UNIVERSITY OF LEEDS

Energy Efficient Distributed Processing for IoT

Barzan Abdulla Yosuf

University of Leeds

School of Electronic and Electrical Engineering

Submitted in accordance with the requirements for the degree

of

Doctor of Philosophy

October, 2019

Intellectual Property Statement

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

The work in Chapter 4 has appeared or will partially appear in publications as follows:

B. Yosuf, M. Musa, T. Elgorashi, A. Q. Lawey and J. M. H. Elmirghani, "Energy Efficient Service Distribution in Internet of Things," *2018 20th International Conference on Transparent Optical Networks (ICTON)*, Bucharest, 2018, pp. 1-4.

The candidate developed the energy efficient resource distribution model and using it to improve the end to end energy efficiency of IoT based service, considering heterogeneous layers of processing.

M. Musa and T. Elgorashi reviewed and validated findings throughout the work.

A. Q. Lawey supported in early design and development of the model.

Professor Elmirghani reviewed the paper and helped with the flow of content and suggested the review scope.

B. A. Yosuf, M. Musa, T. Elgorashi and J. M. H. Elmirghani, "Impact of Distributed Processing on Power Consumption for IoT Based Surveillance Applications," *2019 21st International Conference on Transparent Optical Networks (ICTON)*, Angers, France, 2019, pp. 1-5.

The candidate extended on the previously published idea and considered the prospect of service splitting on reducing energy efficiency of distributed processing for resource intensive applications.

M. Musa and T. Elgorashi reviewed and validated findings throughout the work.

A. Q. Lawey supported in early design and development of the model.

Professor Elmirghani reviewed the paper and helped with the flow of content and suggested the review scope.

Acknowledgements

First and foremost, all praise is due to Allah, for His countless blessings and endless gifts He has bestowed upon me. I invoke peace and blessings on His messenger Mohammad ﷺ [1], for which he said: " He who does not thank the people is not thankful to Allah".[2] I wish to acknowledge my supervisor, Professor Jaafar Elmirghani for his leadership, guidance and patience throughout my entire PhD journey. His continued and constant encouragement is much appreciated. I would also like to show my deepest gratitude to Dr Taisir and Dr Mohammad Musa for their continued support daily, their support and guidance have helped me achieve my goals successfully.

I wish to show my thanks to my wife, Ala, for her unconditional love and sacrifice during some of the most challenging moments in my PhD. Thank you for supporting me to achieve my goals. I wish to thank my parents and the coolness of my eyes, my siblings Rosie, Bahra, Arazu, Mohammad and Barham. You all mean everything good to me in this life. I love you all so much. I wish to thank my mum for keeping me in her prayers and my dad for his constant support throughout my PhD journey. I pray to Allah for their good health and wellbeing.

I would also like to thank my colleagues Sana, Amal, Ida, Randa, Hatam, Osama, Abdulla, Mohammad Hadi, Opeyemi. I enjoyed sharing the same office with them and I thank them for their company and fruitful discussions.

---

1 The Arabic phrase which translates as "peace be upon him".

2 https://sunnah.com/abudawud/43/39

Abstract

The number of connected objects in the Internet of Things (IoT) is growing exponentially. IoT devices are expected to number between 26 billion to 50 billion devices by 2020 and this figure can grow even further due to the production of miniaturised portable devices that are lightweight, energy and cost efficient together with the widespread use of the Internet and the added value organisations and individuals can gain from IoT devices, if their data is processed. These connected objects are expected to be used in multitudes of applications, of which, some are, highly resource intensive such as visual processing services for surveillance based object recognition applications. The sensed data requires processing by the cloud in order to extract knowledge and make decisions accordingly. Given the pervasiveness of future IoT-based visual processing applications, massive amounts of data will be collected due to the nature of multimedia files. Transporting all that collected data to the cloud at the core of the network, is prohibitively costly, in terms of energy consumption.

Hence, to tackle the aforementioned challenges, distributed processing is proposed by academia and industry to make use of a large number of devices located in the edge of the network to process some or all of the data before it gets to the cloud. Due to the heterogeneity of the devices in the edge of the network, it is crucial to develop energy efficient models that take care of resource provisioning optimally. The focus in today's network design and development has shifted towards energy efficiency, due to the rising cost of electricity, resource scarcity and increasing emission of carbon dioxide (CO_2).

This thesis addresses some of the challenges associated with service placement in a distributed architecture such as the fog. First, a Passive Optical Network (PON) is used to connect IoT devices and to support the fog infrastructure. A metro network is also used to connect to the fog and aggregate traffic from the PON towards the core network. An IP/WDM backbone network is considered to model the core layer and to interconnect the cloud data centres. The entire network was modelled and optimised through Mixed Integer Linear Programming (MILP) and the total end to end power consumption was jointly minimised for processing and networking. Two aspects of service placements were examined: 1) non-splitable services, and 2) splitable services. The results obtained showed that, in the capacitated problem, service splitting introduced power consumption savings of up to 86% compared to 46% with non-splitable services. Moreover, an energy efficient special purposed data centre (SP-DC) was deployed in addition to its general purpose counterpart (GP-DC). The results showed that, for very high demands, power savings of up to 50% could be achieved compared to 30% without SP-DC.

