Energy-Efficient Task Allocation in Heterogeneous Fog Computing Infrastructures

Ida Syafiza Md Isa, Nur Latif Azyze Mohd Shaari Azyze, Wan Haszerila Wan Hassan, and Darmawaty Mohd Ali

Original scientific article

***Abstract*—The rapid growth of Internet of Things (IoT) applications has intensified the demand for energy-efficient and low-latency data processing, exposing the limitations of conventional cloud-based architectures that rely on centralized resources. Fog computing addresses this challenge by distributing processing closer to end users. However, limited attention has been given to the impact of heterogeneous fog server capacities across multiple network layers on task allocation and energy consumption. This paper investigates the effects of heterogeneous fog servers on allocation efficiency and network energy performance. Three fog server categories with different processing capacities (34.2k MIPS, 73.44k MIPS, and 1080k MIPS) are evaluated using a Mixed Integer Linear Programming (MILP) model under both limited and unlimited deployment scenarios. The results demonstrate that the fog-based approach significantly reduces total energy consumption by up to 89.6% compared to the conventional cloud-based approach. Moreover, the findings reveal that higher processing server capacity plays a more critical role in improving energy efficiency than server heterogeneity alone, as greater capacity enables demand consolidation and reduces the number of active servers. These insights provide valuable guidance for the energy-efficient design of heterogeneous fog computing infrastructures.**

***Keywords*—energy consumption, fog computing, internet of things (IoT), mixed integer linear programming (MILP).**

I. INTRODUCTION

The rapid expansion of the Internet of Things (IoT) has resulted in a massive proliferation of connected devices, ranging from environmental and industrial sensors to wearable devices and smart home appliances. Each of these devices continuously generates large volumes of data, often in real time, creating what is commonly referred to as a "data deluge" [1]. Managing, processing, and storing this data poses significant challenges for both network and cloud infrastructures, as conventional systems are typically designed for centralized processing.

Many Internet of Things (IoT) devices are resource-constrained and have limited processing power, memory, and storage, which necessitates offloading computation to external servers [2]. While centralized cloud computing provides virtually unlimited computational resources on demand, transmitting all data from end devices to remote cloud servers can strain the network, leading to congestion, increased latency, and higher energy consumption for both data transmission and processing [3]. These issues are particularly pronounced in applications that are time-sensitive, such as autonomous vehicles, industrial automation, healthcare monitoring, and smart city operations, where even slight delays or inefficiencies in data handling can have critical consequences.

Fog computing has emerged as a promising paradigm to overcome the challenges posed by massive IoT data generation and network congestion. Fog computing extends computation and storage resources to the network edge, closer to the data source [4]-[5]. This decentralized approach reduces the need for long-distance data transfer, thereby improving response times and lowering energy consumption in the transport network [6]. Beyond latency and energy benefits, fog networks also enable distributed resource management, enhanced reliability, and scalability [7]. Fog architecture allows for flexible task allocation across multiple fog nodes, which is especially important in networks with heterogeneous device capacities and uneven data traffic. As IoT applications continue to proliferate, understanding how variations in fog server capacity influence task assignment and energy efficiency becomes critical. Investigating these aspects provides insights into optimal fog deployment strategies, ensuring that network resources are effectively utilized while maintaining energy-efficient operation under varying traffic demands.

The main contributions presented in this work are outlined below:

- Development of a mathematical optimization model using Mixed Integer Linear Programming (MILP) to optimize energy consumption of a fog-based network architecture that incorporates heterogeneous fog servers with multiple processing capacities to manage IoT traffic distribution effectively.
- Evaluate the impact of varying capacity levels on task allocation behavior and energy consumption across multiple network layers.

Manuscript received January 20, 2026; revised February 13, 2026. Date of publication October 13, 2026. Date of current version October 13, 2026. The associate editor prof. Claudia Canali has been coordinating the review of this manuscript and approved it for publication.

I. S. Md Isa is with the Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer, Universiti Teknikal Malaysia Melaka, Malaysia (e-mail: idasyafiza@utem.edu.my).

N. L. A. Mohd Shaari Azyze is with the Fakulti Teknologi dan Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka, Malaysia (e-mail: latifazyze@utem.edu.my).

W. H. Wan Hasan is with Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer, Universiti Teknikal Malaysia Melaka, Malaysia (e-mail: haszerila@utem.edu.my).

D. Mohd Ali is with the Wireless Communication Technology Group, Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia (e-mail: darma504@uitm.edu.my).

Digital Object Identifier (DOI): 10.24138/jcomss-2026-0013

1845-6421/03/2026-0013 © 2027 CCIS

- Provide insights into optimal fog deployment strategies, highlighting the trade-off between server capacity and network and processing energy performance.

The remainder of this paper is organized as follows: Section II presents the related work. Section III details the study's methodology and Section IV presents the architecture's evaluation results and corresponding analysis. Finally, Section V concludes by summarizing the key findings.

II. Related Work

Several studies have explored algorithms and frameworks for offloading tasks to fog nodes, incorporating techniques such as multi-criteria decision-making [8]-[10], stable matching [11], multi-objective optimization [12]-[14], and predictive scheduling [15]-[16]. These approaches demonstrate the potential to improve energy efficiency, task completion times, and load balancing across fog resources.

Despite these advances, the studies vary in the specific metrics they prioritize and the network scenarios they consider. For instance, a study by Sulimani et al. proposed a hybrid offloading algorithm to improve load balancing across fog servers [17]. The authors implemented the algorithm using iFogSim and conducted simulations to compare its performance against baseline and state-of-the-art approaches. They measured metrics such as the percentage of balanced fog servers and overall system execution time. The results showed that HybOff maintained approximately 97% balanced fog servers at 150 nodes and significantly improved resource utilization.

Furthermore, another line of work introduced an adaptive multi-criteria resource allocation method for fog-cloud environments [8]. Their approach combined Fuzzy TOPSIS and Fuzzy Analytic Hierarchy Process (FAHP) to rank fog devices and select optimal nodes for task execution. Simulation experiments evaluated load balancing, energy efficiency, and resource utilization. The method achieved an 11% reduction in active fog nodes, a 69% decrease in load-balancing variance, and resource utilization of around 90%. Additionally, Muhammad et al. proposed a stable-matching-based offloading algorithm that assigns tasks to fog nodes considering energy availability and proximity [11]. The researchers simulated different offloading scenarios and compared their approach with baseline methods in terms of latency and energy consumption. Results indicated reductions of approximately 31% in energy consumption and 30% in latency.

