@gmail.com

routers and/or switches that connect end devices. Fog nodes can simply partially store data and forward the remaining to the Cloud. Alternatively, the Fog enables data processing closer to end devices, reducing cost and response time [5]. The main advantages introduced by Fog computing include real-time processing, support for various IoT devices, and increased energy efficiency [6]. However, it complicates the network design and incurs costs for additional network devices [7].

Mist computing makes use of technology advancements in mobile and other end devices to implement data storage and processing in the access network where data is generated. Most data generated by IoT devices can be processed locally, limiting the use of Fog nodes and Cloud data centers [8]. This reduces latency, network traffic, threat to data, and power consumed on data transmission to/from Fog nodes and the Cloud [9]. Currently, mobile devices and sensors suffer from relatively limited storage capacity and processing capabilities compared to future IoT application requirements. Consequently, Mist computing is bounded by the limitations of access devices and is only suitable for lightweight applications with no high data processing demands. Nonetheless, with the high pace at which end-device hardware technology is advancing, Mist computing is considered a big-opportunity technology to cater for next-generation mobile and IoT applications [9][10].

End-to-end delay is considered a major network performance evaluation parameter. Its significance increases when time-sensitive applications are considered. Delay directly influences other important evaluation parameters, including Quality of Experience (QoE). Network delay is influenced by many factors, including the number of hops from source to destination, distances between network nodes, link capacities and data rates, and utilized storage technologies. On the other hand, power consumption is expected to become the new Internet bottleneck, raising concerns about the CO2 footprint of ICT [11]. The power consumption of the network is influenced by the energy efficiency of employed hardware devices, the amount of generated data and its storage location, communication technology, and the amount of required data processing. However, data storage and transmission is the main power consumer in today's Internet.

To the best of our knowledge, no prior work has considered the power efficiency and delay of Mist computing as an independent network layer capable of storing and processing data as a potential solution for offloading Cloud data centers and Fog servers. Therefore in this paper, we consider the storage and processing capabilities of Cloud-Fog-Mist computing and formulate Mixed Integer Linear Programming (MILP) Models to optimize allocating generated data among the three layers. Three MILP models are developed: i) a Power Minimization model to store data at the optimum location so as to minimize network power consumption, ii) a Delay Minimization model that minimizes delay by optimizing data location, and iii) a Joint Power-Delay model that minimizes network power consumption while maintaining the maximum acceptable delay. The objective is to find out the suitability of Mist computing to store and process IoT data. The success of Mist computing implies offloading Cloud servers, local management of data, and improved Quality-of-Service (QoS). The contributions of this work can be summarized as follows:

1) Demonstrate by extensive simulations the impact of Mist computing on power efficiency and delay as an emerging paradigm.

2) Identify the optimal data storage location in a Cloud-Fog-Mist computing network architecture such that power consumption and delay are minimized.

3) Investigate the influence of the number of Fog nodes, data volume, and the maximum allowed delay on data distribution.

The obtained results clearly demonstrate how Mist Computing can reduce the power consumption of the network when data is stored in Mist nodes, given that the mist layer fulfills delay constraints. They also appreciate the role of Cloud Computing in providing high-speed data processing suitable for interactive and real-time IoT services.

The rest of this paper is organized as follows: Section 2 explains related works in the research area. Section 3 describes the design of the considered Cloud-Fog-Mist architecture. Section 4 presents the proposed Power Minimized optimization model and demonstrates the outcomes of the model. Section 5 illustrates the Minimum Delay optimization model and displays the achieved results. Section 6 reveals the Joint Power-Delay Optimization model and the results of considered scenarios. Section 7 is the conclusions.

2. RELATED WORKS

The work in [12] proposed a model to reduce service time for four services in vehicular fog networks: traditional elastic services, interactive elastic services, hard real-time services, and soft real-time services. This is achieved by optimally allocating bandwidth between the Fog computing layer and the vehicular layer considering service type. Paper [13] proposed a dynamic load-balancing approach that allows mobile stations in overlapped coverage areas to select the best access point based on deadline miss and signal power. The proposed method resulted in better performance than no load balancing, workload balancing, and real-time traffic load balancing.

Several previous works have considered both the power consumption and delay of Cloud-Fog architectures [14][15][16]. In [14] researchers select power consumption and delay as indicators to decide the factor by which tasks are offloaded to the Fog. Queuing models are applied to model the execution process at mobile devices and the Fog node. A joint optimization problem that optimizes offloading probability and transmitted power is formulated with the objective to minimize the power consumption subject to the execution delay constraints. Results revealed that jointly optimizing offloading probability and transmit power achieves more power efficiency compared with optimizing a single factor. The study in [15] investigates the trade-off between power consumption and transmission delay in Fog-Cloud computing. A workload allocation problem is formulated to find the optimum location of workload such that power consumption is minimized and service delay limits are not exceeded. Results proved that accommodating some workload at the Fog decreases the communication delay of the Wide Area Network (WAN) at the expense of slightly increasing power consumption. The work presented in [16] evaluates the benefit of Fog computing for three scenarios in terms of power consumption and delay. The three scenarios for network data are i) static content generated in the Fog, ii) dynamic content generated in the Fog, and iii) content pre-downloaded from the Cloud. The evaluation used Broadcast Incremental Power (BIP) and Improved Genetic Algorithm (IGA) algorithms. Demonstrated results recommend using Fog computing for improved energy and delay efficiency considering all scenarios, especially under high traffic.

Even though these studies have evaluated both power consumption and delay, they assume two-tiered network architectures, where the Mist and Fog are viewed as one layer (the Edge network). This approach does not consider the cost of communication between the Mist and the Fog layer. The separation of these two layers becomes significant under high data traffic and when the data storage is available at both layers.

The work in [17] considers a three-layer network architecture where a Kruskal algorithm is minimizes communication delay between nodes. A Lagrange multiplier method reduces data transmission delay between edge nodes and the Cloud. The influence of the number of edge nodes on data processing delay is taken into account. Results showed that the three-layer architecture outperforms the single edge network and the Cloud computing approach in terms of network processing delay. Even though the paper provided detailed information about network delay, it did not consider power consumption.

3. CLOUD-FOG-MIST NETWORK ARCHITECTURE

The network architecture considered in this paper is shown in Fig.1 and consists of three layers:

1) Cloud Computing Layer: this layer contains high-speed core routers and massive data center equipment providing efficient processing and storage for unlimited application data.

2) Fog Computing Layer: the Fog is made up of connected routers and access points located closer to data sources and end devices, offering extended local storage for applications. It connects end devices to the Cloud.

3) Mist Computing Layer: this layer includes end devices such as mobile equipment and sensors. Mist devices are connected to a Fog node, and data is generated at this layer. The amount of generated data depends on the application type and the number of end devices.

Various applications can benefit from a Cloud-Fog-Mist architecture, including IoT and Wireless Sensor Networks (WSN) applications. End devices are deployed to collect and/or generate data, where different applications vary in the amount of generated data and computing capabilities (processing and storage). For instance, a fire detection application periodically transmits temperature readings from sensors. These readings are processed using a simple function to determine if a fire is detected. Such applications may not require large storage for data but are highly sensitive to delay.

In contrast, a surveillance application records continuous video through cameras. These types of applications entail huge data storage, high data processing for detection and recognition, and a relatively higher tolerance for delay. It is worth mentioning that the work in this paper does not consider data download. Data download includes the transmission of results from the location of data processing to the recipient's location and implies many options. The recipient might be a single user or a group of end devices, and the amount of data in each download request depends on the application type. Additionally, the demand for data download can be restricted to devices involved in data generation or can be available to other devices. The influence of download data on network power consumption and delay calls for further investigation.