The performance of the proposed architecture was further examined by considering additional dimensions to the problem of service placements such as resiliency dimension in terms of 1+1 server protection, in the long term network design problem (un-capacitated) and the impact of inter-service synchronisation overhead on the total number service splits per task.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AP	Access Point
CO_2	Carbon Dioxide
CPE	Customer Premises Equipment
CPU	Central Processing Unit
DC	Data Centre
DSL	Digital Subscriber Liner
EDFA	Erbium Fibre Amplifier
FDM-PON	Frequency Division Multiplexing Passive Optical Network
GB	Gigabyte
Gbps	Gigabit Per Second
GHz	Giga Hertz
GP-DC	General Purpose Data Centre
GPU	Graphical Processing Unit
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology
IoT	Internet of Things
IP/WDM	Internet Protocol over Wave Length Division Multiplexing
LAN	Local Area Network
LTE	Long Term Evolution
mAh	Milliampere-hour
Mbps	Megabits Per Second
MILP	Mixed Integer Linear Programming
NaaS	Network as a Service

OFDM-PON	Orthogonal Frequency Division Multiplexing Passive Optical Network
OLT	Optical Line Terminal
ONU	Optical Network Unit
PaaS	Platform as a Service
PON	Passive Optical Network
PUE	Power Usage Effectiveness
QoS	Quality of Service
RFID	Radio Frequency Identification
SaaS	Software as a Service
SOA	Service Oriented Architecture
SP-DC	Special Purpose Data Centre
TDM-PON	Time Division Multiplexing Passive Optical Network
VSN	Virtual Sensor Network
W	Watt
WDM	Wavelength Division Multiplexing
WiFi	Wireless Fidelity

Chapter 1 Introduction

This chapter serves as the basis for motivating the work in this thesis on improving the energy efficiency of distributed processing in the context of IoT. The problem statement is presented and the tools and methodologies utilised in this thesis are summarised. The objectives and contributions are provided and the relevant publications are listed. Finally, an outline of the remainder of the thesis is provided.

1.1 Motivation

The number of connected objects in the Internet of Things (IoT) is growing at unprecedented levels. In 2011, this number surpassed the world's population and by 2020, interconnected devices are expected to range between 26 billion to 50 billion devices [1], [2]. This increase in the number of connected objects is directly linked to the technological advancement in the past decades as this has enabled the production of miniaturised portable devices that are light weight, energy and cost efficient together with the widespread use of the Internet and the added value organisations and individuals can gain from IoT devices if their data is processed. Hence, these trends have made it attractive to integrate and connect almost anything to the Internet which eventually leads to the concept of IoT [3],[4].

There will be multitudes of IoT applications, some are already in existent while others are yet to be realised. Thus, massive amounts of data will be produced, that, if processed centrally by conventional clouds would lead to slow decision making and increased pressure on the already overwhelmed network. Autonomous vehicles, for example, are reported to generate data that is in the range of 1 gigabyte per second [5]. It is evidently clear that transporting all of this data to the cloud for processing is prohibitively costly in terms of bandwidth requirements and energy efficiency. In the past, the main focus of Information and Communication Technologies (ICT) was fixated on performance only. Little or no attention was paid to the amount of power ICT based components consumed and their adverse impact on our environment. The focus has now shifted towards energy efficiency, due to the rising cost of electricity, resource scarcity and increasing emission of carbon dioxide (CO_2) [6]. It is reported that the emission of CO_2 due to ICT based technologies is increasing at an alarming rate of 6% per year. Given this growth rate, the Internet can become responsible for up 12% of the global emissions by 2030 and cloud data centres which are at the heart of the IoT are one of the major components of ICT [7].

In this direction, distributed processing has been proposed by industry and academia as an effective strategy to curb the pressure imposed by the formidable scale of IoT. Fog computing is one of the distributed processing approaches which promises to tackle the aforementioned challenges by utilising the already available computational, storage, and networking resources in serving IoT data at the edge of the network, as close as possible to the source [8]. Oftentimes, decision making can be made better and quicker if the collected data is processed closer to the source [9]. Currently, fog

computing is still in its infancy and a standardised architecture has yet to be agreed. Thus, alternative IoT architectures are increasingly been studied in the research community in terms of efficient resource management and the interplay between fog devices and the cloud, since fog is regarded as a powerful complement to the cloud [10]. A proper resource management scheme is crucial in the fog, since application services can be placed on energy inefficient servers or even further from the source node which may result in higher communication latency as contemporary fog devices have limited processing, storage and communication power [6]. It is expected that through cooperation between fogs and the centralised cloud, a more efficient and greener computing platform can be achieved [11].

1.2 Problem Statement

A large number of fog devices exist at the edge of the network, which collectively provides enormous amounts of computational resources, that, if used, may help in curbing the unnecessary data exchange with the centralised cloud. In order to integrate the vast number of fog devices that are heterogeneous in terms of resources, proper resource management and network design frameworks are needed. These should take into account important dimensions such as energy efficiency, due to its impact on our environment, resilience, due to mission-critical services, and inter-service communications due to end-device cooperation. This study aims first to model the IoT infrastructure from an end-to-end perspective such that all layers of the networking domain are taken into account when an IoT service is launched from the end-device to the ultimate destination in the cloud which is located in the core network.

Passive Optical Networks (PON) have been proposed to support the distributed processing infrastructure as they are increasingly utilised due to their suitability for visual-based services as they provide high data rates, relatively low cost and are very scalable [12]. Several design factors that affect the power consumption of the distributed processing approach are considered. Those include the Power Usage Effectiveness (PUE) of the higher capacity fog layers, the core network and the cloud DC layer. PUE is the ratio of the total power consumption of a node (including cooling and lighting) to the power consumption of the communication and processing equipment. The studies in this thesis included investigations into the short term design problems (capacitated) and the long term ones (un-capacitated) in designing energy efficient network architectures [13].