Meanwhile, Li et al. developed a multi-objective task-offloading algorithm that simultaneously considered energy consumption, task delay, task priority, and load balancing [14]. The tasks were divided into subtasks and assigned to fog nodes using a normalized multi-objective function that integrated all performance metrics. The results indicated that the proposed method achieved modest reductions in overall energy consumption, with energy usage decreasing by approximately 3–5% compared to baseline offloading schemes. Also, the results show that task delays were reduced by 4–6%, and the algorithm significantly improved fairness and load distribution, reducing load imbalance among fog nodes by up to 45%.

In [15], Kumar et al. presented a hybrid fog-cloud task scheduling scheme that combined deep learning for workload prediction with metaheuristic optimization for task assignment. The authors predicted task workloads and used a metaheuristic algorithm to optimize task allocation, aiming to minimize energy consumption and task make span while satisfying quality-of-service requirements. The results demonstrated that the proposed method reduced energy consumption by approximately 15–20% compared to baseline scheduling methods, while task completion times decreased by 10–15%. The study also showed that predictive workload modelling helped prevent overloading specific fog nodes. On top of that, a study by Wang et al. proposed a framework that used dynamic resource management and task offloading for fog computing, designed to handle the heterogeneous and dynamic nature of fog nodes [16]. The framework modelled the task offloading problem as a Markov Decision Process (MDP). It employed deep Q-learning (DQL) to determine optimal offloading decisions based on task demands, node states, and channel quality. Simulation experiments evaluated energy consumption, task completion time, and system reliability under varying network conditions. The results demonstrated that the DQL-based approach increased energy efficiency and reduced task completion time compared to traditional scheduling methods, while maintaining high reliability under dynamic workloads. The study also showed that adaptive, learning-based strategies can effectively manage heterogeneous fog resources, ensuring balanced allocation and robust performance across the network.

While significant progress has been made in fog computing research, several important aspects remain underexplored. To the best of our knowledge, most existing studies focus on optimizing task offloading, scheduling, or energy management, often assuming uniform server capacities or evaluating only a single network layer [18]-[21]. These approaches provide valuable insights into performance improvement; however, they do not fully address scenarios in which IoT traffic varies across geographic locations and network layers. In real-world deployments, some regions generate high volumes of data while others remain relatively low-traffic. To accommodate this uneven distribution efficiently, deploying fog servers with different computational capacities becomes necessary, allowing high-capacity servers to handle dense, high-demand areas and smaller servers to serve low-traffic regions. Although several studies consider heterogeneous fog server capacities and energy efficiency [14]-[16], they typically do not systematically examine how different fog server capacity levels affect task allocation behavior and overall energy consumption across multiple network layers. To bridge these gaps, this study investigates fog-based network architectures employing heterogeneous fog servers with multiple processing capacity levels and evaluates their impact on task allocation and energy efficiency across different network layers. This evaluation highlights how fog server capacity selection influences server allocation and network-wide energy performance.

III. Methodology

As reported in [22] Telekom Malaysia ONE has established two core data centres in Cyberjaya, Selangor, and Iskandar Puteri, Johor. These data centres are connected to the international gateway to provide services, including cloud

TABLE I
SUMMARY OF CONTRIBUTIONS AND STATE-OF-THE-ART APPROACHES IN FOG COMPUTING

Ref.	Approach	Heterogenous	Energy Consumption	Access Network	Metro Network	Key Finding
[17]	Hybrid Offloading (HybOff) Algorithm implemented in iFogSim to improve server load balancing.	Yes	No	Yes	No	Achieved 97% load balancing fairness and increased resource utilization by 50%.
[8]	Adaptive Multi-Criteria Allocation Combined Fuzzy TOPSIS and Fuzzy Analytic Hierarchy Process (FAHP).	Yes	No	Yes	No	Reduced active fog nodes by 11% and decreased load-balancing variance by 69%.
[11]	Stable-Matching Algorithm Assigns tasks based on energy availability and physical proximity.	Yes	Yes	Yes	No	Reduced energy consumption by approx. 31% and latency by 30%.
[14]	Multi-Objective Optimization Normalized function considering energy, delay, priority, and load balancing for subtasks.	Yes	Yes	Yes	No	Reduced load imbalance by up to 45% with a modest energy reduction of 3–5%.
[15]	Hybrid Fog-Cloud Scheduling Deep Learning (workload prediction) + Metaheuristic optimization.	Yes	Yes	Yes	No	Reduced energy consumption by 15–20% and task completion times by 10–15%.
[16]	Deep Q-Learning (DQL) Reinforcement learning framework (MDP) for dynamic resource management.	Yes	Yes	Yes	No	Maintained high reliability and superior energy efficiency under dynamic heterogeneous workloads.
Proposed	Heterogeneous Fog Architecture (MILP) Mixed Integer Linear Programming model evaluating varying server capacities across Access and Metro layers.	Yes	Yes	Yes	Yes	(Expected) Optimizes resource allocation across Access and Metro layers to minimize total power.

computing. In this study, only one core data centre located at Iskandar Puteri, Johor, is considered as the case study to evaluate the performance of the proposed fog architecture under heterogeneous fog server capabilities. Due to the complexity in running the model, we considered only four central regions, the Segamat, Kluang, Skudai and Johor Bahru, six districts and twelve cities. Note that, in this work, for the access layer, we considered the Gigabit Passive Optical Network (GPON. Fig. 1 shows the proposed fog architecture.

There are four main layers in the fog computing architecture, which are the IoT layer, the access network layer, the metro network layer and the core network layer. The IoT layer is responsible for generating the IoT data. The access network layer aggregates the raw data generated from the IoT network layer and forwards it either to the upper layer or to a processing server for computation. The metro network layer aggregates the data from the access layer and forwards it either to a processing server or to the core network layer. The core network layer is used to store the data processed at lower layers. It is also used to complete data processing in the conventional cloud-based approach.

