



Multi-Layer Fog-Cloud Architecture for Optimizing the Placement of IoT Applications in Smart Cities

Mohammad Aldossary*

Department of Computer Science, College of Arts and Science, Prince Sattam bin Abdulaziz University,
Al-Kharj, Saudi Arabia

*Corresponding Author: Mohammad Aldossary. Email: mm.aldossary@psau.edu.sa

Received: 20 August 2022; Accepted: 23 November 2022

Abstract: In the smart city paradigm, the deployment of Internet of Things (IoT) services and solutions requires extensive communication and computing resources to place and process IoT applications in real time, which consumes a lot of energy and increases operational costs. Usually, IoT applications are placed in the cloud to provide high-quality services and scalable resources. However, the existing cloud-based approach should consider the above constraints to efficiently place and process IoT applications. In this paper, an efficient optimization approach for placing IoT applications in a multi-layer fog-cloud environment is proposed using a mathematical model (Mixed-Integer Linear Programming (MILP)). This approach takes into account IoT application requirements, available resource capacities, and geographical locations of servers, which would help optimize IoT application placement decisions, considering multiple objectives such as data transmission, power consumption, and cost. Simulation experiments were conducted with various IoT applications (e.g., augmented reality, infotainment, healthcare, and compute-intensive) to simulate realistic scenarios. The results showed that the proposed approach outperformed the existing cloud-based approach in terms of reducing data transmission by 64% and the associated processing and networking power consumption costs by up to 78%. Finally, a heuristic approach was developed to validate and imitate the presented approach. It showed comparable outcomes to the proposed model, with the gap between them reach to a maximum of 5.4% of the total power consumption.

Keywords: IoT application placement; fog-cloud computing; power consumption; data transmission; optimization; smart cities

1 Introduction

The smart city paradigm can be described as a networked environment that connects different devices and services for efficient solutions. The Internet of Things (IoT) technology is emerging as an essential enabler for the promising future of smart cities that provide advanced computing services to their citizens. Consequently, IoT applications in smart cities generate a huge amount of distributed data (Big Data) that is processed in the cloud. Therefore, cloud computing can be considered as the



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

most suitable platform that provides solutions for the existing smart city applications (e.g., smart healthcare, smart transport and street lighting, smart environmental monitoring, and smart waste management). Usually, in the smart city, various IoT devices such as driverless cars, mobiles, and surveillance devices are connected to the cloud to use their computing services, which puts a huge load on the cloud networks to meet all the requirements of the IoT applications [1]. This could lead to a poor user experience due to the delays in the execution of IoT applications. As a result, cloud performance in the context of smart cities may suffer from different issues, such as service latency, load balancing, resource optimization, energy consumption, and cost optimization [1,2].

In this regard, Cisco has introduced fog computing to process IoT-generated data at the edge of the network (bringing compute resources nearer to end users), which plays an important role in limiting the above issues. The geo-distributed fog nodes would help process IoT tasks close to users and support real time smart city applications to improve overall system performance and reduce service latency, energy consumption, and cost caused by cloud processing [3]. However, resource allocation for IoT applications in fog-cloud environments is challenging because fog servers are usually resource-constrained (they have limited computational resources and storage to process the massive IoT data) and can only run a limited number of services [4]. Thus, there is a need for an efficient optimization approach to place and process IoT applications either in the fog, in the cloud, or both, depending on their requirements.

Some of the related works in the field of smart cities have focused on the execution of IoT applications and services, as presented in [1,2,4-10]. For example, Belli et al. [5] have highlighted the key factors of an IoT infrastructure for smart cities along with their challenges. They have discussed the deployment of IoT-oriented solutions and highlighted the importance of a smart city ecosystem for the services offered, considering the energy sources, efficient distribution, and power consumption of IoT technologies. Similarly, Perera et al. [6] and Badidi et al. [7] have discussed the importance of employing distributed fog servers to support sustainable IoT infrastructure for smart cities from both communication and computing perspectives. In addition, Goudarzi et al. [2] presented a taxonomy for scheduling IoT applications in the fog computing paradigm, considering multiple perspectives (e.g., system design, optimization characteristics, application structure, decision engines, and system performance evaluation).

Muneeb et al. [4] proposed a multi-layer architecture for analyzing IoT data in a cloud-fog computing, considering service latency, network usage, and energy consumption. Additionally, Cabrera et al. [8] presented a dynamic model for application placement in smart cities based on certain Quality of Service (QoS) requirements, such as load balancing and service latency. However, their models have not addressed the execution of IoT services for smart city applications. Qayyum et al. [9] proposed a simulation-based framework for IoT devices to achieve maximum resource usage and throughput in a collaborative fog environment. However, their work was relied on the capacity of IoT devices, and any limited access to these devices would impact the performance of their proposed framework. Moreover, Clarindo et al. [10] introduced a three-layer architecture in the context of smart cities that helps a smart city manager to analyze IoT data and perform a decision-making process. However, all the listed works have not considered the cost of resource usage, power consumption, and service latency in multi-layer environments. Some of these techniques have considered service latency by focusing mainly on application placement, while other metrics such as cost and power consumption, need further investigation [11].

The main motivation for this work was to employ a multi-layer fog-cloud architecture to efficiently place and process IoT applications in real time. This approach takes into account the requirements

of IoT applications, available resource capacities, and geographical locations of both fog and cloud servers, which would play an important role in improving IoT application placement decisions. In this work, extensive experiments were conducted with various IoT applications (e.g., augmented reality, infotainment, healthcare, and compute-intensive) to optimize data transmission, power consumption, and cost over a fog-cloud architecture. Additionally, comparative experiments were conducted to evaluate the feasibility of the proposed approach and compare the outcomes with the traditional cloud-based environment. The contributions of this work are summarized as follows:

- Develop a mathematical model to analyze the placement of IoT applications in a jointly fog-cloud system, considering application requirements, available resources, and server locations with respect to users.
- Optimize the placement of IoT applications over a fog-cloud architecture to minimize data transmission and power consumption for processing and networking along with their associated costs.
- Evaluate the efficiency and feasibility of the proposed approach and its models, and compare the outcomes with the traditional method in the context of smart cities using various IoT applications.