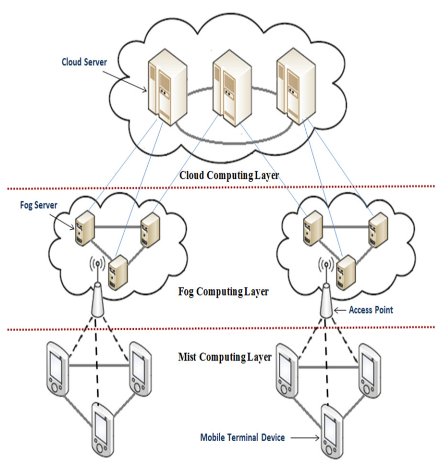

Fig.1: *Cloud-Fog-Mist Network Architecture.*

4. POWER MINIMIZATION MODEL

The aim of the Power Minimization MILP model is to distribute data storage among the three network layers (Cloud, Fog, and Mist) so that the total network power consumption is minimized. This section explains the design of the MILP model and demonstrates attained results.

4.1 Model Design

The model consists of parameters, variables, the objective function, and constraints. Following is the explanation of each component.

4.1.1 Input Parameters

The power minimization model defines several parameters shown in Table 1. The model defines parameters to specify the capacity of each network layer. In addition, it defines power consumption parameters for storage and data transmission between network layers.

Table 1: *Power Minimization Model Parameters.*

Parameter	Description
N	number of Fog nodes
M	number of Mists devices connected to each Fog node
$Data_{ij}$	generated data from Mist node j connected to Fog node i
$object\_size$	size of generated data objects
$fog\_servers$	Number of fog servers in each node
C_mist	capacity of a Mist device
C_fog	capacity of a Fog server
$idle\_mist$	Mist device idle power
$idle\_fog$	Fog server idle power
$idle\_cloud$	Cloud servers and data center idle power
P_store_mist	power to store a file in Mist
P_store_fog	power to store a file in Fog
P_store_cloud	power to store a file in Cloud
P_trans_mist	power to transmit a file to Mist
P_trans_fog	power to transmit a file to Fog
P_trans_cloud	power to transmit a file to Cloud

4.1.2 Model Variables

The model defines variables to hold storage and transmission data at each network layer. It also defines variables to retain the amount of data to store in each network layer to achieve the most power efficiency. Table 2 shows these variables.

Table 2: *Power Minimization Model Variables.*

Variable	Description
D_mist_{ij}	data stored in Mist ij
D_fog_i	data stored in Fog j
D_cloud	data stored in Cloud
P_mist	total power to store data in Mist
P_fog	total power to store data in Fog
P_cloud	total power to store data in Cloud
$P\_trans_{M\_M}$	power to transmit data from a node to another node in the Mist
$P\_trans_{M\_F}$	power to transmit data from a node in the Mist to the Fog
$P\_trans_{M\_C}$	power to transmit data from a node in the Mist to the Cloud

4.1.3 Power Consumption Analysis

The power minimization model defines equations (1) to (6) below to calculate the power consumption of storage and data transmission in the Mist, Fog, and Cloud:

1] Power consumption of stored data at each layer: Equations (1), (2), and (3) define the power consumption of storing data at the Mist, Fog and Cloud layers, respectively.

$$P\_mist = \sum_{i=1}^{N}\sum_{j=1}^{M} D\_mist_{ij} \times P\_store\_mist + (idle\_mist \times M \times N) \quad (1)$$

$$P\_fog = \sum_{i=1}^{N} D\_fog_i \times P\_store\_fog + (idle\_fog \times N) \quad (2)$$

$$P\_cloud = D\_cloud \times P\_store\_cloud + idle\_cloud \quad (3)$$

2] Power consumption of data transmission from Mist to network layers: Equation (4) calculates the power consumed in transmitting data from one node to another node in the Mist. Equation (5) finds the required power to transmit data generated in the Mist to the Fog. Equation (6) determines the power consumption of transmitting data from the Mist to the Cloud.

$$P\_trans_{M\_M} = \sum_{i=1}^{N}\sum_{j=1}^{M} D\_mist_{ij} \times P\_trans\_mist \quad (4)$$

$$P\_trans_{M\_F} = \sum_{i=1}^{N} D\_fog_i \times P\_trans\_fog \quad (5)$$

$$P\_trans_{M\_C} = D\_cloud \times P\_trans\_cloud \quad (6)$$

4.1.4 Objective Function

The objective of the Power Minimization model, shown in Equation (7), is to minimize the power of storing data in the network architecture by minimizing the power of data transmission between layers and the power to store data in each layer.

$$Minimize: \; P\_cloud + P\_fog + P\_mist + P\_trans_{M\_M} + P\_trans_{M\_F} + P\_trans_{M\_C} \quad (7)$$

4.1.5 Constraints

The model defines a set of constraints. Constraint (8) ensures the total data is distributed between network layers. Constraint (9) ensures the data stored in each Fog node does not exceed the maximum capacity of the Fog node. Constraint (10) ensures data stored in each Mist device does not exceed its maximum capacity.

$$\sum_{i=1}^{N} \sum_{j=1} Data_{ij}$$

$$\sum_{i=1}^{N} \sum_{j=1} D\_mist_{ij} + \sum_{i=1}^{N} D\_fog_i + D\_cloud \quad (8)$$

$$D\_fog_i \leq C\_fog \times fog\_servers, \forall \; i \in N \quad (9)$$

$$D\_mist_{ij} \leq C\_mist, \forall \; i \in N, \forall \; j \in M \quad (10)$$

4.2 Power Minimization Model Evaluation

The Power Minimization Model is evaluated using the Cloud-Fog-Mist network architecture explained in Section 2. The model is implemented using AMPL/CPLEX optimization tool run on an HP Laptop with 3GHz CPU and 8GB RAM. The objective function and constraints are applied to the model file, and input data is added to the data file. The optimum solution is found using the dual simplex method. The following subsections present input values and demonstrate by results the model's outputs.

4.2.1 Input Values and Model Implementation

This model determines the amount of data stored in the Mist, Fog, and Cloud layers (D_mist_{ij}, D_fogi, and D_cloud) to consume the least possible power. The Mist deploys 20 Samsung Galaxy Tab 3 Lite devices for data generation. These devices communicate with each other using Wi-Fi interfaces. Each Mist device is assumed to generate 2000 objects. Three object sizes are selected to evaluate the influence of small, moderate, and large objects on attained results: a) 5MB, b) 10MB, and c) 12.5MB. A Raspberry Pi 3 is utilized as a Fog gateway that connects Mist devices to the Cloud. Five Fog nodes are connected to the Cloud, where each Fog node has 1 to 10 servers. The Cloud has 5 CRS-3 core routers and a single Catalyst 6509 data center switch, where the data center is assumed to have unlimited storage capacity. Table 3 shows the values of input parameters used to run the model. The storage and power consumption parameter values of the Samsung Galaxy Tab 3 Lite, Raspberry Pi 3, and Cloud switch and router are derived from [18] and [19].