It is worth noting that, in this work, the candidate locations for deploying processing servers in the fog computing approach are at the access and metro network layers. To benchmark our proposed fog-based approach, we considered a conventional approach in which the candidate location for deploying the processing server is in the core network layer. Also, three types of processing servers with different processing capabilities are considered for placement at the candidate locations to process the service demand. For this study, the terms processing server and fog server are used interchangeably to refer to the same computational units.

To model the energy consumption, we utilize a linear power consumption model [23] for both networking and processing components, which separates the total energy draw into two components: an idle power (baseline power consumed when the equipment is powered on but is not actively processing or forwarding any data), P_{idle} and a dynamic component that scales proportionally with the operational load (data traffic or computational demand), E. Fig. 2 shows the power profile model used in this work to determine the energy consumed by the networking equipment and also the processing equipment.

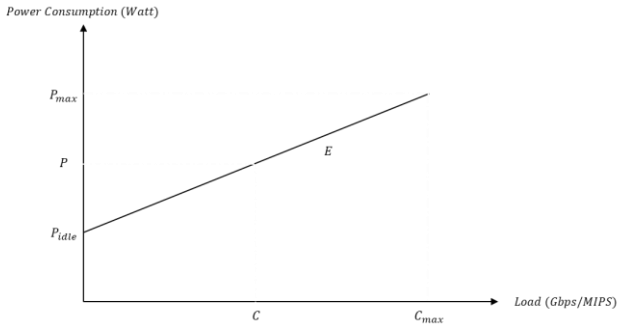


Fig. 2. Power profile model.

For this study, the optimization is carried out as a static snapshot for offline network planning. The service demand generated by IoT users is assumed to be finite and deterministic, and the processing capacities of processing servers are fixed and known. As the formulation is based on a Mixed Integer Linear Programming (MILP) framework, no assumptions regarding continuity, monotonicity, or smoothness of system functions are required. To determine the optimal configuration for the proposed fog computing approach, we developed the following MILP model. The model is developed using AMPL software and solved using CPLEX 12.5, executed on a 2.2 GHz processor. The objective function of the MILP model is to reduce the energy consumption of the networking at the access,

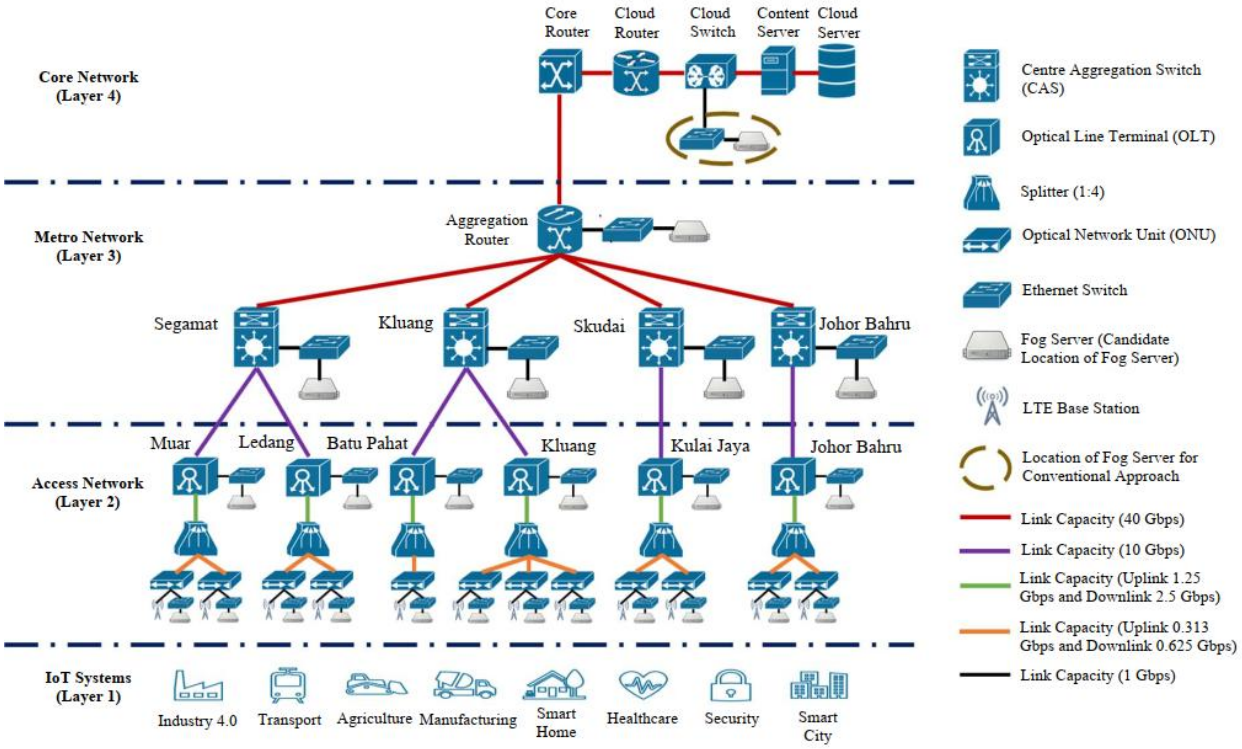


Fig. 1. Proposed Fog Computing Architecture.

$EAccess$ metro, $EMetro$ and core layers, $ECore$ and processing equipment, $EPro$ as in Equation (1). Meanwhile, Equations (2) – (5) are used to determine the energy consumed by the networking equipment for each layer and by the processing equipment. Note that the energy consumed by the networking equipment is based on its idle power, the activation of the devices, X_i, and its energy per bit, as well as the total traffic traversing the equipment, T_i. Meanwhile, energy consumed of the processing equipment is based on the idle power and the number of servers that are activated, NFS_i and its power per MIPS and the total processing demand i.e. number of users, wsd_{sd} and the processing demand per user, $MIPS$. It is also worth to note that, we consider the power usage effectiveness of 1.5 for access and metro layer while 2.5 for core layer and processing equipment [24].