The remainder of this paper is structured as follows: Section 2 describes the proposed multi-layer system architecture and its implementation details. The model of the proposed system is described in Section 3. The experimental setup and model parameters are provided in Section 4. The performance evaluation and analysis of the results are included in Section 5. In Section 6, a heuristic approach has been developed to validate the introduced approach in terms of power consumption, data transmission, and associated costs. Finally, conclusions and future work are presented in Section 7.

2 The Proposed System

The concept of smart cities has emerged with many urban services and initiatives (e.g., smart environment, smart economy, smart living, smart governance, etc.) that efficiently manage and optimize resources to achieve sustainability [12]. Therefore, providing innovative IoT services and solutions to enhance people's quality of life is one way to achieve this aim [7]. However, the deployment of IoT services in the context of smart cities would require considerable communication and computing resources, which would consume a lot of energy and increase operating costs [11,13].

In this section, an efficient optimization approach is presented for placing and processing IoT applications in a multi-layer fog-cloud architecture, considering data transmission, power consumption, and associated costs. The approach shown in Fig. 1 achieves these objectives by using four different layers: the IoT layer, the orchestrator layer, the fog layer, and the cloud layer, which collaborate with each other. The following are the descriptions of each layer.

2.1 IoT Layer

The IoT layer comprises a network of heterogeneous IoT devices that are geographically distributed and interconnected via wireless technologies (e.g., Zigbee, LoRa, Wi-Fi, NB-IoT, and 4G/5G) under the network communication standard-the Local Area Network (LAN) [13]. These technologies are available in many places in the smart city (e.g., smart homes, public transportation, hospitals, and traffic lights). Furthermore, IoT devices are usually resource-constrained, with a limited battery life. Hence, this layer would aim to collect data from IoT devices and send/offload them either in the fog or in the cloud for processing based on the orchestrator's decisions.

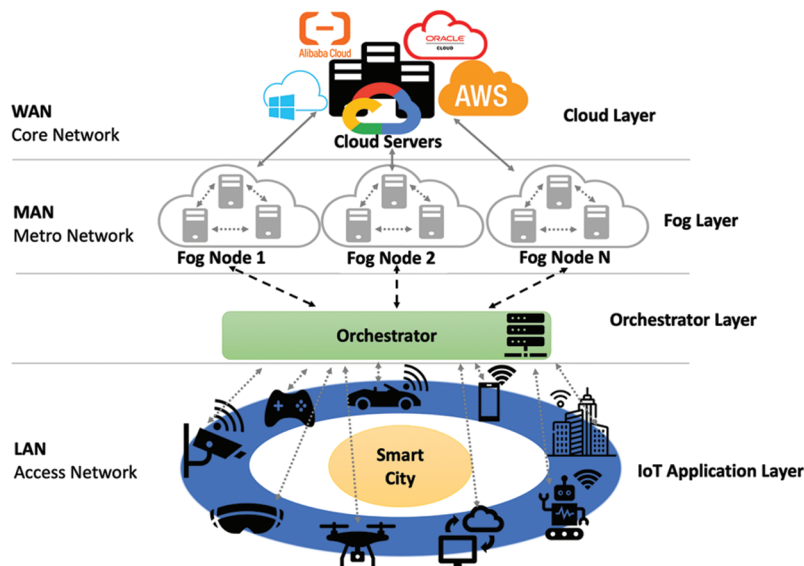


Figure 1: An overview of the multi-layer system architecture

2.2 Orchestrator Layer

Each IoT application (which consists of multiple tasks) would be accessed by the system orchestrator. The orchestrator would then decide which tasks to run/execute locally on the fog nodes and which tasks to place/offload on the cloud server [3]. This would be based on the predefined requirements of IoT applications, such as how much CPU capability or speed (the number of Million Instructions Per Second (MIPS)) is required to execute the application, as well as the data transmission, power consumption, and cost of performing the tasks either in the fog, in the cloud, or both. Also, this layer would help optimize the placement of IoT applications in the proposed system and process them efficiently.

2.3 Fog Layer

Data generated by IoT devices is sent/offloaded to geo-distributed fog nodes, which are located close to the network edge (nearer to the end-user) using the network communication standard-the Metro Area Network (MAN). The fog layer plays an essential role in smart cities' services that require immediate responses (e.g., e-healthcare systems and driverless cars). Thus, finding the optimal location of fog nodes would allow the execution of IoT applications and provide data analytics in real time due to the reduced communication latency between IoT devices and fog nodes [1]. However, fog nodes are limited compared to the cloud in terms of data storage and processing, which should be taken into consideration when performing the placement decision.

2.4 Cloud Layer

Due to the scalability and high availability of cloud computing resources, this layer can work in parallel with the fog layer to handle the requirements of IoT applications that cannot be executed in fog nodes. This allows IoT applications that would not require low latency to be processed in the cloud using the network communication standard-the Wide Area Network (WAN) in order to improve their performance considerably. Also, in this approach, the cloud layer is mainly responsible

for IoT applications that are compute-intensive and complex tasks, which require sufficient resources to perform them [11].

3 The Proposed Model

The Mixed Integer Linear Programming (MILP) model is extensively used for optimization problems, where the objective is a linear function of variables. It can also be used to approximate nonlinear functions by linear functions and constraints [2], which are used in this work. When using the MILP model, it's important to understand that it does not necessarily find the optimal solution. Instead, it finds a solution that is near to an optimal solution.

In this section, the problem was formulated, and the MILP model was developed to optimize several objectives, such as power consumption, data transmission, and their associated costs over a fog-cloud environment.

3.1 Power Consumption Optimization

As shown in Table 1, parameters and variables were introduced to describe the power consumption for processing IoT applications in a fog-cloud computing, along with their models.