Table 3: *Power Minimization Model Variables.*

Parameter	Values
N	5
M	20
$Data_{ij}$	2000 objects
$object\_size$	5MB, 10MB, 12.5MB
$fog\_servers$	1-10
C_mist	32 GB
C_fog	32 GB
$idle\_mist$	4.7 W
$idle\_fog$	180 W
$idle\_cloud$	12818 W
P_store_mist	0.1 W
P_store_fog	0.19 W
P_store_cloud	1.7 W
P_trans_mist	0.13 W
P_trans_fog	1.2 W
P_trans_cloud	1.8 W

4.2.2 Data Distribution for Minimum Power

Fig.2 throughout Fig.5 show the outputs of running the Power Minimization MILP model. Fig.2, Fig.3, and Fig.4 demonstrate the distribution of data among the three layers using different Fog servers when the size of generated data files is 5MB, 10MB, and 12.5MB, respectively. When the size of data files is small (5MB), the storage capacity of the mist layer is sufficient to accommodate generated data. Therefore in Fig.2, 100% of data is stored in the Mist since it consumes the least power to store data in this network architecture. This result is observed under all assumed Fog servers (1 to 10).

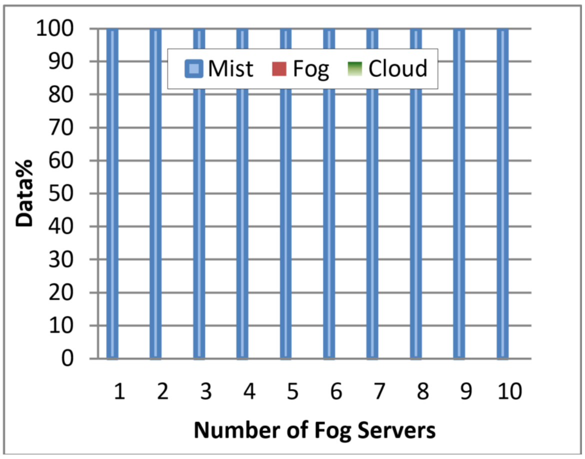

Fig.2: *Data Distribution when File=5MB.*

In the case where generated data is larger than the capacity of the Mist (10MB and 12.5MB), the model

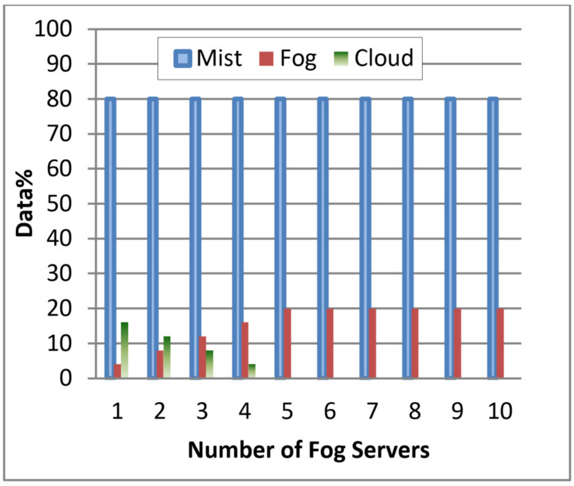

Fig.3: *Data Distribution when File=10MB.*

attempts to store the data in the Mist first. Then the model populates data in the second most energy-efficient layer, the Fog. Depending on the storage capacity of the Fog, the data not stored in the Fog will be transmitted to the Cloud. In Fig.3 and Fig.4, when the number of Fog servers is small, the Fog capacity is limited, resulting in more significant amounts of data stored in the Cloud. As the storage capacity of the Fog increases, the percentage of data stored in the Fog rises while it drops in the Cloud. If generated data is moderate (10MB), the Mist and Fog are sufficient to store all data when Fog servers are greater than 3 (Fig.3). Nevertheless, when generated data is high (12.5MB), the Cloud is still required to retain remaining data as Fig.4 shows. In other words, the MILP model stores data starting from the layer that consumes the least power to the layer that consumes the highest power, as power consumption is the decisive factor for data storage.

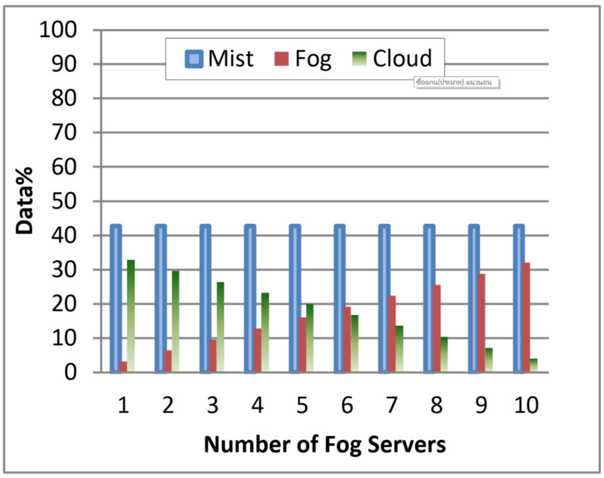

Fig.4: *Data Distribution when File=12.5MB.*

4.2.3 The Influence of Generated Data and the Number of Fog Servers on Power Consumption

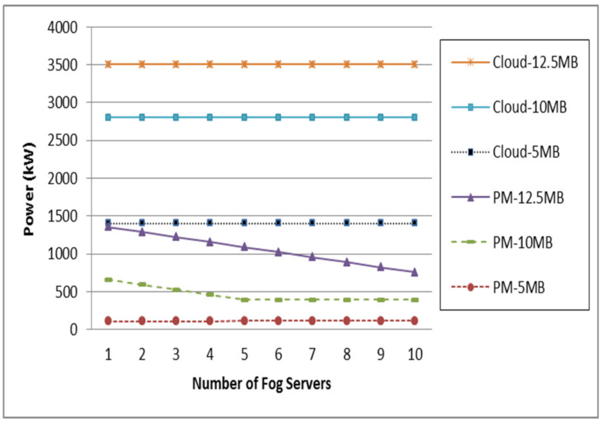

Fig.5: *The Influence of File Sizes and Fog Servers on Power Consumption for the Typical Cloud and the Power Minimization Model.*

The Power Minimization model is executed to evaluate the power consumption of the network when generated file sizes are (5MB, 10MB, and 12.5MB) under different Fog servers. The resultant power consumption is shown in Fig.5. In addition, the power consumption of the network is evaluated under the same scenarios when typical mobile Cloud computing is deployed. That is when all data generated at end devices is stored in the Cloud. When generated data files are small (5MB), all data is stored in the Mist. The proposed MILP model saves a maximum of 92.3% of power (average 92.0%) compared to the power consumption when all data is stored in the Cloud. Increasing the number of Fog servers does not influence power consumption, as the Fog layer is not utilized under this scenario.

When generated data files are moderate (10MB), the power consumption of the network decreases when utilizing more servers at the Fog since more data is stored closer to end devices in the more power-efficient layers. This trend continues until the combined Mist-Fog storage capacity is sufficient to accommodate all generated data, resulting in maximum power saving of up to 86.1% (average 83.6%). Further increasing the number of Fog servers does not influence network power consumption.

Under large data files (12.5MB), the storage capacity of the Mist and Fog is insufficient to accommodate all generated data. Therefore, increasing the number of Fog servers leads to storing more data locally, which consequently causes more power savings. Maximum savings in power consumption achieved by the proposed MILP model under this scenario are up to 78.4% (average 69.8%).

5. MINIMUM DELAY MILP MODEL

The aim of the Delay Minimization model is to store data at the location that will result in the minimum network delay. The model takes into account processing delay and transmission delay. Therefore delay is defined as the time from the start of saving data to the end of data processing. This is calculated by finding the summation of the transmission delay from the Mist to the storage location and the processing delay at that location. For simplicity, data storage and processing occur at the same layer in the network architecture; no data is stored in one location and transmitted for processing in another. Hence, each generated data file is stored in exactly one location. This section explains the proposed Minimum Delay MILP model, reveals input data, and demonstrates attained results.