Objective: Minimized

$$EAccess + EMetro + ECore + EPro \tag{1}$$

where

$$EAccess = 1.5 \cdot \left[\sum_{i \in Onu} X_i . Ionu + T_i . Eonu + \sum_{i \in Olt} X_i . Iolt + T_i . Eolt + \sum_{i \in Eth} X_i . Ieth + T_i . Eeth + \sum_{i \in Lte} X_i . Ilte + T_i . Elte \right] \tag{2}$$

$$EMetro = 1.5 \cdot \left[\sum_{i \in Cas} X_i . Icas + T_i . Ecas + \sum_{i \in Agr} X_i . Iagr + T_i . Eagr \right] \tag{3}$$

$$ECore = 2.5 \cdot \left[\sum_{i \in Crt} X_i . Icrt + T_i . Ecrt + \sum_{i \in Clr} X_i . Iclr + T_i . Eclr + \sum_{i \in Cls} X_i . Icls + T_i . Ecls + \sum_{i \in Cts} X_i . Icts + T_i . Ects + \sum_{i \in Cst} X_i . Icst + T_i . Ecst \right] \tag{4}$$

$$EPro = 2.5 \cdot \left[\sum_{i \in Fog} NFS_i \times Ifog + \sum_{i \in Fog} Wsd_{s,d} \times MIPS \times Efog \right] \tag{5}$$

Several constraints are also considered in this model. Constraint (6) is used to assign the users in each IoT group, $User_s$ to the fog server, $wsd_{s,d}$. The Y_d is a binary number, where 1 indicates the node d is used to place the server. Meanwhile, constraint (7) ensures that all users are assigned to a server.

$$Wsd_{s,d} \leq User_s \times Y_d \quad ; \quad \forall s \in IoT, d \in FOG : s \neq d \tag{6}$$

$$\sum_{d \in FOG: s \neq d} Wsd_{s,d} = User_s \quad ; \quad \forall s \in IoT \tag{7}$$

Constraint (8) is to determine the total traffic, Lsd_{sd} from the IoT group to the server. This is based on the assignment of users to the server. wsd_{sd} and the size of data per user, $Data$. Meanwhile, constraint (9) ensures that the incoming traffic equals the outgoing traffic at the intermediate node, except for the source and destination nodes.

$$Lsd_{s,d} = Wsd_{s,d} \times Data \quad ; \quad \forall s \in IoT, d \in FOG: s \neq d \tag{8}$$

$$\sum_{j \in Nm[i],\ i \neq j} Lsd_ij_{s,d,i,j} - \sum_{j \in Nm[i],\ i \neq j} Lsd_ij_{s,d,j,i} = \begin{cases} Lsd\,[s,d] & if\ i = s \\ -Lsd\,[s,d] & if\ i = d \\ 0 & otherwise \end{cases} \tag{9}$$

$$\forall s \in IoT, d \in FOG, i \in N: s \neq d$$

Constraint (10) is to ensure the total processing demand does not exceed the maximum capacity of the processing server, $CFOG_d$. Note that, variable NFS_d is used in the constraint to allow more than one server to be placed at each node (i.e., an unlimited scenario). Constraint (11) ensures that the servers will only be placed at the node if the node is activated, Y_d.

$$\sum_{s \in IoT} Wsd_{sd} \times MIPS \leq CFOG_d \times NFS_d \quad ; \quad \forall d \in FOG \tag{10}$$

$$Y_d \leq NFS_d \quad ; \quad \forall d \in FOG \tag{11}$$

Constraint (12) is to determine the link capacity between the Ethernet switch and the processing server based on the capacity of each link, λ and the number of servers deployed at the node, NFS_j. Constraint (13) is used to make sure that the total traffic traversing the networking equipment does not exceed the maximum capacity of the network link, $Link_{ij}$ and the link between the Ethernet switch and the fog node, Lsx_{ij}.

$$Lsx_{i,j} = \lambda \times NFS_j \ ; \ \forall i \in ETH, j \in FOG: i \neq j \tag{12}$$

$$\sum_{s \in IoT} \sum_{d \in FOG: s \neq d} Lsd_ij_{s,d,i,j} \leq Link_{i,j} + Lsx_{i,j} \ ; \tag{13}$$

$$\forall i \in N, j \in Nm[i]: i \neq j$$

Constraints (14) - (15) are used to determine the total traffic traversing each networking equipment, T_i based on the total traffic traversing each link, $Trafficc_{ij}$. Meanwhile, constraint (16) defines the active node that is used to relay the traffic, X_i. Note that M is a significant number that is used to ensure binary X_i is equal to 1 if the traffic traversing node i.

$$\sum_{s \in IoT} \sum_{d \in FOG: s \neq d} Lsd_ij_{s,d,i,j} = Trafffic_{i,j} \ ; \quad \forall i \in N, j \in Nm[i]: i \neq j \tag{14}$$

$$\sum_{j \in Nm[i]: i \neq j} Trafffic_{i,j} = T_i \ ; \ \forall i \in N \tag{15}$$

$$X_i \geq \sum_{j \in Nm[i]: i \neq j} Traffic_{i,j} \ ; \ \forall i \in N \tag{16}$$

$$X_i \leq M \times \sum_{j \in Nm[i]: i \neq j} Trafffic_{i,j} \ ; \ \forall i \in N$$

Note that the MILP formulation for the proposed fog-based approach and the conventional cloud-based approach are the same as in equations (1) – (16). However, for the fog-based approach, the candidate locations of the fog server, FOG are at the access network layer i.e., optical network equipment (ONU) and optical line terminal (OLT), and the metro network layer i.e., aggregation switch (AGS) and aggregation router (AGR). Meanwhile, for the conventional approach, the candidate location of the fog servers, FOG is at the core network layer, i.e., cloud switch.

IV. Results and Discussions

This section presents the results and analysis of the proposed fog-based approach in terms of the energy consumption of the networking and processing equipment. As explained previously, we utilize a linear model to characterize the power consumption of the networking and processing components. It is worth noting that the results are based on an offline planning evaluation using deterministic demand. This approach enables a systematic comparison of energy performance across various fog deployment strategies and server capacities. Table I shows the maximum power consumption, P_{max}, the idle power consumption, P_{idle}, and the maximum capacity of the networking and processing equipment, C_{max}, used to evaluate the proposed architecture. Notably, this analysis excludes the energy consumption of associated IoT equipment.