1.3 Research Objectives

The work reported in this thesis has the following objectives:

1. To model and evaluate an end-to-end IoT infrastructure that is supported by fog and cloud processing.
2. To study capacitated and un-capacitated network design problems that consider joint minimisation of networking and processing power consumption in the placement of resource intensive services such as visual and object recognition applications.
3. To evaluate the impact of service splitting on improving the total power consumption of the distributed processing approach, in capacitated and un-capacitated network design problems.

4. To evaluate the provisioning of resilience for the proposed architecture, in terms of energy consumption overheads.
5. To evaluate the impact of inter-service communication for synchronisation and cooperation. This provides an insight on whether service splitting among low-power fog and IoT devices is beneficial if processing overheads are accounted for.

1.4 Thesis Contributions

To knowledge, the thesis made the following contributions:

1. It developed four new MILP models that can be used in the design of energy efficient distributed processing architectures, for both the short term and long term network design problems.
2. It designed and optimised the placement of non-splittable services for a number of active IoT scenarios that capture different distributions of source nodes in the network topology.
3. It evaluated through MILP, the impact of allowing service splitting on improving the total power consumption for both the short and long term network design problems.
4. It examined through MILP optimisation, the resilience of server protection for the main scenarios of active IoT source nodes that were considered in the thesis.
5. Finally, a MILP model was developed to incorporate the concept of inter-service communication to study its impact on total power consumption.

1.5 Related Publications

The following list includes publications that resulted from the work presented in this thesis:

B. Yosuf, M. Musa, T. Elgorashi, A. Q. Lawey and J. M. H. Elmirghani, "Energy Efficient Service Distribution in Internet of Things," *2018 20th International Conference on Transparent Optical Networks (ICTON)*, Bucharest, 2018, pp. 1-4.

B. A. Yosuf, M. Musa, T. Elgorashi and J. M. H. Elmirghani, "Impact of Distributed Processing on Power Consumption for IoT Based Surveillance Applications," *2019 21st International Conference on Transparent Optical Networks (ICTON)*, Angers, France, 2019, pp. 1-5.

Barzan. Yosuf, M. Musa, T. E. H. El-Gorashi and J. M. H.Elmirghani,"Energy Efficient Distributed Processing for IoT", *submitted to IEEE Access.*

Barzan. Yosuf, M. Musa, T. E. H. El-Gorashi and J. M. H.Elmirghani," Resilient and Energy Efficient IoT Processing", *to be submitted to IEEE Access.*

1.6 Thesis Organisation

Following the introduction in Chapter 1, this thesis is organised as follows: Chapter 2 reviews the concept of the Internet of Things (IoT) and highlights the key enabling technologies. It also overviews the reference architecture of IoT, its elements, service types, applications and concludes by listing a number of challenges facing IoT.

Chapter 3 provides an overview of distributed processing paradigms and fog computing. It also provides a critical review of the related works which motivated this thesis.

In Chapter 4, the problem of energy efficiency of distributed processing is tackled given non-splittable processing services, for both capacitated and un-capacitated network design problems. A visual based object recognition application was considered and the workload characteristics were determined. In addition to multiple layers of fog, two types of data centres were examined: 1) general-purpose data centres, which are energy inefficient and 2) special-purpose data centres, that are highly energy efficient and specialised in particular forms of processing. In most cases, distributed processing produced significant savings however for very high workloads it was observed special-purpose data centres were more favourable as they introduced savings of up to 50% in power consumption.

Chapter 5 examines the influence of service splitting on improving the energy efficiency of the proposed distributed processing approach. It was observed that in the un-capacitated case, following optimisation for energy efficiency, service splits mostly occurred between the IoT and the CPE fog layers, for relatively low workload volumes however as the workload increased, the cloud

and metro fog interplay produced better results due to processing inefficiency of CPE fog servers.

Chapter 6 focuses on resiliency for the architecture considered in Chapter 4 and in Chapter 5. The studies examined a 1+1 server protection scheme through geographical node disjoint constraints. The results showed that the highest percentage power overhead occurred when protecting low workloads which are within the capacity of IoT devices due to the activation of optical network units (ONUs) to get to another IoT device. For relatively higher workloads, it was observed that all layers of the proposed architecture were utilised in the protection however for very high workloads, primary and backup servers were predominantly placed in the cloud DCs due to their processing efficiencies.

Chapter 7 examined the impact of inter-service communications as a result of device cooperation and synchronisation.

Chapter 8 summarises the contributions of this thesis and suggests possible directions for future work.

Chapter 2

Background Review of Internet of Things (IoT)

2.1 Introduction

The Internet of Things (IoT), is regarded as a novel paradigm that is expected to pave the way for a plethora of applications that contribute significantly to enhancing our daily lives, in domains such as security, agriculture and health care, to name a few [14], [15]. Researchers in the past two decades have projected that billions of everyday objects will be connected over the Internet [16]. Such objects will range from simple devices such as RFID tags, temperature sensors to smart devices such as mobile phones, vehicles, surveillance cameras, etc. [17]. Cisco, reported in 2011, that by the year 2020, the number of connected IoT objects will reach around 50 billion [18]. While on the other hand, a more recent investigation by Gartner quoted this figure to be around 26 billion devices by 2020 [19]. Regardless of the correctness of the two aforementioned predictions, the matter of the fact is that IoT objects will be several times more than the estimated 7.7 billion of the world's population [20]. These connected objects are all expected to have the ability to communicate with each other and cooperate in order to reach a common goal [3].