5.1 Model Design

The Minimum Delay MILP model consists of parameters, variables, the objective function, and constraints. Following is the explanation of each component.

5.1.1 Input Parameters

The Minimum Delay MILP model defines a number of parameters shown in Table 4. The defined capacity parameters are similar to those specified by the Power Minimization MILP model. In addition, the Minimum Delay model defines processing delay parameters at each network layer and transmission delay parameters for transmitting data between network layers.

Table 4: *Minimum Delay Model Parameters.*

Parameter	Description
N	number of Fog nodes in the network
M	number of Mists nodes in each Fog node
$Data_{ij}$	generated data from Mist node j connected to Fog node i
C_mist	capacity of a Mist device
C_fog	capacity of a Fog server
T_mist	Mist transmission delay
T_fog	Fog transmission delay
T_cloud	Cloud transmission delay
Hz_mist	Mist device processor speed
Hz_fog	Fog server processor speed
Hz_cloud	Cloud server processor speed

5.1.2 Model Variables

Similar to the Power Minimization MILP model, the Minimum Delay MILP model defines variables to hold the total storage and transmission data at each network layer. In addition, it defines variables to track processing and transmission delay at each network layer. Table 5 explains these variables.

Table 5: *Minimum Delay Model Variables.*

Variable	Description
D_mist_{ij}	data stored in Mist ij
D_fog_i	data stored in Fog j
D_cloud	data stored in Cloud
PD_mist	Mist processing delay
PD_fog	Fog processing delay
PD_cloud	Cloud processing delay
TD_mist	delay of transmitting a file from a node to another node in the Mist
TD_fog	delay of transmitting a file from the Mist to the Fog
TD_cloud	delay of transmitting a file from the Mist to the Cloud
$Total\_Delay$	total network delay

5.1.3 Delay Calculations

The Delay Minimization MILP model defines the following equations (11) to (17) to calculate processing delay, transmission delay and total network delay:

1] Processing delay of stored data at each layer:

Equations (11), (12), and (13) define the processing delay of the Mist, Fog, and Cloud layers, respectively.

$$PD\_mist = \sum_{i=1}^{N}\sum_{j=1}^{M} D\_mist_{ij}/Hz\_mist \tag{11}$$

$$PD\_fog = \sum_{i=1}^{N} D\_fog_i/(Hz\_fog \times fog\_serveres) \tag{12}$$

$$PD\_cloud = D\_cloud/Hz\_cloud \tag{13}$$

2] Delay of data transmission from Mist to network layers:

Equation (14) calculates the delay in transmitting data from one node to another node in the Mist. Equation (15) finds the delay of transmitting data generated in the Mist to the Fog. Equation (16) determines the delay in transmitting data from the Mist to the Cloud.

It is worth mentioning that the transmission data rate varies from one network to another, where the links connecting the Fog and Cloud have the highest data rate. This can be employed when data aggregation at the Fog is considered where data can be transmitted in parallel from Fog to Cloud. Nevertheless, the total delay in transmitting a particular data file from Mist to Cloud in this scenario is not reduced since end-to-end delay is influenced by the limited bandwidth at the network edge.

$$TD\_mist = \sum_{i=1}^{N}\sum_{j=1}^{M} D\_mist_{ij} \times T\_mist \tag{14}$$

$$TD\_fog = \sum_{i=1}^{N} D\_fog_i \times T\_fog \tag{15}$$

$$TD\_cloud = D\_cloud \times T\_cloud \tag{16}$$

3] Calculation of the total delay of the network:

$$Total\_Delay = PD\_cloud + PD\_fog + PD\_mist + TD\_cloud + TD\_fog + TD\_mist \tag{17}$$

5.1.4 Objective Function

The objective of the Delay Minimization model is to minimize the total delay of the network architecture, calculated in Equation (17), by minimizing the delay in transmitting data between Mist, Fog, and Cloud layers and the delay of processing data stored in each layer. Equation (19) shows the objective function:

$$Minimizez : Total\_Delay \tag{18}$$

5.1.5 Constraints

Similar to the Power Minimization model, this model defines Constraints (8), (9), and (10) explained in the previous section.

5.2 Minimum Delay Model Evaluation

5.2.1 Input Values and Model Implementation

The objective of this model is to find the optimum amount of data to store in the Mist, Fog, and Cloud (D_mist_{ij}, D_fog_{ij}, and D_cloud_{ij}) such that the total network delay is minimum. Table 6 shows the values of input parameters used to run the model. The network architecture is considered with 20 and 30 Mist devices to evaluate the distribution of data for different processing capabilities at the Mist layer. It is also evaluated under different Fog servers (1 to 10). The devices deployed in each network layer are similar to those considered in the Power Minimization MILP model. The processing and transmission delay parameter values are derived from [17], where the transmission delay of the Fog and Mist are negligible.

5.2.2 Data Distribution for Minimum Delay

The Delay Minimization MILP model is run with Mist=20 devices to exemplify a current Mist layer with moderate computing capabilities and Mist=30

Table 6: *Minimum Delay Model Parameter Values.*

Parameter	Values
N	5
M	20, 30
$Data_{ij}$	2000 objects
$object\_size$	5MB, 10MB, 12.5MB
$fog\_servers$	1-10
C_mist	32 GB
C_fog	32 GB
T_mist	0 ms
T_fog	0 ms
T_Cloud	0.0001 ms
Hz_mist	0.4 GHz
Hz_fog	1.5 GHz
Hz_cloud	50 GHz

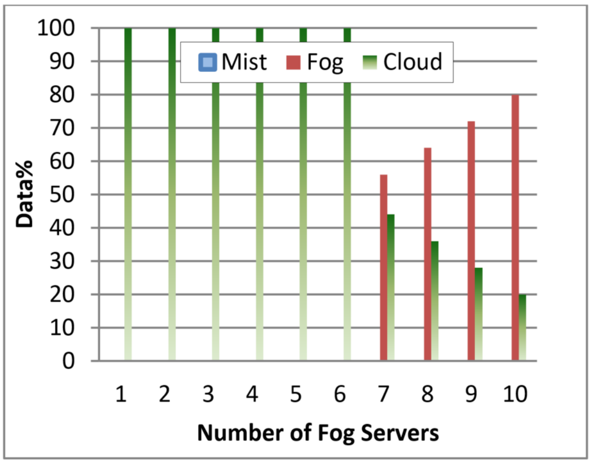

Fig.6: *Data Distribution when Mist=20 and File=5MB.*

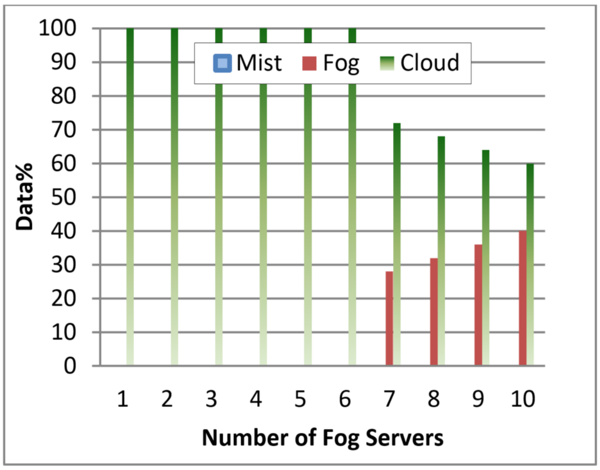

Fig.7: *Data Distribution when Mist=20 and File=10MB.*

devices to consider a Mist layer with improved capabilities exemplifying future Mist computing. When the number of Mist devices and Fog servers is small (Mist=20 and Fog servers < 7), the total Mist-Fog processing capabilities are low. In this scenario, the Cloud is considered the most efficient layer in the architecture, and therefore all generated data is stored in the Cloud. This is clearly observed in Fig.6 throughout Fig.8. These figures also demonstrate data distribution when the number of Fog servers is large (≥ 7). In this case, the Fog processing capabilities increase, and it becomes more efficient to store data closer to the source. Consequently, data is populated in the Fog layer where the percentage of this data is relative to the storage capacity of the Fog.