It is worth noting that the Energy (J), measured in Joules, represents the total amount of power utilized over a duration of time. This relationship can be expressed as:

$$Energy\ (Joule) = Power\ (W) \times Time\ (s) \tag{17}$$

Since the operational time of the proposed model is precisely one second, its total energy consumption is mathematically identical to its average power consumption, as shown by Equation (17).

Two main scenarios are considered to evaluate the performance of the proposed fog-based approach. In the first scenario, the model allows more than one processing server to be deployed at each candidate node. Meanwhile, in the second scenario, the deployment is limited to only one processing server per candidate node. These scenarios are essential for evaluating the impact of server deployment density on energy efficiency. To ensure the robustness of the findings, the evaluation spans a diverse range of scenarios, including varying service demand volumes, multiple levels of server

heterogeneity, and both limited and unlimited server deployment constraints.

TABLE II
INPUT PARAMETERS FOR NETWORKING AND PROCESSING EQUIPMENT

Equipment	P_{max} [w]	P_{idle} [w]	C_{max}
Cloud Storage, CST	4900 [23]	463.10 [25]	75.6 TB [23]
Content Server, CTS	380.80 [23]	324.82 [23]	1.80 Gbps [23]
Cloud Switch, CLS	2020 [23][26]	1818 [26]	320 Gbps [23]
Cloud Router, CLR	4550 [23]	4095 [23]	560 Gbps [23]
Core Router, CRT	12300 [23]	11070 [23]	4480 Gbps [23]
Aggregation Router, AGR	4550 [23]	4095 [23]	560 Gbps [23]
Centre Aggregation Switch, CAS	1766 [23]	1589 [23]	256 Gbps [23]
Optical Line Terminal, OLT	1940 [23]	60 [23]	8600 Gbps [23]
Optical Network Unit, ONU	15 [23]	9 [23]	0.3 Gbps [23]
LTE-M Base Station, LTE	528 [27]	333 [27]	0.3 Gbps [27]
Ethernet Switch, ETH	3.52 [23]	0.57 [23]	16 Gbps [23]
Fog Server A, FSA	95 [28]	57 [28]	34.2k MIPS [28]
Fog Server B, FSB	95 [29]	57 [29]	73.44k MIPS [29]
Fog Server C, FSC	74.20 [30]	45 [30]	1080k MIPS [30]

A. *Unlimited Scenario*

In this scenario, the model allows more than one processing server to be deployed at each candidate node. Also, different types of processing servers and heterogeneous combinations are considered, i.e., processing servers A and B, as well as processing servers A, B, and C, to evaluate performance under the fog-based approach. This scenario is essential for assessing the maximum potential energy efficiency and performance of the fog-based approach when there are no deployment restrictions, as it allows the system to use as many servers as needed to meet service demands.

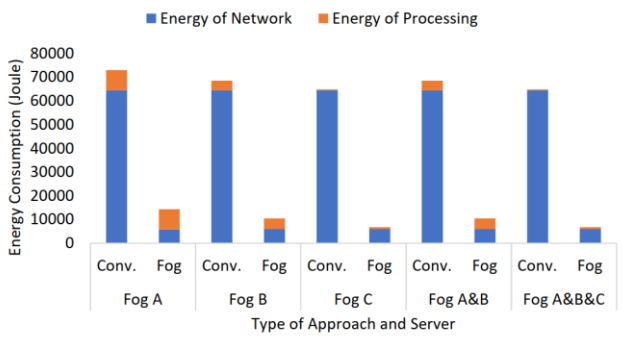


Fig. 3. Total energy consumption of fog-based approach and conventional approach for different processing server capacities.

Fig. 3 illustrates the total energy consumption of the fog-based and conventional approaches under different processing server capacity configurations. The results show that the fog-based approach achieves lower total energy consumption than the conventional approach for all processing server capacities. The lower total energy in the fog-based approach is primarily due to reduced networking power. By placing the processing servers near service demands, the system minimizes the volume of data traversing the network. In contrast, in the conventional approach, data must traverse networking equipment across all layers to reach the processing server at the cloud layer, thereby increasing the energy consumption of the networking equipment.

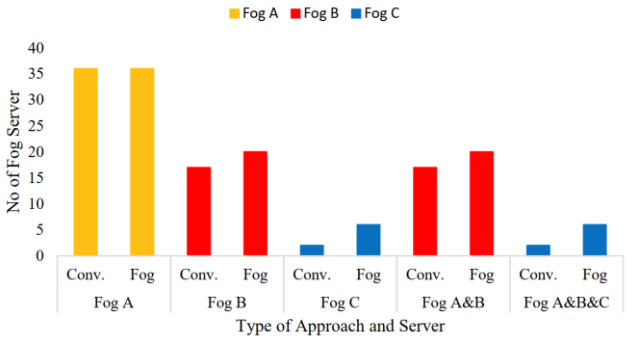


Fig. 4. Number of processing servers utilized in the fog-based approach and the conventional approach for different processing server capacities.

The results also reveal that the total energy consumed for both approaches, decreases as processing server capacity increases. Specifically, the energy savings achieved by the fog-based approach compared to the conventional approach are approximately 80.4%, 84.8%, and 89.6% for processing server A, processing server B, and processing server C, respectively. This improvement is due to the reduced number of utilized processing servers, as more service demands can be consolidated in the processing server, as shown in Fig. 4. Fig. 3 also indicates that when processing servers A and B are allowed to be deployed in the network, the total energy consumption increases in both approaches, resulting in an energy saving of 84.8% for the fog-based approach. However, when all three types of processing servers are permitted, the fog-based approach achieves an energy saving of 89.6%. It is worth noting that the energy savings in the fog-based approach for processing servers A and B are identical to those obtained when only server B is considered, whereas the energy savings for processing servers A, B, and C are similar to those obtained when only server C is considered. This is due to the same number of processing servers used in these scenarios, as shown in Fig. 4. These results highlight that higher processing server capacity is crucial to improving the overall energy efficiency of the fog-based approach, rather than the deployment of heterogeneous processing servers.