There are manifold definitions of the term IoT and they all imply the same concept of ubiquitous connectivity between the physical "things" anytime, from any place and for anyone [21]. Hence, it is certainly safe to suggest that connectivity from "any place" is virtually impossible without wireless capability

[22]. While most people regard IoT as large scale sensors, health and fitness monitors, self-driving cars, it stands for much more than that [23]. IoT will transform the modern world and reshape the industries within it. For example, smart meters will be used to enable better management of utilities across the national grid, sensors and actuators will allow automation a factory floor, city surveillance cameras will be used to help law enforcement agencies to prevent crimes before they happen through the aid of machine learning and pattern recognition algorithms [24]. The list of potential applications of the IoT is endless. It is not of any surprise that the IoT is included in the list of the top six “Disruptive Civil Technologies” by the US National Intelligence Council [17]. This is an indication of the important role the IoT will have in the near future and like the present Internet of today, it could contribute greatly to every domain of society. In 2012 alone, it was reported that IoT based application systems generated a revenue of $4.8 trillion [25]. It is still early days for the IoT and this figure could easily rise above and beyond expectations.

2.2 Reference IoT Architecture

Generally, in the literature [26],[27], a common high-level reference architecture for the IoT is proposed that comprises of several layers as depicted in Figure 2.1. These layers are briefly described below in a bottom-up fashion:

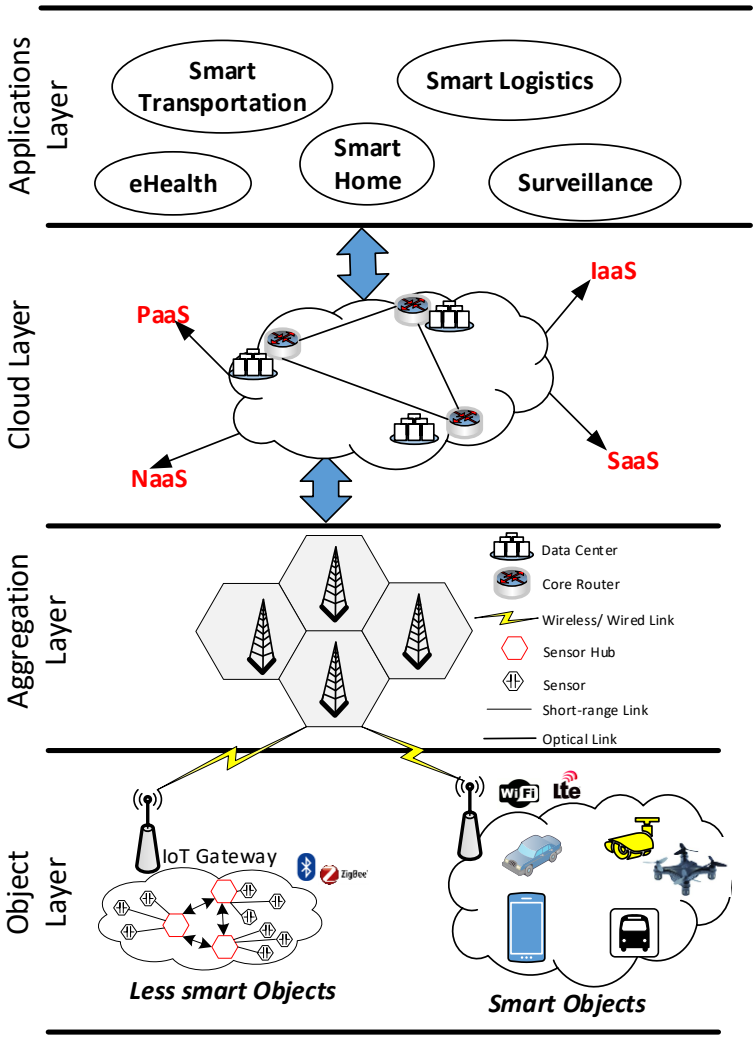


Figure 2.1 A High Level Reference Architecture of Internet of Things (IoT)

Object Layer: This is the lowest layer in the IoT architecture and is often called the perception layer. Its main purpose is to perceive the raw data from the environment. All the data collection and sensing are carried out in this layer. The IoT objects comprise of a variety of devices such as temperature sensors, smartphones, vehicles, drones, cameras etc. [5].

Aggregation Layer: This layer can also be called the network layer as it provides the networking infrastructure to securely forward the aggregated data from the objects to the cloud for processing. The transmission can be done

through wired or wireless mediums. Typically, communication technologies such as WiFi, Bluetooth, ZigBee, LTE, etc are used, depending on the type of the devices in the object layer and the intended application [28]. In most cases, an IoT gateway device is used to aggregate raw data from the resource constrained devices in the object layer (especially the less smart ones) [29].

Cloud Layer: This can also be called the middleware layer as it receives huge volumes of data from the network layer [30]. The main purpose of this layer is service management and data storage. It has an analytical centre to process the aggregated data and take automatic decisions based on the results of the analysis and feeds the output into the application layer [29]. This layer facilitates data access and storage through services in the cloud such as infrastructure-as-a-service (IaaS), Platform-as-a-service (PaaS), Software-as-a-service (Saas) and Network-as-a-service (NaaS).

Application Layer: This layer is at the top of the architecture as is used in the presentation of the final data to the final user [3]. It receives information from the cloud and in return provides management services for the application presenting that information[30]. Hence, the application layer presents the IoT data in the form of smart city, smart home, eHealth, smart transportation, surveillance etc [28].