Table 1: Parameters and variables of power consumption for processing in a fog-cloud environment

| Environment | Notations | Description | |
|---------------|------------|------------------|------------------------------------------------------------------------------------------------------------|
| Cloud and Fog | Parameters | $IPC^{(cloud)}$ | Internal power consumption of cloud networks. |
| | | $IPF^{(f)}$ | Internal power consumption of fog networks. |
| | | $PC^{(power)}$ | Power consumed by cloud server. |
| | | $PF^{(power)}$ | Power consumed by fog server. |
| | | $Cloud^{(MIPS)}$ | Maximum MIPS of cloud. |
| | | $Fog^{(MIPS)}$ | Maximum MIPS of fog. |
| | | $PM^{(cloud)}$ | Power/MIPS ratio in cloud. |
| | | $PM^{(fog)}$ | Power/MIPS ratio in fog. |
| | | $PUE^{(cloud)}$ | Cloud-power usage effectiveness (PUE). |
| | | $PUE^{(fog)}$ | Fog-power usage effectiveness (PUE). |
| | | N | Nodes created in fog-cloud architecture. |
| | | C | Nodes created in cloud layer. |
| | | F | Nodes created in fog layer. |
| | | I | IoT devices set. |
| | | s and d | Fog-cloud nodes–source (s) and destination (d) indices. |
| | | P | Program set. |
| | | S_i | Devices of program i. |
| | | UD_i | Upload/download data rates of program i. |
| | | TL_i | Task length of program i. |
| | Variables | SA_d | $\begin{cases} = 1 & \text{when server } d \in N \text{ is activated} \\ = 0, & \text{if not} \end{cases}$ |

(Continued)

Table 1: Continued

| Environment | Notations | Description |
|-------------|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| | $SR_{i,d}$ | $\begin{cases} = 1 \text{ when program } i \in \mathbb{I} \text{ is running on server } d \in \mathbb{N} \\ = 0, \text{ if not} \end{cases}$ |
| | $MIPS_{i,s}^{iot}$ | MIPS required by program i placed in fog or cloud s . |
| | SM_s^{iot} | Sum of MIPS required by program i placed in fog or cloud s . |
| | UD_i | Upload/Download data rates of program i . |
| | $TF_{i,s,d}^{iot}$ | Traffic flow from IoT devices located in d for a program i placed in fog or cloud s . |
| | $TF_{s,d}$ | Traffic flow from IoT devices located in d to fog or cloud s . |

The cloud and fog layers' power consumption includes:

- (1) Power consumption of cloud layer (Cloud):

$$PUE^{(cloud)} \left(\sum_{s \in \mathbb{N}} MIPS_{i,s}^{iot} PM^{(cloud)} + \sum_{s \in \mathbb{N}} IPC^{(cloud)} TF_{s,d} \right) \forall s \in \mathbb{C} \quad (1)$$

- (2) Power consumption of fog layer (Fog):

$$PUE^{(fog)} \left(\sum_{s \in \mathbb{N}} MIPS_{i,s}^{iot} PM^{(fog)} + \sum_{s \in \mathbb{N}} IPF^{(fog)} TF_{s,d} \right) \forall s \in \mathbb{F} \quad (2)$$

Eqs. (1) and (2) calculate the total power consumption of cloud and fog layers.

3.2 Data Transmission Optimization

The parameters and variables were introduced along with their models to describe the power consumption of the network for processing IoT applications in fog-cloud computing, as presented in Table 2.

Table 2: Parameters and variables of power consumption for networking in a fog-cloud environment

| Networks | Notations | Description | |
|----------------------|-------------|---------------------------|------------------------------------------------------------------------------------------------------------|
| Access network (LAN) | Parameters | $S_{i,s,d}$ | Number of IoT devices in node d with uploading traffic of IoT application i hosted in node s . |
| | | $TF_{i,s,d}$ | Traffic flow from IoT program i located in d to fog or cloud s as: $TF_{i,s,d} = UD_{i,s,d} UD_i$ |
| | $OLT^{(P)}$ | OLT power consumption. | |
| | $ONU^{(P)}$ | An ONU power consumption. | |
| | $OLT^{(B)}$ | OLT bit rate. | |

(Continued)

Table 2: Continued

| Networks | Notations | Description | |
|---------------------|----------------------------------------------------------------------------------------|--------------------|--------------------------------------------------------------------|
| Metro network (MAN) | Variables | $ONU^{(B)}$ | ONU bit rate. |
| | | $PUE^{(N)}$ | Network-power usage effectiveness (PUE). |
| | | LN | Large number. |
| | Parameters | $ONU_s^{(N)}$ | ONU terminals number. |
| | | $OLT_s^{(N)}$ | OLT overall power usage. |
| | | $MR^{(PWR)}$ | Metro routers' maximum power usage. |
| | | $MS^{(PWR)}$ | Metro switches' maximum power usage. |
| | | $MR^{(BW)}$ | Metro routers' bandwidth. |
| | | $MS^{(BW)}$ | Metro switches' bandwidth. |
| | | $MR_s^{(C)}$ | Metro routers' count in node s . |
| Core network (WAN) | Parameters | $MS_s^{(C)}$ | Metro switches' count in node s . |
| | | $m \text{ and } n$ | Indices of the network architecture. |
| | Variables | N_m | A collection of nodes m (neighbors) in a fog-cloud architecture. |
| | | $RP^{(PWR)}$ | Router port-power usage. |
| | | $T^{(PWR)}$ | Transponder-power usage. |
| | | $A^{(PWR)}$ | Amplifier-power usage. |
| | | $OS^{(PWR)}$ | An optical switch-power usage. |
| | | B | Link bandwidth |
| | | g | Maximum gap among amplifiers EDFAs (measured in kms). |
| | | $A_{m,n}$ | Amplifiers number among node pairs $(m, n) \in C$. |
| R_d | Router ports in $d \in C$. | | |
| T_d | Total traffic in $d \in N$. | | |
| $F_{m,n}$ | Fibers count per link $(m, n) \in C$. | | |
| $T_{m,n}^{s,d}$ | Traffic among node pairs $(s, d) \in C$ to transmit the physical link $(m, n) \in C$. | | |

Power consumption of the local area network (LAN) includes:

$$PUE^{(N)} \left(\sum_{s \in N} ONU_s^{(N)} ONU^{(P)} \right) + \left(\sum_{s \in N} OLT_s^{(N)} OLT^{(P)} \right) \tag{3}$$

The total power consumption of the local network is estimated in Eq. (3), considering Optical Network Units (ONU), Optical Line Terminals (OLTs) devices, and Power Usage Effectiveness (PUE) of the network.