In contrast, when the number of Mist devices is large (Mist=30), the mist can process data at a higher speed. Therefore the model attempts to store generated data in the Mist, as can be seen in Fig.9 throughout Fig. 11. When the file size is small (5MB), data is stored only in the Mist, as apparent in Fig. 9. The exception is when the processing capabilities of the Fog exceed that of the Mist; when the number of Fog servers is vast (9 and 10). In this situation, the data is stored in both the Mist and the Fog. However, as the amount of generated data increases (10MB and 12.5MB), the capacity of the edge network becomes insufficient to accommodate all generated data. When the Fog servers are a few (< 7), the Cloud is utilized to store the remaining data, which is not stored in the Mist. For more prominent number of Fog servers (≥ 7), data that is not stored in the Mist is stored in the Fog until it reaches its total capacity, and the rest of the data is stored in the Cloud, as shown in Fig. 11.

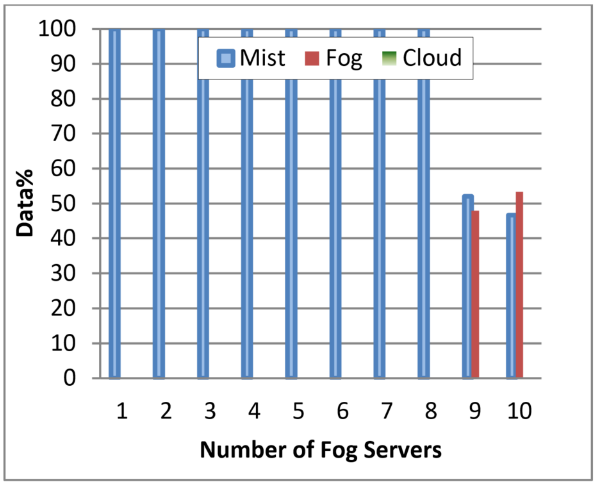

Fig.9: *Data Distribution when Mist=30 and File=5MB.*

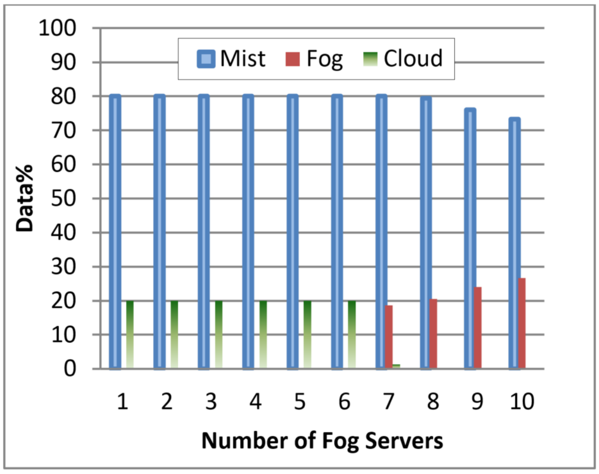

Fig.10: *Data Distribution when Mist=30 and File=10MB.*

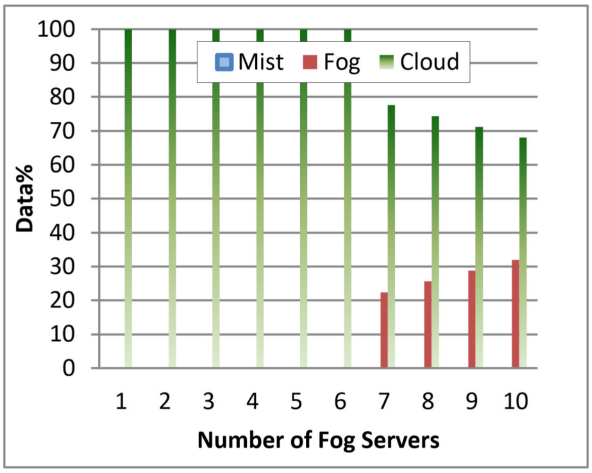

Fig.8: *Data Distribution when Mist=20 and File=12.5MB.*

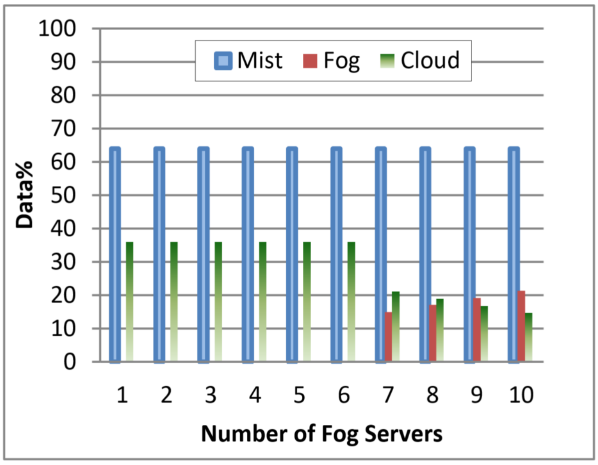

Fig.11: *Data Distribution when Mist=30 and File=12.5MB.*

5.2.3 The Influence of File Sizes and Fog Nodes on Network Delay

Fig. 12 and Fig. 13 demonstrate the influence of file sizes and the number of Fog servers on delay when Mist=20 and Mist=30, respectively. They also compare resultant network delay under a typical Cloud and when using the Minimum Delay MILP model. Obviously, under the same processing capabilities, increasing the size of files causes an increase in delay, as more data requires additional processing. Increasing the number of Fog servers does not influence network delay for values < 7 since data is stored in the Cloud, whereas Fog servers are not utilized in the data processing. When Fog servers are ≥ 7, data files are stored in the Fog. Therefore, increasing the number of Fog servers implies more processing capabilities which leads to the delay gradually decreasing.

When Mist=20, the maximum reduction in network delay introduced by the Minimum Delay MILP model is up to 26.9% (average 5.9%), 13.4% (average 2.9%), and 10.7% (average 2.3%) when file sizes are 5MB, 10MB, and 12.5MB, respectively. These savings increase to 25.9% (average 18.4%), 21.5% (average 15.4%), and 18.1% (average 12.5%) when file sizes are 5MB, 10MB, and 12.5MB, respectively, considering Mist=30.