The basic workflow of IoT depicted in Figure 2.2 can be described in a simplified version as follows [30]:

1) Objects begin by sensing and making identifications to communicate object-specific information. Such sensed information can be a temperature reading, orientation, motion, video or audio, etc.
2) Processing of received information by smart devices which leads to making informed decisions such as triggering an actuator or object identification within a video/image file.
3) Feeding back information on the current status of the system to the administrator.

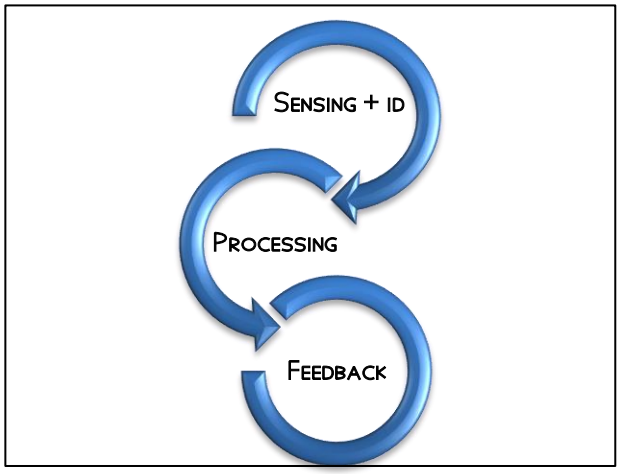


Figure 2.2 Basic Workflow in IoT

2.3 Enabling Technologies

Many key technology enablers are driving IoT into actualisation [3], some of which are described below:

1) **Wireless and Low-Power Communication:** For a long time, the "anytime, anywhere, any media" vision has been pushing forward the advances of communication technologies beyond boundaries. In this

regard, wireless technologies have played a major role as it was reported in 2010 that, the ratio between radios and people neared the 1:1 value [3]. In addition to this, the increase of the number of transistors on a single chip whilst simultaneously achieving reductions in size, weight, energy consumptions and monetary cost of the actual radio chip can push the aforementioned ratio above orders of magnitude [31]. Additionally, recent developments in lightweight protocols such as the IEEE 802.15.4 low-power wireless personal area network (6LoWPAN) has made it possible for the IP stack to connect a huge number of tiny and battery constrained sensor devices to the Internet [32].

2) **Smart Devices:** with the recent technological advancement in embedded systems and wireless communication, it is no longer the case that the IoT should include only simple sensors and actuators performing primitive tasks. As a result of the advances, various kinds of objects emerge as powerful devices that can sense, process and communicate over the network [33]. Such devices are expected to have capabilities of stationary servers residing in the cloud data centre [34]. Additionally, computational capability on portable devices have followed the Moor's law for the past two decades and this is anticipated to continue the same way in the upcoming periods [34]. In the literature, the key functionalities of smart devices are categorised into three main groups namely, context awareness, device connectivity and autonomy [35].

3) **Data and Energy Storage:** Taking the most recent years as example, in 2012 data storage capabilities on a smart phone/tablet was about

32GB and as of 2019, Apple's latest iPhone 11 is equipped with 256GB of internal storage [36]. It is of no doubt that memory and storage technologies will only get better in the future. This coupled with the invention of lithium batteries, it is only a matter of time before IoT objects penetrate every sector of our modern society. For instance, an RFID transmitter can operate up to one year on a single lithium coin cell battery with 240-mAh [37].

4) **Cloud & Virtualisation:** Virtualisation is the key technology of the cloud. It allows for the creation of various logical infrastructures on the same physical hardware [38]. These logical infrastructures may comprise of computing and networking resources. The virtualisation of clouds enables small to large organisations to lease powerful resources on a pay-as-you-go basis such as Software-as-a-Service (SaaS), Platform-as-a-Service(PaaS), and Infrastructure-as-a-Service (IaaS) [28]. This implies that IoT system novelists do not have to own expensive infrastructure in order to be able to run IoT analytics, but instead, they only have to bear the cost of usage of the service(s) [38]. Moreover, as well as compute resources for data analytics, virtualization has also enabled IoT network resources to become decoupled from traditional proprietary hardware. Hence, allowing for small IoT developers to request multiple heterogeneous virtual IoT networks on a pay as you go basis, which is mostly referred to as Virtual Sensor Networks (VSNs) [39]. Thus, both compute and network resources have become fully flexible and can dynamically be shared to achieve rapid development of new IoT services [11], [40].

5) **Next Generation Access Networks:** Conventionally, the last drop to the end-devices is provided by fixed-line access technologies such as copper based xDSL WiFi modems as well as wireless mobile technologies such as 3G/LTE etc. Given, the enormous expected growth in the number of connected IoT nodes and the need to access the distant cloud for data processing, the aforementioned access technologies could not cope with the front/backhauling demand of the IoT due to lack of bandwidth as well as their energy inefficiencies [41]. Thus, a number of important advancements have been made to tackle the above issues such as a multitude of heterogeneous access networks integrated into a single platform (5G) to provide better and seamless data exchange with the cloud. A number of efficient schemes allow for the integration of a wireless front to offer ubiquitous services for mobile/fixed nodes and fibre links to provision for the backhauling which is not possible with the wireless infrastructure on its own [42]. In the access part of the network, Passive Optical Networks (PONs) have been hailed as the most attractive solution due to their high bandwidth, low cost and point-to-multipoint architecture [17]. This technology offers great bandwidth in the uplink and downlink and has the potential to allow for network convergence. Since the required data rates over wireless/mobile networks are achieved through decreased cell distances, a large number of cells in urban areas will be deployed, this particularly makes PON an attractive access networking solution due to its point-to-multipoint architecture [43].