Power consumption of the metro network (MAN) includes:

$$PUE^{(N)} \left((MR_s^{(C)} MR_s^{(PWR)}) + (MS_s^{(C)} MS_s^{(PWR)}) \right) \forall s \in N \tag{4}$$

The metro network's total power consumption is determined by Eq. (4).

Power consumption of the wide area network (WAN) includes:

$$\begin{aligned} \text{PUE}^{(N)} \left(\sum_{d \in N} \text{RP}^{(PWR)} R_d + \sum_{m \in N} \sum_{n \in N, m: n \neq m} \sum_{s \in N} \sum_{d \in N: s \neq d} T_{m,n}^{s,d} T^{(PWR)} \right. \\ \left. + \sum_{m \in N} \sum_{n \in N, m: n \neq m} A^{(PWR)} F_{m,n} A_{m,n} + \sum_{d \in N} \text{OS}_d^{(PWR)} \right) \end{aligned} \quad (5)$$

The core network's total power consumption is determined by Eq. (5).

By taking into account Eqs. (1)–(5), the MILP model is expressed as follows:

The objective: minimizing the Total Power Consumption (TPC):

$$\text{LAN} + \text{IoT} + \text{MAN} + \text{Fog} + \text{WAN} + \text{Cloud} \quad (6)$$

Eq. (6) is used to calculate the total power consumption of the proposed approach.

Subject to the following **constraints**:

IoT application:

$$\sum_{s,d \in N} \text{UD}_{i,s,d} = \sum_{s,d \in N} \text{TF}_{i,s,d}^{\text{iot}} \quad \forall i \in I \quad (7)$$

Constraint (7) ensures that all IoT application traffic is controlled in the cloud or fog layers.

Availability of fog and cloud nodes in a smart city:

$$\left. \begin{aligned} \sum_{s \in N} \text{TF}_{i,s,d}^{\text{iot}} &\geq \text{SR}_{i,d} \\ \sum_{s \in N} \text{TF}_{i,s,d}^{\text{iot}} &\leq \text{LN SR}_{i,d} \end{aligned} \right\} \quad \forall d \in N, i \in I \quad (8)$$

Constraint (8) ensures that $\text{SR}_{i,d} = 1$, if the processing node $d \in N$ is turned on to place the IoT application $i \in I$; otherwise, $\text{SR}_{i,d} = 0$.

Physical link:

$$\left. \begin{aligned} T_{m,n}^{s,d} &\geq \text{TF}_{m,n}^{s,d} \\ T_{m,n}^{s,d} &\leq \text{TF}_{m,n}^{s,d} \end{aligned} \right\} \quad \forall s, d, m, n \in C \quad (9)$$

Constraint (9) ensures that physical link $(m, n) \in C$ is on when there is a flow among the nodes $(s, d) \in C$ transmitted over links $(m, n) \in C$.

Cloud and fog processing:

$$\text{MIPS}_{i,d}^{\text{iot}} = \text{SR}_{i,d} \text{MIPS}_{i,d}^{\text{iot}} \quad \forall d \in N, i \in I \quad (10)$$

$$\text{MIPS}_d^{\text{iot}} = \sum_{i \in I} \text{MIPS}_{i,d}^{\text{iot}} \quad \forall d \in N \quad (11)$$

Constraint (10) calculates the processing conditions of IoT application $i \in I$ of both cloud and fog layers, while constraint (11) calculates the entire processing of both cloud and fog layers $d \in N$.

Traffic flow (data transmission) in the core network:

$$\text{TF}_{s,d} = \sum_{i \in I} \text{TF}_{i,s,d}^{\text{iot}} \quad \forall s, d \in C \quad (12)$$

Constraint (12) computes data transmission among core nodes due to the placement of IoT applications in the cloud.

Total number of router ports in the core network:

$$RP_d \geq \frac{\sum_{s \in C} TF_{s,d}}{B} \quad \forall d \in C \quad (13)$$

Constraint (13) computes the router ports count at every core node.

Total number of metro routers:

$$MR_s^{(C)} \geq 2 \frac{\sum_{i \in I} \sum_{d \in (f \cap C)} TF_{i,s,d}}{MR^{(B)}} \quad \forall s \in N \quad (14)$$

The number of active routers in each metro network s is provided by constraint (14).

Total number of metro switches:

$$MS_s^{(C)} \geq \frac{\sum_{i \in I} \sum_{d \in (f \cap C)} TF_{i,s,d}}{MS^{(B)}} \quad \forall s \in N \quad (15)$$

Constraint (15) computes the number of switches in the metro network s .

Communication network-total data transmission:

$$TF_d = \sum_{i \in I} \sum_{d \in N} TF_{i,s,d} \quad \forall s \in N \quad (16)$$

The total data transmission in each node s is provided by constraint (16).

3.3 Associated Costs Optimization

In this subsection, the energy cost (\$ per kWh) is examined under the considered scenarios. The parameter related to the energy cost is given in Table 3.

Table 3: Parameter of energy cost for processing and networking in a fog-cloud environment

| Parameter | Cost |
|-----------------------------------|----------------------|
| Cost per energy produced (C). | \$0.18 per kWh [11]. |

By taking into account Eq. (6), the MILP model can be expressed as follows:

The objective: minimizing the total energy cost:

$$C * TPC \quad (17)$$

4 Experimental Design and Model Parameters

In this section, the experiment design and model parameters were presented in order to evaluate the proposed model in a geo-distributed fog-cloud environment.

4.1 Design of the Experiment

The proposed approach was assessed in a simulated environment using the MILP model. This simulation runs on a Lenovo laptop running Windows 10, an Intel Core i7-11800H, a 2.30 GHz processor, 1 TB of storage, and 32 GB of RAM. The network consists of a cloud server, a set of 50 fog nodes, and 10 million IoT devices distributed across United States (US) cities, as shown in Fig. 2. The distribution of fog nodes is also based on the population size of each state.