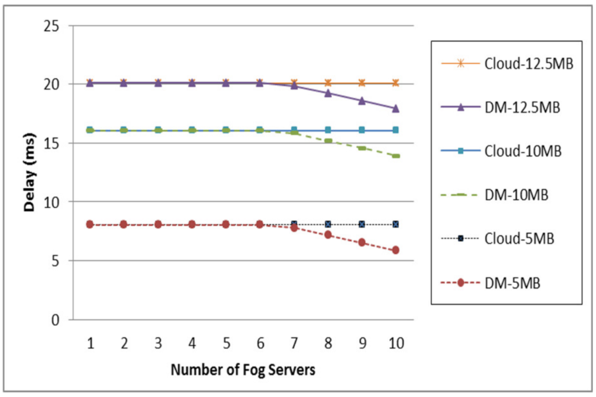

Fig.12: *Influence of File Sizes and Fog Servers on Delay when Mist=20 for Typical Cloud and Delay Model .*

The obtained results imply that when the Mist is equipped with devices of high-speed processing capabilities, the most efficient option is to store data locally. This is also true when the application requires no or minimal processing. In contrast, in the case where the application requires high data processing, the Cloud remains the most efficient option for data storage. Advancements in smart mobile and sensing devices promote the success of Mist computing as an independent layer that reduces network traffic without efficiency drawbacks.

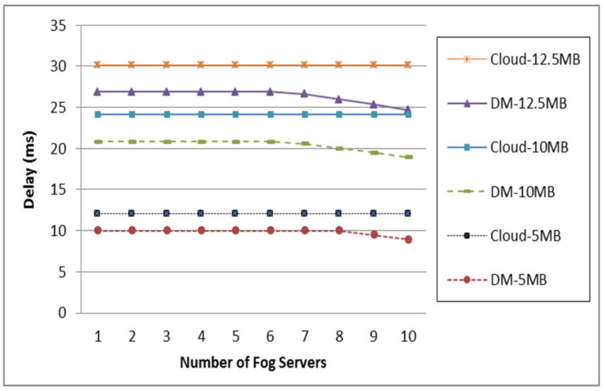

Fig.13: *Influence of File Sizes and Fog Servers on Delay when Mist=30 for Typical Cloud and Delay Model.*

6. JOINT POWER-DELAY OPTIMIZATION MODEL

The Joint Power-Delay Optimization model merges the concepts of both Power Minimization and Minimum Delay MILP models. The model's objective is that, given a maximum acceptable network delay defined by the application, it determines the optimum data storage location such that power consumption is minimized without exceeding the allowed delay. For this reason, the Joint Power-Delay Optimization MILP model defines all parameters, variables, constraints, and equations defined in the previously mentioned MILP models.

6.1 Model Modified Items

In addition to all items defined in the Power Minimization and the Minimum Delay MILP models, the Joint Power-Delay Optimization MILP model defines the following:

6.1.1 Parameters:

Table 7: *Parameter Values of Joint Power-Delay Optimization Model.*

Parameter	Description
Max_Delay	maximum acceptable delay defined by the application

The Joint Power-Delay Optimization model defines the parameter Max_Delay as shown in Table 7.

6.1.2 Objective Function:

The objective of this model is to minimize the total power consumption of the network without exceeding the maximum acceptable delay. Therefore, the objective function of the model is similar to the objective function of the Power Minimization model stated in Equation (7).

6.1.3 Constraints:

In addition to Constraints (8), (9), and (10) ensured by previous models, the Joint Power-Delay Optimization MILP model defines Constraint (19), which ensures that the total network delay does not exceed the maximum acceptable delay specified by the application.

$$Total\_delay \leq Max\_Delay \tag{19}$$

6.1.4 Input Parameter Values

The Joint Power-Delay Optimization model is executed using input values similar to those considered in the Power Minimization and Minimum Delay MILP models. Table 8 lists the maximum acceptable delay under file sizes 5MB, 10MB, and 12.5MB, respectively. These ranges are chosen to clearly demonstrate the shift in optimum data storage location from one network to another under minor changes in maximum acceptable delay.

Table 8: *Parameter Values of Joint Power-Delay Optimization Model.*

Parameter	Value
Max_Delay	8-10, 16-24, 20-28 seconds

6.2 Joint Power-Delay Optimization Model Evaluation

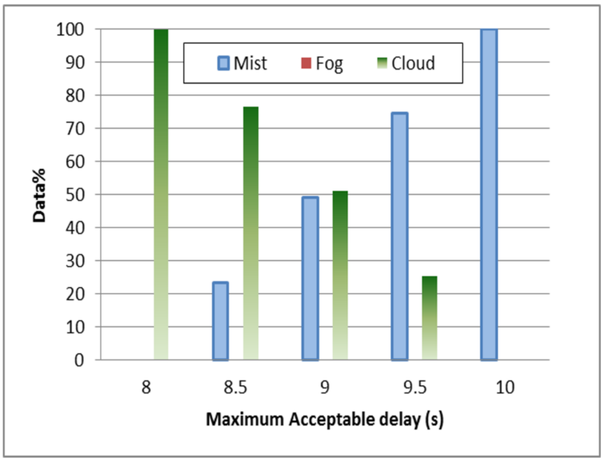

Fig.14: *Data Distribution when File Size=5MB and Fog Servers=1 for Different Values of Max. Acceptable Delay.*

From previous results obtained by the Power Minimization model and the Minimum Delay model, it can be observed the most power-efficient layer is the Mist, while storing data in the Cloud offers the least delay. This MILP model balances power and delay and finds the best location for data storage to provide the best QoS for applications. This model verifies the amount of data to be stored in the Mist, Fog,

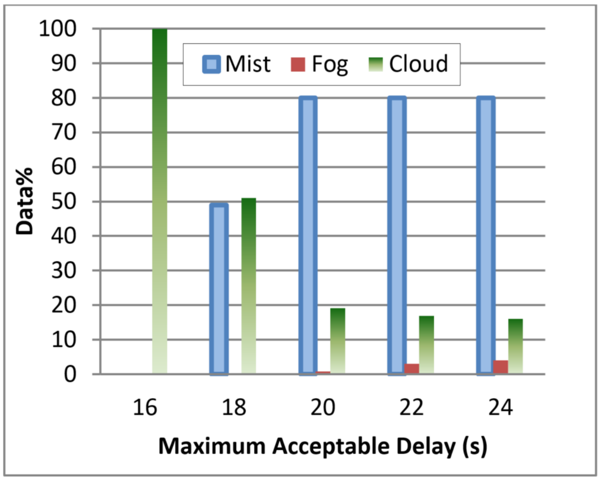

Fig.15: *Data Distribution when File Size=10MB and Fog Servers=1 for Different Values of Max. Acceptable Delay.*

and Cloud layers (D_mistij, D_fog_{ij}, and D_cloud_{ij}) such that the power consumption is minimized while the maximum accepted delay allowed by the application is not exceeded.

Results of the model are demonstrated in Fig. 14 throughout Fig. 20. Fig.14, Fig.15, and Fig.16 illustrate data distribution when 1 Fog server is utilized, and file sizes are 5MB, 10MB, and 12.5MB, respectively. When delay sensitivity is high, the model stores all data in the Cloud to guarantee that network delay is below the maximum acceptable value. As this value grows, the model enjoys more flexibility to find more power-efficient solutions. Additional data is gradually stored in the Mist to reduce network power consumption until the Mist is fully populated with data under higher delay values. The amount of data stored in the Fog under this scenario is negligible.