2.4 IoT Service Types

The IoT has embedded itself in many aspects of our daily lives. In this section, some of the applications and services of IoT are reviewed. The applications of the IoT are so vastly endless, we can merely scratch the surface. In the literature, works such as [44], [45] and [46] have grouped the IoT applications into four main classes that can be used as umbrella terms to aid our understanding of this imminent technology. As depicted in Figure 2.3, these classes consist of 1) identity-related services, 2) information aggregation services, 3) collaborative-aware services and 4) ubiquitous services.

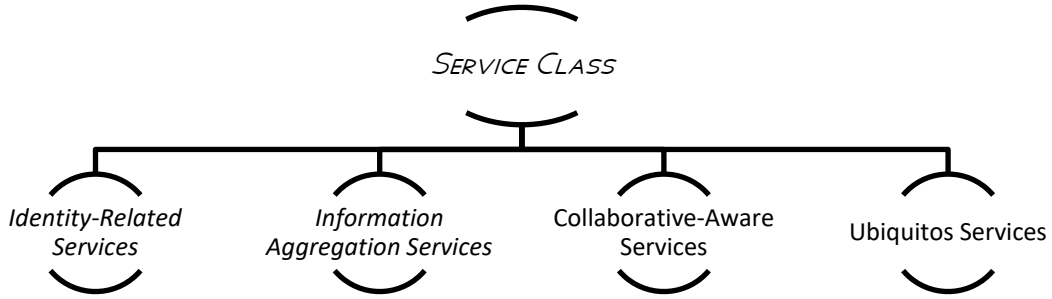


Figure 2.3 Categorised groups of IoT services [44].

The most basic and yet important service out of the four is the identity-related services. The reason behind this is that any application that requires digitalising the properties of the physical “things”, must first have the means to identify those “things” [44]. Whilst, on the other hand, Information Aggregation Services are reliant upon the first category as such services refer to the process of acquiring data from various sensors, processing this data then transmitting the obtained knowledge via IoT to the application in question. In the aggregation process, services may use different communication

channels to acquire data from different types of sensors. For instance, RFID tags could be used in an application to identify objects, whilst simultaneously using a ZigBee network to aggregate sensor data, as illustrated in Figure 2.1. All of these functionalities are incorporated into a device mostly referred to as “IoT Gateway” [47]. An IoT gateway can be a device or software that acts as a bridge between the objects and the cloud since all the data moving to the cloud has to pass through the gateway. Often most, the data passing through the gateway is mined in order to extract knowledge before it reaches the cloud. This extracted knowledge is then acted upon by Collaborative-Aware Services, which require “thing-to-thing” as well as “thing-to-person” communication. Thus, provisioning for collaborative aware services, the IoT network infrastructure will require extensive reliability and a significant increase in speed [46]. Ubiquitous Services, however, are the essence of IoT. Such services are not only collaborative, but aim to provide collaborative services “anytime”, “anywhere”, and for “anyone” [48].

2.5 IoT Applications

Having reviewed the categories of the IoT services and applications in the previous subsection, the current subsection will provide examples applications that fall into each type service class since this is a useful way of providing a basic framework to build an application upon a particular type of class [46]. With the potentialities offered by the IoT, it is not possible to list every possible application since only a small number is currently deployed in our society [3].

Logistics: This is one of the common examples of identity related services. IoT devices such as RFID tags are used to track and trace almost every link in the supply chain ranging from purchasing of raw materials, products to after sales and inventory management [49]. RFID tags used to be known as the digital version of barcodes, however, they offer much more in that they can be utilised to track items in realtime which provides useful information about their current status accurately and timely [21]. Thus, various enterprises can better manage their inventory and resources by being able to plan ahead of time, thus responding promptly and accurately to the dramatic nature of markets in a short time [3].

Smart Transportation Systems: Smart motorways and roads are already deployed throughout most cities of the UK [50]**.** The application comprises of thousands of different types of sensors, cameras and digital displays to monitor and respond to the dynamic change in traffic on the motorway [51]. The National Roads Telecommunications Service (NRTS) project was implemented as part of the UK's smart motorway project and its second phase contract has already been handed over to telnet Technology Services in 2017 [52]. This project works on providing the backbone network that enables the 7 regional control centres throughout the UK to connect to up 30,000 technology assets such as CCTV cameras, message signs, weather condition sensor networks and motorway incident detection and automatic signalling (MIDAS) loops installed every 500 meters to detect the current state of traffic due to an incident [53]. This type of application exhibits elements spanning across the identity-related services and information aggregation services. In fact, every potential IoT application will at least incorporate some parts of identity-related services due to the fact that every object within the network needs a digital

identity. As for the second class, the devices in the network are only used to aggregate data about their ambient environment and forward them towards the regional control centres for decision making purposes. Figure 2.4 is an illustrative depiction of the different types of device networks used for information aggregation for the regional control centres.