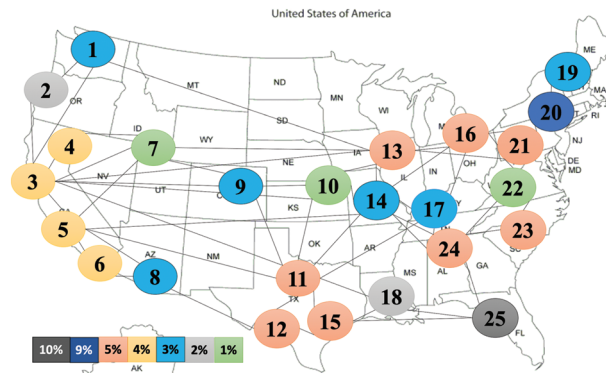


Figure 2: AT&T network topology (US) with the population percentage in each state

In the context of smart cities, the number of resources requested by each IoT application varies, which would help in investigating and evaluating the placement approaches (e.g., placing IoT applications either in the fog, in the cloud, or both) in order to find the optimal location. In this approach, the cloud server manages the fog nodes and can orchestrate the entire system's resources. Let us assume that the number of fog nodes should be increased first to meet the requirements of IoT applications. Thus, IoT applications could be placed locally on the fog nodes, provided that the fog nodes can meet these requirements. If the requirements of the IoT applications exceed the capacity of the fog nodes, the IoT applications could be efficiently placed on a cloud server.

In each experiment, 10 million IoT devices were simulated: 30% were augmented reality (AR) users, 30% were infotainment users, 20% were healthcare users, and 20% were compute-intensive users as shown in Table 4. These percentages were determined based on [8,14] to simulate realistic environments, where augmented reality and infotainment applications had the largest share (i.e., IKEA Mobile App, Disney Coloring Book, Weather Channel Studio Effects) and (i.e., Netflix, Snapchat, Instagram), respectively; followed by healthcare applications (i.e., Remote Patient Monitoring, Heart-Rate Monitoring, Glucose Monitoring), and compute-intensive applications (i.e., Video Editing and Processing, Software Compiling, Machine Learning).

4.2 Input Parameters of the Models

Table 4: Input parameters and values of the model (IoT applications, cloud and fog, and networks) [8,11,13,14]

| Parameters | Notations | Values |
|-------------------------|------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IoT Applications | Number of devices ($\text{IoT}_s^{\text{(number)}}$)-upload/download data rates (UD_i). | 10,000,000 IoT devices, 30% augmented reality users-with upload/download data rates 1500/25 KB, 30% infotainment users-with upload/download data rates 25/1000 KB, 20% healthcare users-with upload/download data rates 20/1250 KB, 20% compute-intensive users-with upload/download data rates 2500/200 KB. |

(Continued)

Table 4: Continued

| Parameters | Notations | Values |
|---------------------------------------------------------------|------------------------------------------------------------------------|-----------------------------------------|
| Cloud and Fog | Cloud processing node's maximum capacity (Cloud ^(MIPS)). | 100640 MIPS |
| | Fog processing node's maximum capacity (Fog ^(MIPS)). | 27000 MIPS |
| | Internal power consumption of cloud networks (PM ^(cloud)). | 2.48 Watts/Gbps |
| | Internal power consumption of fog networks (PM ^(fog)). | 2.70 Watts/Gbps |
| | Cloud power usage effectiveness (PUE) (PUE ^(cloud)). | Traditional: 1.3 and best practice: 1.1 |
| | Fog power usage effectiveness (PUE) (PUE ^(fog)). | Traditional: 2.5 and best practice: 1.5 |
| Networks | Router port power usage (RP ^(PWR)). | 638 Watts |
| | Transponder power usage (T ^(PWR)). | 129 Watts |
| | Switch power usage (SW _d ^(power)). | 85 Watts |
| | EDFA power usage (A ^(power)). | 11 Watts |
| | Gap among EDFAs (<i>S</i>). | 80 KM |
| | Network-power usage effectiveness (PUE ^(network)). | 1.5 |
| | Metro router port power usage (MR ^(PWR)). | 30 Watts |
| | Metro switch power usage (MS ^(PWR)). | 470 Watts |
| | Metro router port bitrate (MR ^(B)). | 40 Gbps |
| | Metro switch bitrate (MS ^(B)). | 600 Gbps |
| Access network OLT power consumption (OLT ^(PWR)). | 1842 Watts | |

5 Results and Discussion

An extensive number of experiments were carried out in this section to show the performance of the proposed approach. First, the power consumption of both processing and networking was analyzed and examined under two conditions of Power Usage Effectiveness (PUE): 1) the traditional PUE, where the Cloud PUE is set to (1.3) and the Fog PUE is set to (2.5); and 2) the best practice PUE, where the Cloud PUE is set to (1.1) and the Fog PUE is set to (1.5). Second, the associated costs are highlighted to show how much savings are achieved with the proposed approach. Finally, the total data transmissions in different network layers were investigated to optimally place and process IoT applications over a fog-cloud architecture.

5.1 Power Consumption

In the following, the power consumption was optimized considering two approaches as aforementioned: traditional PUE and best practice PUE. In the traditional PUE, the energy efficiency of using cloud and fog resources is low due to the high impact of cooling and other resources on the total power consumption, i.e., Cloud PUE (1.3) and Fog PUE (2.5), while the best practice PUE minimizes this impact, i.e., Cloud PUE (1.1) and Fog PUE (1.5).

5.1.1 Power Usage Effectiveness (PUE)–Traditional vs. Best Practice

Fig. 3 shows the comparison of the total power consumption of two different scenarios: the fog-cloud approach vs. the cloud-only approach under the traditional PUE. This comparison summarizes that each IoT application using the fog-cloud approach always results in power savings compared to the cloud-only approach; except for the compute-intensive applications, as they require high processing capacity (i.e., MIPS), which is only available at the cloud layer. Likewise, Fig. 4 illustrates the comparison of the total power consumption of two different scenarios: the fog-cloud approach vs. the cloud-only approach, but under the best practice PUE. In addition to the previous discussion (about Fig. 3), this comparison also shows that placing IoT applications in the fog layer always results in power savings, as power consumption is partially eliminated by using the core network. These trends were observed in both traditional PUE and best practice PUE approaches.