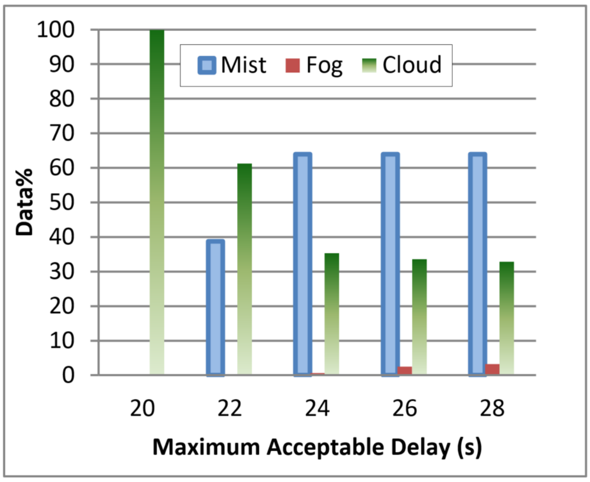

Fig.16: *Data Distribution when File Size=12.5MB and Fog Servers=1 for Diff. Values of Max. Acceptable Delay.*

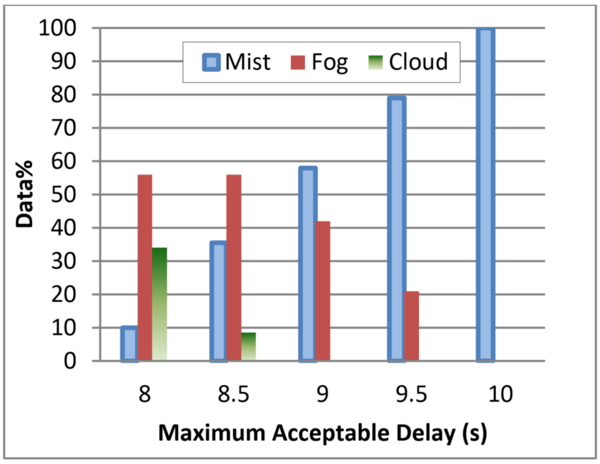

Fig.17: *Data Distribution when File Size=5MB and Fog Servers=7 for Different Values of Max. Acceptable Delay.*

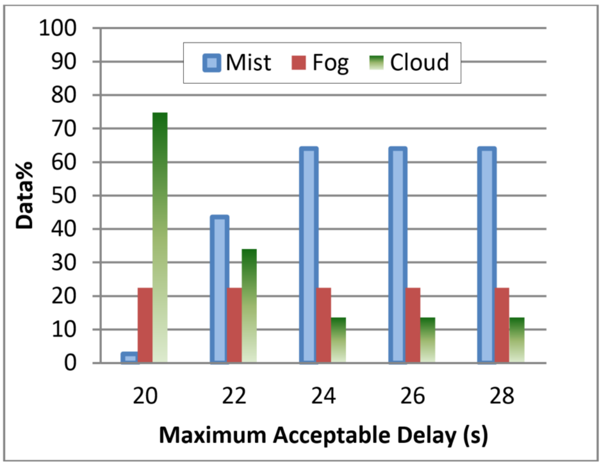

Fig.19: *Data Distribution when File Size=12.5MB and Fog Servers=7 for Diff. Values of Max. Acceptable Delay.*

In Fig.17, Fig.18, and Fig.19, the Fog is considered with 7 servers and file sizes are 5MB, 10MB, and 12.5MB, respectively. In this scenario, the Fog results in lower power consumption than the Cloud. Therefore, when the maximum acceptable delay is low, the model stores the data in the Fog first before storing the remaining data in the Cloud. When the maximum acceptable delay is high, the model's behavior is similar to the previous scenario.

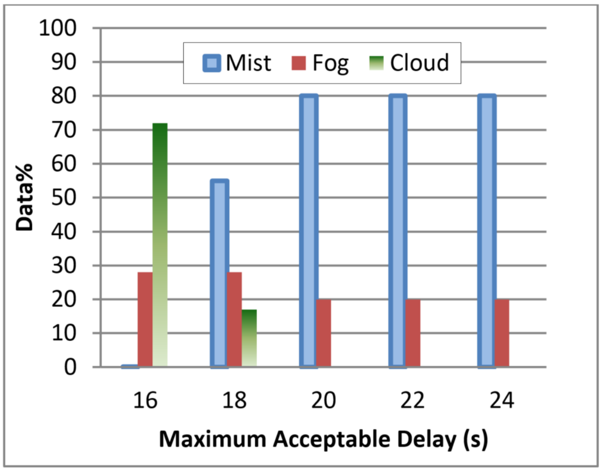

Fig.18: *Data Distribution when File Size=10MB and Fog Servers=7 for Different Values of Max. Acceptable Delay.*

6.3 The Influence of Maximum Acceptable Delay on Power Consumption

The Joint Power-Delay Optimization MILP model is run for different scenarios to signify applications of various delay constraints under moderate and high Fog capabilities. The power consumption results are demonstrated in Fig.20. From Fig.20 we make the following observations:

1. Generally, increasing the number of Fog servers results in more power efficiency for all values of the maximum acceptable delay. In other terms, provisioning additional storage and processing resources at the edge of the network saves considerable power.
2. Delay-tolerant applications can achieve more power efficiency since more data distribution options are accessible to them with no QoS violations. In other words, data can be stored in the Mist, Fog or Cloud while maintaining acceptable delay. This note should be considered when provisioning resources for different classes of service.
3. When network resources are accurately allocated, having more data to store does not necessarily imply increased delay or more power consumption. This observation is important for emerging paradigms such as Mist computing, as they are expected to cater to future IoT applications.
4. When power consumption and network delay are concerned, there exists an optimal data storage solution where these factors are decreased. The optimal solution depends on the following:
   a. Storage and processing capabilities of network devices: Improvements in device processing capabilities reduce processing delay, consequently favoring storing data in the related network layer.
   b. Power consumption parameters of network devices: Variation in the power efficiency of network devices leads to alternate optimum data storage distribution, since the models select the most power-efficient solution that complies with delay constraints.
   c. Amount of generated data: The amount of data produced by applications decides the storage location, mainly when limited storage capacities are available. Data processing requirements also determine the best storage location as it is

best to store data closest to the data processing site. The work in this paper considers uniform data generation for all Mist devices. An enhanced study may assume different data distribution and processing requirements. This requires further investigation.

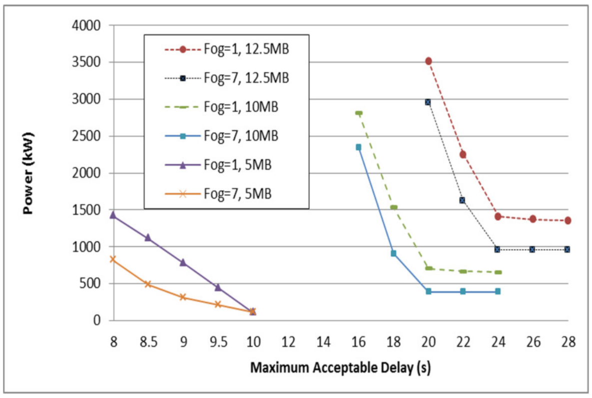

Fig.20: *Power Consumption of Network for Diff. Values of File Sizes, Fog Servers, and Max. Acceptable Delay.*

In general, Mist provides power efficiency but suffers from a higher delay. The Cloud consumes higher energy but outperforms the Mist in processing speed. The resultant trade-off is governed by delay constraints, where more delay tolerance pushes data toward the network's edge to save more power. In contrast, delay-sensitive interactive data that requires processing is moved toward the Cloud.

It is worth mentioning that the experiments conducted in this work consider files of fixed sizes. In practice, the sizes of generated data files vary and follow a particular distribution concerning service type [20]. Furthermore, the amount of generated data differs from one end device to another. In this work, for simplicity, all devices are assumed to generate the same amount of data. A more complex model can be developed where Mist devices generate different numbers of data files, and the sizes of these files follow a particular distribution. This is beyond the scope of our evaluation and requires further investigation.