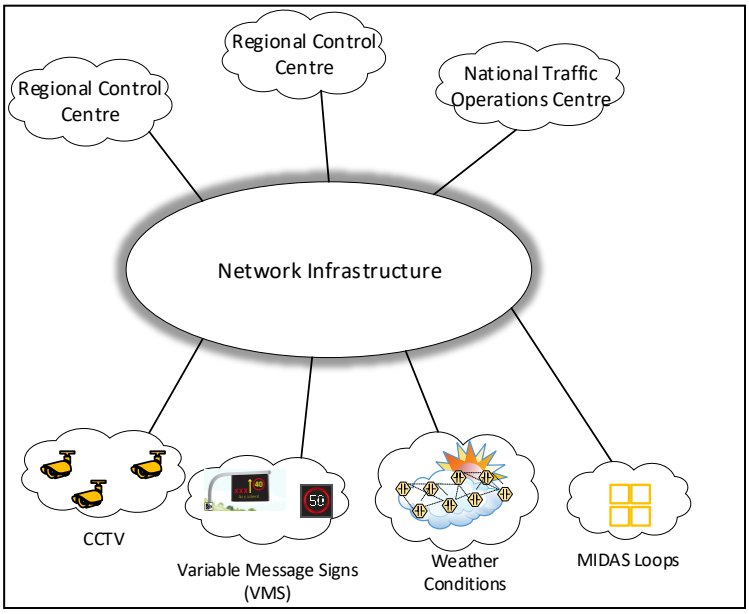


Figure 2.4 A high-level depiction of the applications and Services offered by UK's smart motorway systems.

Smart Surveillance: In a smart city, large numbers of intelligent video surveillance cameras will be distributed along roads or strategic intersections of a busy road to provide a strong sense of assurance to the general public [54]. These intelligent cameras will be exploited to run object recognition algorithms and machine learning tools to alert law enforcement agencies to take action proactively and prevent potential crimes [55]. The different cameras are also used in some applications to cooperate with one another to identify and track vehicles in motion [3]. This overall makes these intelligent devices somewhat autonomous and collaborative in the sense that there is some sense of collaboration between the objects in question and there is autonomous decision makings taking place [46]. In a nutshell, a service is

classed as collaborative-aware if the data collected by the objects are acted upon at the source node and decisions for performing certain tasks are taken. One of the prerequisites of creating a collaborative-aware service is having computational power onboard IoT objects, hence these devices can no longer be used for sensing purposes only.

2.6 Challenges

In the near future, the IoT will revolutionise the shape of today's Internet which will lead to endless economics and societal benefits, but at the same time, it is faced with many key challenges that need serious considerations [27], [56]. Some of these challenges are briefly described as follows [30]:

- **Object ID Management:** As is apparent, the IoT ecosystem will consist of billions of heterogeneous objects, that are going to be used to provide innovative services. A network of objects at such scale will require these objects to have unique IDs over the Internet, thus, this calls for proper object ID management scheme, that is able to dynamically assign and manage unique identity for all the objects. Traditionally, the domain name system (DNS) has been used over the years in current networks for such purposes, however, such system cannot adapt to the IoT environment as many of the "things" or objects will be mobile and resource constrained [11].
- **Interoperability and Standardisation**: The general prerequisite of IoT systems is to support openness and interoperability since different IoT objects need to connect through different interfaces to provide the required services [57]. Many of such objects are going to be proprietary

hardware, manufactured by various vendors, standardisation of the IoT is vital to provide better interoperability for all the “things” [58].

- **Security, Privacy & Trust**: The IoT presents serious security related challenges that are reported in the IERC 2010 Strategic and Innovation Roadmap. While each domain, security, privacy and trust is faced with unique challenges in the IoT, they all share a number of non-functional requirements such as 1) light-weight solutions to support constrained objects, and 2) scalable to billions of devices. Moreover, any solution must address object heterogeneity and diversity as well as being seamlessly integrated into real-world [1]. Below is a brief description of each domain:
    - **Trust**: Since IoT-based systems and application will scale over multiple administrative domains, it is bound to involve multiple stakeholders and ownerships. Hence, there needs to be a proper trust framework that enables the users of the IoT system to have full confidence that the services being provided can be relied upon.
    - **Security**: The IoT will comprise of a large number of physical objects that are potentially spread over in a large geographical region. Similar to DoS/DDOS attacks on the current Internet, the IoT will not be exempt also from such threats. It is necessary to have specific techniques and mechanisms to ensure that IoT services cannot be disrupted or undermined [1].

Privacy: The IoT will use different forms of identification technologies such as RFID tags, that may be associated with objects, from which people’s location can be inferred, as an example. Thus, it is very important to have the right

mechanisms in place that prevents the inference on personal information and allows IoT users who wish to keep their details anonymous. Also, another measure to protect personal information is to keep data as local as possible making use of decentralised processing and key management [59].

- **Network Channel Bandwidth:** The vast number of connected "things" will create data at an exponential rate [60]. A connected object such as a vehicle can generate tens of MB's of data per second. Such data will comprise of 1) the vehicle's mobility and its routes, 2) vehicles operating conditions, 3) the vehicle's ambient environment such as road and weather conditions, 4) videos recorded by the vehicle safety camera [5]. An autonomous vehicle is reported to generate even more data, at around 1 GB per second [61]. Transporting all the data over the network to the cloud will demand huge network bandwidths. The wireless medium used to transmit IoT data should be able to handle such scale of sensors without any data loss due to network congestion [30], [25].

- **Energy Efficiency of IoT (or Green IoT):** Due to the formidable volume of data generated by billions of objects, the intervention of cloud becomes a necessity as processing will be required to extract knowledge from all that data. In order for the cloud to process, it is required to be running around the clock which leads to enormous consumption of power and this subsequently leads to higher CO_2 and eventually takes a deep toll on the environment [25]. In addition to the environmental impact, energy consumption is directly linked with

network operators' operational expenditure as well as social and political issues [62]. Thus, it is important to note that in this regard, energy harvesting and low-power ICs are central in the development of the IoT [22].