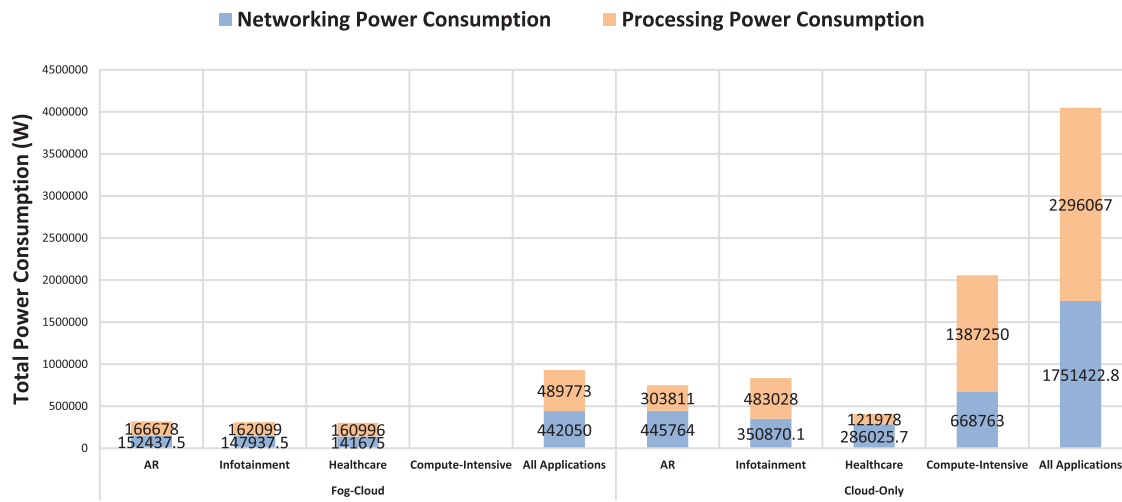


Figure 3: Power consumption for fog-cloud approach vs. cloud-only approach–the traditional PUE

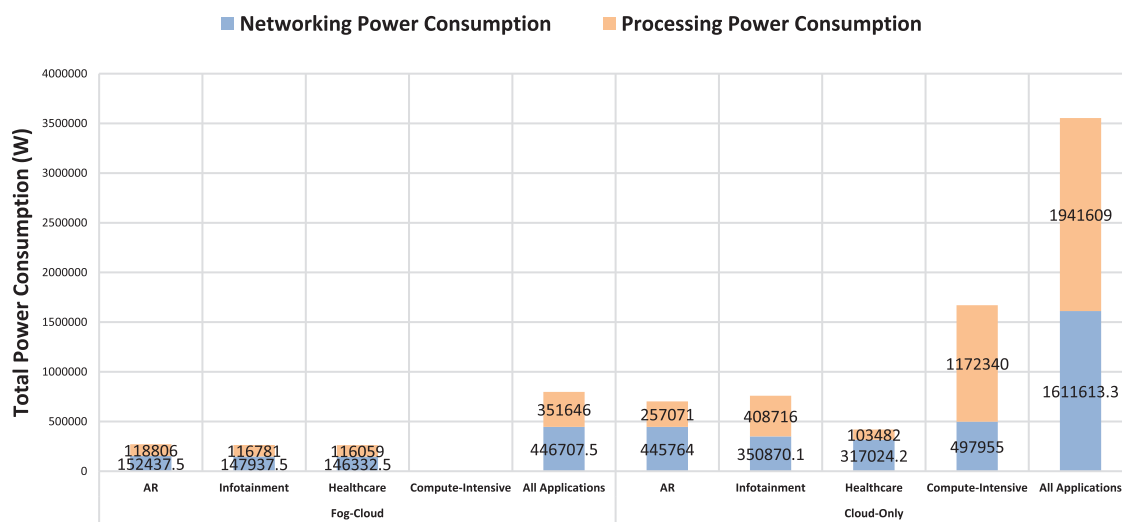


Figure 4: Power consumption for fog-cloud approach vs. cloud-only approach–the best practice PUE

Thus, the orchestrator made appropriate decisions about placement approaches, considering both processing and networking power consumption. Consequently, it decided to execute all IoT applications locally on the fog nodes to optimize overall power consumption and to place and process only the compute-intensive applications on the cloud server to meet the applications' requirements.

5.1.2 Associated Costs—Traditional vs. Best Practice

Fig. 5 shows the total cost savings for hosting IoT applications in the fog-cloud approach compared to the cloud-only approach under the traditional PUE. The results show that placing and processing IoT applications in the fog-cloud approach (using the fog layer) has reduced the total cost (in terms of energy cost for placing and processing IoT applications) by 57%, 63%, and 26%, respectively, compared to the cloud-only approach for AR, infotainment, and healthcare applications. For the compute-intensive applications, no savings are shown when a fog-cloud approach is performed, as these applications are only placed on the cloud server for processing in a cloud-only approach. However, when all applications are combined in one scenario, the total cost saving of using the fog-cloud approach hits 77%.