Results attained in this evaluation can be utilized to provide lower limits on expected power consumption and delay resultant values, as additional factors such as power consumption of cooling and cabling are not considered. Nevertheless, data distribution over the three network layers is expected to follow the same trend obtained by the proposed models, allowing actual values to be derived.

7. CONCLUSIONS

This paper has investigated the power consumption and delay of a Cloud-Fog-Mist network architecture. It has explored the placement of generated data such that power consumption and delay are minimized. Three Mixed Integer Linear Programming (MILP) models have been developed: the Power Minimization model, which finds the optimum location to store data, so that network power consumption is minimized, the Minimum Delay Model, which stores generated data such that network delay is minimized, and the Joint Power-Delay Optimization model, which minimizes power consumption of data storage without violating a maximum acceptable delay. The proposed models specify the ideal location(s) to store data in the network given the amount of generated data, service delay constraints, and hardware capabilities. Additionally, they can be used to evaluate the feasibility of employing the edge layer (Fog and Mist) for data storage and determine hardware requirements in the Fog layer such that power consumption is minimized.

After extensive experiments, the results revealed that given the advanced processing capabilities of the Cloud, it continues to provide high-speed data processing for IoT applications successfully. This is most suitable with interactive and real-time services. In addition, the Cloud accommodates applications with massive data storage requirements that exceed the current storage capabilities of the edge network. The main drawback of the Cloud is the associated excessive power consumption compared to the power consumed when utilizing the edge network for data storage and processing.

Mist computing provides a power-efficient solution for local data storage and processing, most effective for delay-tolerant applications, as the processing capabilities of the Mist network remain modest. Mist networking can thus be considered a successful solution for IoT applications with more flexibility on time constraints. A hybrid Mist-Fog-Cloud architecture can be utilized to distribute the location of stored data concerning application type and the class of service. This saves considerable power without compromising quality.

Our future work includes evaluating the proposed models taking into account download network traffic and evaluate how it influences attained results. This involves investigating a distributed approach where data generated in a particular Mist node is requested in another Mist node connected through multiple Fog nodes. Another future direction is to evaluate networks with wirelessly transmitted large data files to explore possible improvements in network efficiency for Multimedia IoT (M-IoT).

References

[1] R. Ara, M. Rahim, S. Roy, K. Uzzal and U. Prodhan, "Cloud Computing: Architecture, Services, Deployment Models, Storage, Benefits and Challenges," *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 4, no.4 pp. 837-842, Jun. 2020.

[2] Cisco.(2022, Aug. 16). *Cisco Annual Internet Report (2018–2023) White Paper*[online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html .

[3] D. Bhuriya and A. A. Sharma,, "Study on Pros, Cons, and Application of Cloud Computing," *International Journal of Research and Analytical Reviews*, vol. 6, no.2, pp. 959-964, 2019.

[4] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow and P. A. Polakos, "A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges," in *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 416-464, Firstquarter 2018.

[5] T. Pfandzelter and D. Bermbach, "IoT Data Processing in the Fog: Functions, Streams, or Batch Processing?," *2019 IEEE International Conference on Fog Computing (ICFC)*, Prague, Czech Republic, pp. 201-206, 2019.

[6] A. A. Abi Sen and M. Yamin, "Advantages of using Fog in IoT applications," *International Journal of Information Technology*, vol. 13, pp. 829-837, 2021.

[7] M. K. Saroa and R. Aron, "Fog Computing and Its Role in Development of Smart Applications," *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, Melbourne, VIC, Australia, pp. 1120-1127, 2018.

[8] Yogi, M. et al. "Mist Computing: Principles, Trends, and Future Direction," *ArXiv abs/1709.06927*, 2017.

[9] S. Ketu, and P. K. Mishra, "Cloud, Fog, and Mist Computing in IoT: An Indication of Emerging Opportunities," *IETE Technical Review*, vol. 39, no. 3, pp.713-724, 2021.

[10] N. Mohan and J. Kangasharju, "Placing it Right!: Optimizing Energy, Processing, and Transport in Edge-Fog Clouds," *Annals of Telecommunications*, vol. 73, pp. 463–474, 2018.

[11] N. I. Osman, "Will video caching remain energy efficient in future core optical networks?," *Digital Communications and Networks*, vol. 3, no. 1, pp. 39-46, 2017.

[12] F. Lin, Y. Zhou, G. Pau and M. Collotta, "Optimization-Oriented Resource Allocation Management for Vehicular Fog Computing," in *IEEE Access*, vol. 6, pp. 69294-69303, 2018.

[13] M. Collotta, G. Pau, V. M. Salerno and G. Scatà, "A distributed load balancing approach for industrial IEEE 802.11 wireless networks," *Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012)*, Krakow, Poland, pp. 1-7, 2012.

[14] Z. Chang, Z. Zhou, T. Ristaniemi and Z. Niu, "Energy Efficient Optimization for Computation Offloading in Fog Computing System," *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Singapore, pp. 1-6, 2017.

[15] R. Deng, R. Lu, C. Lai, T. H. Luan and H. Liang, "Optimal Workload Allocation in Fog-Cloud Computing Toward Balanced Delay and Power Consumption," in *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1171-1181, Dec. 2016.

[16] A. Mebrek, L. Merghem-Boulahia and M. Esseghir, "Efficient green solution for a balanced energy consumption and delay in the IoT-Fog-Cloud computing," *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*, Cambridge, MA, USA, pp. 1-4, 2017.

[17] G. Li, J. Wand, J. Wu and J. Song, "Data Processing Delay Optimization in Mobile Edge Computing," *Wireless Communication and Mobile Computing*, vol. 2018, 2018.

[18] F. Jalali, "Energy Consumption of Cloud Computing and Fog Computing Applications," Department of Electrical and Electronic Engineering, University of Melbourne, Australia, 2015.

[19] H. Neukirchen. (2021, Sep. 3). *Power Consumption of Raspberry Pi 4 Versus Intel J4105 System*[online]. Available: https://uni.hi.is/helmut/2021/06/07/power-consumption-of-raspberry-pi-4-versus-intel-j4105-system/.

[20] N. I. Osman, T. El-Gorashi and J. M. H. Elmirghani, "The impact of content popularity distribution on energy efficient caching," *2013 15th International Conference on Transparent Optical Networks (ICTON)*, Cartagena, Spain, pp. 1-6, 2013.

Niemah Izzeldin Osman received the B.Sc. degree (first class honours) in Computer Science from Sudan University of Science and Technology, Khartoum, Sudan, in 2002, the M.Sc. degree (with distinction) in Mobile Computing from the University of Bradford, U.K., in 2006 and the Ph.D. degree in Communication Networks from the University of Leeds, U.K in 2015. She is currently an Associate Professor at the Department of Computer Systems and Networks, Sudan University of Science and Technology, Sudan. Her current research interests include performance evaluation of low power networks and Mist Computing.

Mahmoud Al-Siddig Eisa received the B.Sc. degree (first class honours) in Computer Systems and Networks from Sudan University of Science and Technology, Khartoum, Sudan, in 2022. He is currently a software tester and quality assurance engineer at uTest. His research interests are Cloud Computing and software quality assurance.

Sabah Mohammed Imam received the B.Sc. degree (first class honours) in Computer Systems and Networks from Sudan University of Science and Technology, Khartoum, Sudan, in 2022 Her current research interests include Cloud Computing, network security and firewalls.