We also assess the performance of the fog-based approach under increasing traffic conditions for different processing server capacities and heterogeneous processing server deployment. We consider the service demand required by a user to be 1000 MIPS, 2000 MIPS, and 3000 MIPS. Note that increasing the service demand also increases the amount of data traversing the network. Fig. 5 illustrates that increasing traffic, i.e., service demand, increases the total energy consumed. This is due to the increased amount of data that must be transmitted and processed in the network, hence increasing the energy

consumption of the networking and processing equipment. However, the results in Fig. 5 indicate that processing energy consumption plays a dominant role in the total energy consumption, particularly when low-capacity processing servers are considered, i.e., processing server A, processing server B, and a combination of processing server A and B. This is shown in Fig. 6, where, when a high-capacity processing server C is considered, as well as when heterogeneous processing servers A, B, and C are deployed, the same number of servers is utilized to serve the increasing demand. This highlights that higher-capacity processing servers enable greater consolidation of service demands. It is worth noting that although heterogeneous processing servers can be deployed in the network, higher processing server capacity plays a more significant role in reducing total energy consumption than server heterogeneity. This is because higher-capacity processing servers can accommodate more service demands within fewer active servers, thereby improving overall energy efficiency.

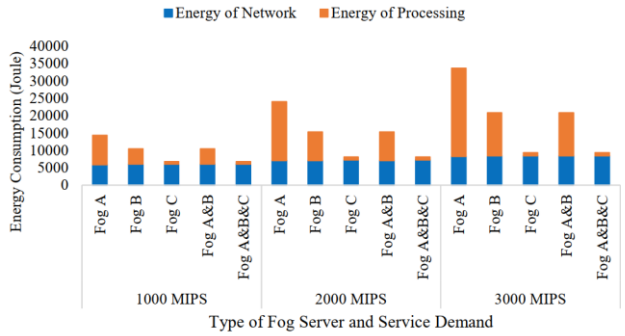


Fig. 5. Total energy consumption of the fog-based approach with increasing traffic for different processing server capacities.

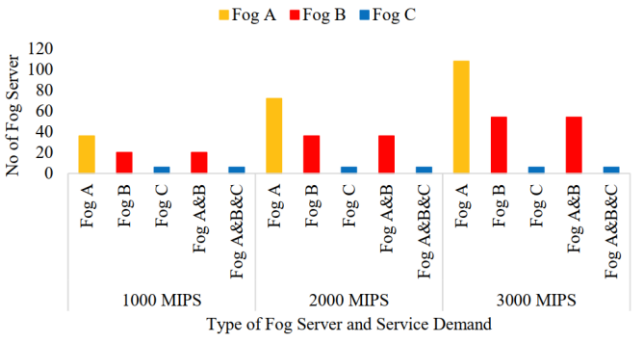


Fig. 6. Number of processing servers utilized in the fog-based approach with increasing traffic for different processing server capacities.

B. *Limited Scenario*

In this scenario, the deployment of the processing server is limited to one per candidate node. Therefore, we include constraint (18), where the total number of servers at each candidate node, NFS_d should not exceed more than one, as below;

$$NFS_d \leq 1 \;\; ; \;\; \forall d \in FOG \tag{18}$$

We consider only heterogeneous combinations of processing servers, which are processing servers A and B, as well as processing servers A, B, and C, to assess the performance of the fog-based approach under deployment constraints. Note that each user requires a service demand of 1000 MIPS. This scenario is essential for examining the impact of deployment restrictions on the fog-based approach's energy efficiency. Besides, it also reflects practical limitations in real-world fog computing deployments.

The results in Fig. 7 indicate that considering higher processing server capacities in heterogeneous combinations leads to lower total energy consumed in both limited and unlimited scenarios. This is primarily due to a significant decrease in the energy consumed by the processing server, as fewer processing servers are used to serve the same demand, as shown in Fig. 8. Besides, Fig. 8 shows that only processing server C, which has the highest processing capacity, is utilized in both the limited and unlimited scenarios to minimize the number of active servers.

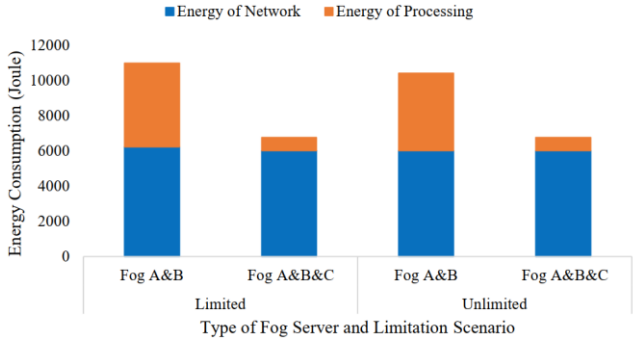


Fig. 7. Total energy consumption of the fog-based approach under limited and unlimited scenarios with different heterogeneous combinations of processing servers.

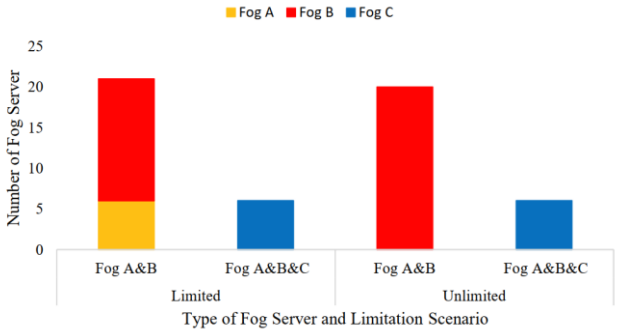


Fig. 8. Number of processing servers utilized in the fog-based approach under limited and unlimited scenarios with different heterogeneous combinations of processing servers.

The results in Fig. 7 also show that the unlimited scenarios yield lower total energy consumption than the limited scenario when processing servers with low and medium capacities are considered. This is because, in the unlimited scenario, the total number of processing servers in use is reduced, with most servers deployed near service demand. This reduces the total amount of data traversing the network equipment, since service demands can be processed closer to the users. In contrast, limiting the number of servers to deploy at each candidate node increases the number of servers in use. As illustrated in Fig. 8, both processing servers A and B are activated to meet the demands. This leads to lower server utilization efficiency.

However, the results in Fig. 7 further show that when the processing server's capacity is high within a heterogeneous combination, the total energy consumption is identical in both scenarios. This is because the number of processing servers utilized in both scenarios is identical, as shown in Fig. 8. This finding demonstrates that the limiting scenario resulted in higher energy consumption than the unlimited scenario. However, this increase can be mitigated by deploying a higher-capacity processing server rather than relying on multiple lower-capacity processing servers in a heterogeneous combination.