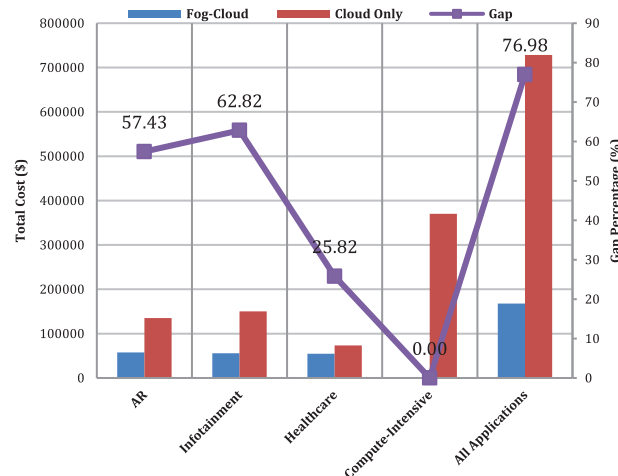


Figure 5: Total cost saving for hosting IoT applications in fog-cloud vs. cloud-only—the traditional PUE

Also, Fig. 6 depicts the total cost savings for hosting IoT applications in the fog-cloud approach compared to the cloud-only approach under the best practice PUE. The comparison results show that placing and processing IoT applications in the fog-cloud approach (using the fog layer) results in total cost savings of 61%, 65%, and 38% compared to the cloud-only approach for AR, infotainment, and healthcare applications, respectively. Similar to Fig. 5, the compute-intensive applications were placed and processed only on the cloud server; therefore, no savings were achieved. In addition, when all applications are combined into one scenario, the overall cost saving of using the fog-cloud approach reaches 78%.

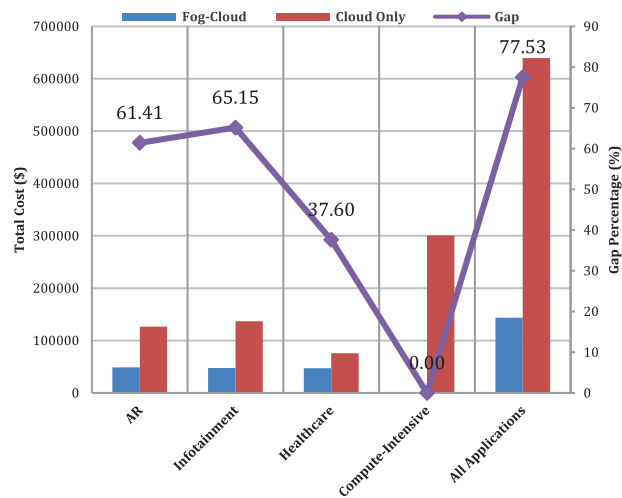


Figure 6: Total cost saving for hosting IoT applications in fog-cloud vs. cloud-only—the best practice PUE

5.2 Data Transmission

Fig. 7 shows a comparison between the total data transmission in the access network vs. the core network. This indicates that the fog-cloud approach was able to save 64% (equivalent to 385 Gigabits per second (Gbps)) on data transmission in the core network compared to the cloud-only approach. This saving was achieved by transmitting most IoT applications to the fog layer via the access network for processing, rather than traversing the core network to execute IoT applications in the cloud layer.

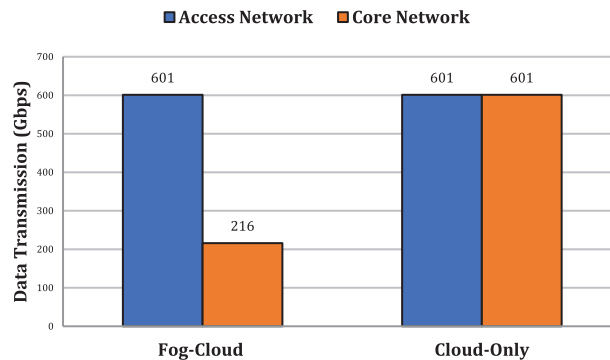


Figure 7: Comparison between total data transmission in the access network vs. the core network

6 The Heuristic Approach

In this section, a heuristic approach to validate the IoT application placement results delivered by the proposed model (MILP model) was introduced. Yet, optimizing the placement of IoT applications using the MILP model to reduce power consumption, data transmission, and cost is considered as a Non-deterministic Polynomial (NP)-hard problem [11]. The MILP model could not find a solution in polynomial time. Since the placement of IoT applications over a fog-cloud architecture with

$N = 75$ placement locations (25 cloud locations and 50 fog locations) requires an examination of many solutions set as $\left(\sum_{i=1}^N \frac{N!}{(N-i)!}\right)$.

To solve this problem, a sorted list of nodes (in different cities) from the lowest to the highest weight was created, and this list was also used to decide the best location for placing IoT applications. The sorted list helps to reduce the search time for evaluating $|N|$ combinations, which also reduces the time complexity by a factor of 1.6×10^{10} . Let us consider (N) as the number of IoT applications and (i) as the number of various fog nodes and cloud server locations. Thus, the combination of these parameters will result in finding the optimal or near-optimal location for placing IoT applications in a set of possible fog nodes or a cloud server to minimize power consumption in a fog-cloud architecture. Algorithm 1 illustrates the heuristic approach to efficiently place and process IoT applications in a fog-cloud architecture.

Algorithm 1: Finding the optimal placement of IoT applications

$(F_{f,p,c})$: Fog f power consumption of running IoT application p , taking into consideration the placement c .

$(C_{c,p,c})$: Cloud c power consumption of running IoT application p , taking into consideration the placement c .

$(Cost_{p,cf,c})$: Total costs of running IoT application p in cloud/fog cf , taking into consideration placement c .

Input: The list of sorted nodes (E)

Output: Optimal power consumption (CFC)

Optimal placement of IoT application (J')

Total Costs (TC)

1: **for each** (Application Type $x \in p$) **do**

2: **for each** (Placement $i \subset c$) **do**

3: **for each** (Node $d \in s$) **do**

4: **for each** (Fog and Cloud candidate $cf \in s$) **do**

5: CFC = $(PUE^{(cloud)} E_s C_{c,p,c})(PUE^{(fog)} E_s F_{f,p,c})$

6: **end for**

7: **end for**

8: $TC_{p,c} = Cost_{p,cf,c} * (\sum_{s \in N} CFC) + MAN_{p,c} + LAN_{p,c}$

9: **end for**

10: $TC_p = \text{Min} \{TC_{p,c}\}$

11: $J' = J$

12: **end for**

13: Calculate $TC = \sum_{x \in p} TC_p$

For each type of IoT application (determined by its resource requirements), the algorithm computes the overall cost TC_a associated with each IoT application $a \in p$ and its placement $J \subset E$. First, the access and core networks (data transmission) were distributed to nodes s and d using a multi-hop heuristic approach described in [13], and then the power consumption of the access and core networks was estimated. Finally, the optimal location (J') for hosting IoT applications was chosen and the overall cost of the fog-cloud architecture (TC) was estimated.