V. Conclusion

This paper investigated the impact of deploying heterogeneous processing servers on the energy efficiency of a fog-based approach compared to the conventional cloud-based approach. Also, to provide baseline comparisons and to better understand the impact of server capacity on overall system performance, individual processing server types were also considered. A Mixed Integer Linear Programming (MILP) model was developed with the aim of minimizing the total energy consumption of the networking and processing equipment across multiple network layers. The results demonstrate that, compared to the cloud-based approach, the fog-based approach consistently achieves significant energy savings primarily due to reduced networking energy consumption as processing servers are deployed closer to users. Furthermore, the findings reveal that while heterogeneous processing server deployments offer flexibility, higher processing server capacity plays a more decisive role in improving energy efficiency. This is because it enables greater consolidation of service demands, hence reducing the number of active servers. We also evaluate the impact of restricting the number of processing servers deployed at each candidate node, i.e., a limited scenario, on the performance of the fog-based approach. In both limited and unlimited deployment scenarios, similar energy consumption levels were observed when heterogeneous deployments included the highest-capacity processing server, as the same number of servers was utilized. This indicates that processing capacity is more influential than heterogeneity alone. Although deployment constraints generally lead to higher energy consumption, this effect can be mitigated by deploying higher-capacity processing servers. Future work will extend this study to better reflect realistic operating conditions and enable scalable deployment. This includes developing heuristic methods inspired by the MILP results for large-scale scenarios and exploring multi-objective optimization to balance energy efficiency with latency constraints and dynamic traffic demands.

Acknowledgement

This research article work is funded by the Ministry of Higher Education (MOHE) of Malaysia through the Fundamental Research Grant Scheme (FRGS), No: FRGS/1/2023/ICT08/UTEM/03/01 and UTeM Short Term Grant Scheme (PJP), No: PJP/2025/FTKE/PERSPEKTIF/SA0086.

References

[1] S. Mazloom and S. D. Gordon, "Secure Computation with Differentially Private Access Patterns," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, New York, NY, USA: ACM, Oct. 2018, pp. 490–507. doi: 10.1145/3243734.3243851.

[2] I. Radhakrishnan, S. Jadon, and P. B. Honnavalli, "Efficiency and Security Evaluation of Lightweight Cryptographic Algorithms for Resource-Constrained IoT Devices," *Sensors*, vol. 24, no. 12, p. 4008, Jun. 2024, doi: 10.3390/s24124008.

[3] A. Badshah, A. Daud, O. I. Aboulola, A. Bukhari, B. Alshemaimri, and H. Dawood, "Network optimization by regional computing for UAVs' big data," *Front. Comput. Sci.*, vol. 7, Jun. 2025, doi: 10.3389/fcomp.2025.1565716.

[4] White Paper, "Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are What You Will Learn," 2015.

[5] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Fog Computing and the Internet of Things (IoT): A Review," in *Proceedings - 2021 8th IEEE International Conference on Cyber Security and Cloud Computing and 2021 7th IEEE International Conference on Edge Computing and Scalable Cloud, CSCloud-EdgeCom 2021*, IEEE, 2021, pp. 10–12. doi: 10.1109/CSCloud-EdgeCom52276.2021.00012.

[6] T. Kim, S. Yoo, and Y. Kim, "Edge/Fog Computing Technologies for IoT Infrastructure," *Sensors*, vol. 21, no. 9, p. 3001, Apr. 2021, doi: 10.3390/s21093001.

[7] A. Reyana, S. Kautish, K. A. Alnowibet, H. M. Zawbaa, and A. Wagdy Mohamed, "Opportunities of IoT in Fog Computing for High Fault Tolerance and Sustainable Energy Optimization," *Sustainability*, vol. 15, no. 11, p. 8702, May 2023, doi: 10.3390/su15118702.

[8] A. A. A. Gad-Elrab, A. S. Alsharkawy, M. E. Embabi, A. Sobhi, and F. A. Emara, "Adaptive Multi-Criteria-Based Load Balancing Technique for Resource Allocation in Fog-Cloud Environments," *International Journal of Computer Networks and Communications*, vol. 16, no. 1, pp. 105–124, Jan. 2024, doi: 10.5121/IJCNC.2024.16107.

[9] F. Alenizi and O. Rana, "Minimising Delay and Energy in Online Dynamic Fog Systems," Academy and Industry Research Collaboration Center (AIRCC), Nov. 2020, pp. 139–158. doi: 10.5121/csit.2020.101513.

[10] R. Naha, S. Garg, S. K. Battula, M. B. Amin, and D. Georgakopoulos, "Multiple linear regression-based energy-aware resource allocation in the Fog computing environment," *Computer Networks*, vol. 216, Oct. 2022, doi: 10.1016/j.comnet.2022.109240.

[11] W. N. W. Muhamad *et al.*, "Energy-efficient task offloading in fog computing for 5G cellular network," *Engineering Science and Technology, an International Journal*, vol. 50, p. 101628, Feb. 2024, doi: 10.1016/J.JESTCH.2024.101628.

[12] A. Gopu *et al.*, "Energy-efficient virtual machine placement in distributed cloud using NSGA-III algorithm," *Journal of Cloud Computing*, vol. 12, no. 1, Dec. 2023, doi: 10.1186/s13677-023-00501-y.

[13] C. Guerrero, I. Lera, and C. Juiz, "Evaluation and efficiency comparison of evolutionary algorithms for service placement optimization in fog architectures," *Future Generation Computer Systems*, vol. 97, pp. 131–144, Aug. 2019, doi: 10.1016/j.future.2019.02.056.

[14] S. K. Panda, T. Pounjula, B. Ravirala, and D. Taniar, "An energy, delay and priority-aware task offloading algorithm for fog computing incorporating load balancing," *The Journal of Supercomputing 2024 81:1*, vol. 81, no. 1, pp. 52-, Oct. 2024, doi: 10.1007/S11227-024-06557-0.

[15] A. Heirati, M. Fartash, and M. Khalily-Dermany, "Optimized Task Scheduling in Fog-Cloud Computing Using Hybrid Deep Learning and Metaheuristic Algorithms," *Neural Processing Letters 2025*, Dec. 2025, doi: 10.1007/S11063-025-11819-W.

[16] H. M. Abdelghany, "Dynamic Resource Management and Task Offloading Framework for Fog Computing," *Journal of Grid Computing 2025 23:2*, vol. 23, no. 2, pp. 19-, May 2025, doi: 10.1007/S10723-025-09804-7.

[17] H. Sulimani *et al.*, "HybOff: a Hybrid Offloading approach to improve load balancing in fog environments," *Journal of Cloud Computing 2024 13:1*, vol. 13, no. 1, pp. 113-, May 2024, doi: 10.1186/S13677-024-00663-3.

[18] D. Sahu *et al.*, "Optimizing energy and latency in edge computing through a Boltzmann driven Bayesian framework for adaptive resource scheduling," *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-16317-6.

[19] G. Li and J. Li, "Optimizing low-power task scheduling for multiple users and servers in mobile edge computing by the MUMS framework," *Heliyon*, vol. 10, no. 11, Jun. 2024, doi: 10.1016/j.heliyon.2024.e31622.

[20] X. Gu, C. Ji, and G. Zhang, "Energy-optimal latency-constrained application offloading in mobile-edge computing," *Sensors (Switzerland)*, vol. 20, no. 11, Jun. 2020, doi: 10.3390/s20113064.
[21] J. Yan, Q. Lu, and G. B. Giannakis, "Bayesian Optimization for Task Offloading and Resource Allocation in Mobile Edge Computing," in *Conference Record - Asilomar Conference on Signals, Systems and Computers*, IEEE Computer Society, 2022, pp. 1086–1090. doi: 10.1109/IEEECONF56349.2022.10051868.
[22] Telekom Malaysia, "Digital Nation Agenda, Hyperconnected Ecosystem: Unloacking Potentials of Cities and Industries," TM ONE. Accessed: Feb. 11, 2021. [Online]. Available: https://www.tmone.com.my/resources/think-tank/tag/data-centre-services/ence-with-end-to-end-cloud-infrastructure/
[23] I. S. B. M. Isa, T. E. H. El-Gorashi, M. O. I. Musa, and J. M. H. Elmirghani, "Energy Efficient Fog-Based Healthcare Monitoring Infrastructure," *IEEE Access*, vol. 8, pp. 197828–197852, 2020, doi: 10.1109/ACCESS.2020.3033555.
[24] I. S. M. Isa, T. E. H. El-Gorashi, M. O. I. Musa, and J. M. H. Elmirghani, "Energy efficient fog based healthcare monitoring infrastructure," *IEEE Access*, vol. 8, pp. 197828–197852, 2020, doi: 10.1109/access.2020.3033555.
[25] "Storage Performance Council – Silverton Consulting." Accessed: May 28, 2024. [Online]. Available: https://silvertonconsulting.com/tag/storage-performance-council/
[26] A. Vishwanath, F. Jalali, R. Ayre, T. Alpcan, K. Hinton, and R. S. Tucker, "Energy consumption of interactive cloud-based document processing applications," *IEEE International Conference on Communications*, pp. 4212–4216, 2013, doi: 10.1109/ICC.2013.6655224.
[27] I. S. M. Isa, M. O. I. Musa, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "Energy efficient and resilient infrastructure for fog computing health monitoring applications," *International Conference on Transparent Optical Networks*, vol. 2019-July, Jul. 2019, doi: 10.1109/ICTON.2019.8840438.
[28] "Intel Xeon Processor E52420 15M Cache 1.90 GHz 7.20 GTs Intel QPI Product Specifications." Accessed: Jun. 08, 2024. [Online]. Available: https://ark.intel.com/content/www/us/en/ark/products/64617/intel-xeon-processor-e5-2420-15m-cache-1-90-ghz-7-20-gt-s-intel-qpi.html
[29] "Intel® Xeon® Processor X5675." Accessed: Jun. 08, 2024. [Online]. Available: https://www.intel.com/content/www/us/en/products/sku/52577/intel-xeon-processor-x5675-12m-cache-3-06-ghz-6-40-gts-intel-qpi/specifications.html
[30] "NVIDIA T4 Tensor Core GPU for AI Inference | NVIDIA Data Center." Accessed: Jun. 08, 2024. [Online]. Available: https://www.nvidia.com/en-us/data-center/tesla-t4/



Ida Syafiza Md Isa received her Ph.D. degree from the University of Leeds, U.K., in 2020, and worked on energy-efficient access network design for Internet of Things (IoT) applications. She is currently a Senior Lecturer with Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. She has published several articles in this area. Her research interests include network architecture design, energy efficiency, network optimization, mixed integer linear programming, and IoT healthcare systems.



Nur Latif Azyze Mohd Shaari Azyze is a Senior Lecturer at Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia. He holds Ph.D. degree from the University of Leeds, U.K., in 2022, M.Eng degree in Electrical Mechatronics and Control Engineering, and B.Eng in Electrical Mechatronics from Universiti Teknologi Malaysia. His research area is embedded systems, robotics, rehabilitation engineering, and the Internet of Things.



Wan Haszerila Wan Hassan is a lecturer at the Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. She received her first degree in Electrical and Electronic Engineering and her master's degree in Electrical, Electronic, and Telecommunication Engineering from Universiti Teknologi Malaysia (UTM), Malaysia, in 2011 and 2013, respectively. She has completed her Ph.D. degree with the Faculty of Electrical Engineering, Universiti Teknologi MARA (UiTM), Malaysia in 2025. Her research interests include wireless communication and QoS provisioning in wireless networks.



Darmawaty Mohd Ali is an Associate Professor at the Faculty of Engineering, Universiti Teknologi MARA (UiTM). In 2012, she was awarded a PhD from Universiti Malaya (UM). She received her first degree from Universiti Kebangsaan Malaysia (UKM), with Honors in Electrical, Electronic, and Systems Engineering, graduating in 1999. She began her career as a product engineer before pursuing her master's degree at Universiti Teknologi Malaysia (UTM). Her research interests include wireless access technology and the provision of quality of service in wireless networks.