This heuristic approach was evaluated using a Lenovo laptop running Windows 10, an Intel Core i7-11800H, a 2.30 GHz processor, 1 TB of storage, and 32 GB of RAM. Also, the heuristic approach

used the same AT&T network topology as the MILP model and showed a comparable result to the MILP model after 10 s of evaluation. According to Fig. 8, the gap between the heuristic approach and the MILP model was at most 5.4% of the total power consumption, and it achieved similar savings.

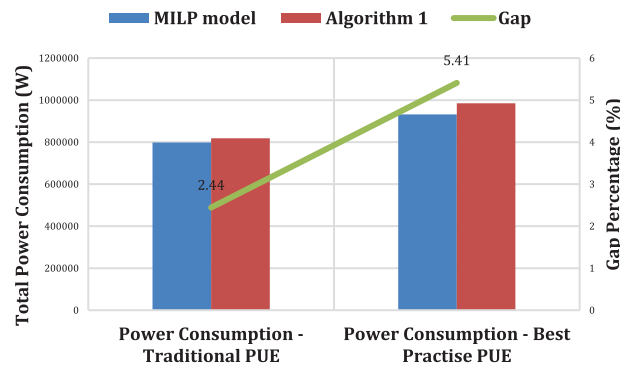


Figure 8: Gap between the MILP model and Algorithm 1

7 Conclusion and Future Work

In this paper, an efficient optimization approach using a multi-layer fog-cloud architecture was introduced to overcome the challenges of placing and processing IoT applications in real time. The proposed approach considered the requirements of IoT applications, the available resource capacities, and the geographical locations of fog and cloud servers to efficiently optimize the placement decisions for IoT applications. Furthermore, a simulation experiment was conducted with different IoT applications. The results showed that the proposed approach outperformed the existing cloud-based approach in terms of reducing data transmission by 64% and the associated processing and networking power consumption costs by up to 78%. Finally, a heuristic approach was developed to validate and imitate the demonstrated approach. It showed comparable outcomes to the proposed model, with a gap between them reached to a maximum of 5.4% of the total power consumption. In future work, an extension of the fog-cloud system architecture would be designed using intelligent decision making and machine learning techniques to handle unpredictable cases, considering mobility patterns (e.g., driverless cars and drones) in smart city environments.

Acknowledgement: The author would like to thank the Deanship of Scientific Research, Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia, for supporting this research.

Funding Statement: The author received no specific funding for this study.

Conflicts of Interest: The author declares that he has no conflicts of interest to report regarding the present study.

References

- [1] O. A. Mahmood, A. R. Abdellah, A. Muthanna and A. Koucheryavy, "Distributed edge computing for resource allocation in smart cities based on the IoT," *Information*, vol. 13, no. 7, pp. 328, 2022.
- [2] M. Goudarzi, M. Palaniswami and R. Buyya, "Scheduling IoT applications in edge and fog computing environments: A taxonomy and future directions," *ACM Computing Surveys*, vol. 1, no. 1, pp. 43, 2022.

- [3] A. Liutkevičius, N. Morkevičius, A. Venčkauskas and J. Toldinas, “Distributed agent-based orchestrator model for fog computing,” *Sensors*, vol. 22, no. 15, pp. 5894, 2022.
- [4] M. Muneeb, K. -M. Ko and Y. -H. Park, “A fog computing architecture with multi-layer for computing-intensive IoT applications,” *Applied Sciences*, vol. 11, no. 24, pp. 11585, 2021.
- [5] L. Belli, C. Antonio, D. Luca, F. Gianluigi, A. Paolo *et al.*, “IoT-enabled smart sustainable cities: Challenges and approaches,” *Smart Cities*, vol. 3, no. 3, pp. 1039–1071, 2020.
- [6] C. Perera, Y. Qin, J. C. Estrella, S. Reiff-Marganiec and A. V. Vasilakos, “Fog computing for sustainable smart cities,” *ACM Computing Surveys*, vol. 50, no. 3, pp. 1–43, 2018.
- [7] E. Badidi, Z. Mahrez and E. Sabir, “Fog computing for smart cities’ big data management and analytics: A review,” *Future Internet*, vol. 12, no. 11, pp. 190, 2020.
- [8] C. H. Cabrera Jojoa, S. Svorobej, A. Palade, A. Kazmi and S. Clarke, “MAACO: A dynamic service placement model for smart cities,” *IEEE Transactions on Services Computing*, pp. 1, 2022. <https://doi.org/10.1109/TSC.2022.3143029>.
- [9] T. Qayyum, Z. Trabelsi, A. W. Malik and K. Hayawi, “Multi-level resource sharing framework using collaborative fog environment for smart cities,” *IEEE Access*, vol. 9, pp. 21859–21869, 2021.
- [10] J. P. Clarindo, J. P. C. Castro and C. D. Aguiar, “Combining fog and cloud computing to support spatial analytics in smart cities,” *Journal of Information and Data Management*, vol. 12, no. 4, pp. 342–360, 2021.
- [11] M. Aldossary and H. A. Alharbi, “An eco-friendly approach for reducing carbon emissions in cloud data centers,” *Computers, Materials & Continua*, vol. 72, no. 2, pp. 3175–3193, 2022.
- [12] N. Mohamed, J. Al-Jaroodi, S. Lazarova-Molnar and I. Jawhar, “Applications of integrated IoT-fog-cloud systems to smart cities: A survey,” *Electronics*, vol. 10, no. 23, pp. 2918, 2021.
- [13] H. A. Alharbi and M. Aldossary, “Energy-efficient edge-fog-cloud architecture for IoT-based smart agriculture environment,” *IEEE Access*, vol. 9, pp. 110480–110492, 2021.
- [14] T. T. Khanh, V. Nguyen and E. -N. Huh, “Fuzzy-based mobile edge orchestrators in heterogeneous IoT environments: An online workload balancing approach,” *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–19, 2021.