2.7 Summary

Internet of Things (IoT) is regarded as an emerging paradigm and its actualisation is imminent in the near future. This chapter has briefly introduced the concept of IoT and its elements that are used to define and shape it. Although a standard architecture is yet to be agreed upon by all the stakeholders of this revolutionary technology, a reference architecture which is widely used in the literature was reviewed by depicting a high-level network architecture with a detailed description of all the layers involved in the IoT stack. This is followed by the most important technologies that are seen as major enabling factors in realising the IoT, types of service classes and example applications within each class and the chapter was concluded by identifying the key challenges that are reported to impede the progress of the IoT.

The next chapter surveys various distributed processing paradigms as well as the related work on energy efficient distributed solutions in the context of IoT.

Chapter 3

Distributed Processing and Related Works

3.1 Introduction

Generally, distributed processing refers to a decentralised system in which computational tasks are subdivided between multiple networked devices, and these devices communicate with one another through the network to achieve the original goal of the application. One of the earliest known versions of distributed processing was introduced back in the 1970s which comprised of a local area network (LAN) that interconnected serval computers and allowed multiple applications to communicate among themselves and develop a collective solution for a computational problem [63]. However, since then, distributed processing has evolved into various new paradigms, namely fog computing, edge computing, mobile cloud computing, cloudlets, clouds, etc [64]. These paradigms all have at least one thing in common which is processing end-devices data over a given communication network, as close as possible to the end device. However, they differ in terms of the scale of hardware deployment and their level of proximity to the IoT end devices [65]. For instance, clouds are accessed via the core network while fog computing, edge computing, mobile cloud computing and cloudlets could be one hop away from the end devices. Also, cloud data centres house thousands of powerful servers and compared to the other paradigms their resources are virtually unlimited [12].

Generally, cloud computing is known as a centralised solution in which end users share a large, centrally managed pool of resources which are offered to the end-users on a pay as you go basis. Such resources are mainly classed into three groups, namely Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [66]. With IaaS, end-users are offered virtualised storage, computation and network whilst PaaS are services that provide the needed environment for application development and SaaS combines IaaS and PaaS to allow end-users to outsource all their computational needs from the cloud [65].

On the other hand, fog computing is classed as a decentralised computing paradigm due to its distributed nature and the number of nodes is greater than the centralised cloud by several orders of magnitude [67], [68]. Fog computing aims to extend the functionalities of the centralised cloud closer and even onto the end-devices themselves, hence the OpenFog Consortium as the main promoter of fog computing defines the fog architectures as a “cloud closer to the ground” [69]. As shown in Figure 3.1, fog computing can be represented through a hierarchically, usually in three layers. The bottom-most layer is where all the IoT end-devices are located and these are expected to have limited computation, storage and networking capacities, whilst the higher layers are expected to contain the more powerful devices. Any device that is equipped with communication, computational and storage resources can act as a fog node [70]. At the edge of the network, large numbers of potential fog nodes exist, collectively. They can offer enormous amounts of computing power as they are spread across millions of devices which include routers, switches, gateways, smartphones, surveillance cameras etc. [66].

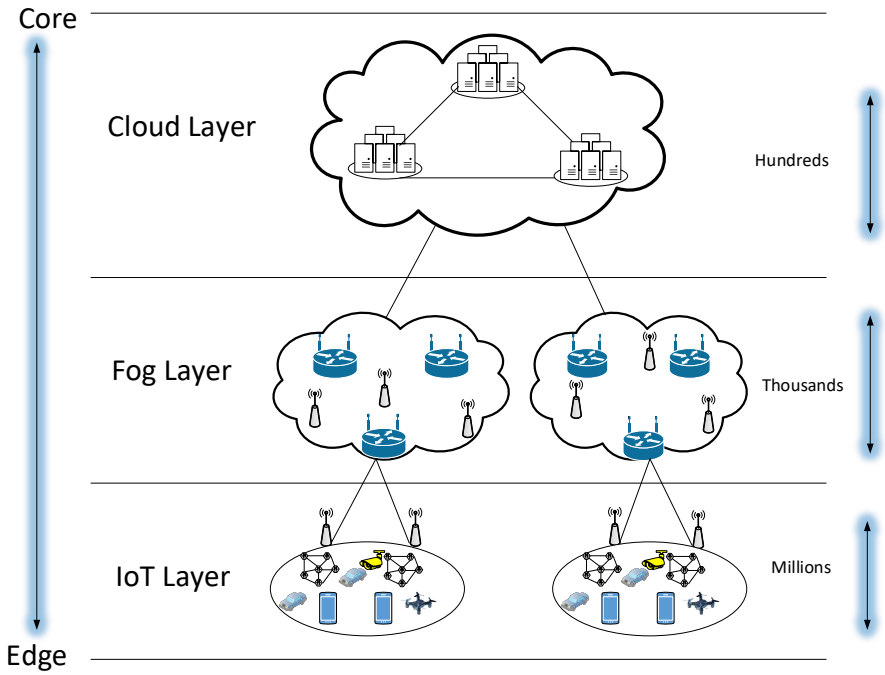


Figure 3.1 A High-Level Architecture of Fog Computing Supported by Cloud Resources

It must be noted that fog cannot be implemented solely on its own, but rather, the cloud must be used to complement the limitations of the fog such as processing compute-intensive application components, whilst on the other hand, the less compute-intensive components can be processed on the fog nodes [2], [11].

3.2 Related Work

The work proposed in this thesis reduces the energy consumption of distributed processing in the context of IoT and fog computing through the optimum placement of application services, from an end to end perspective. This subsection summarises previous research work related to energy consumption minimisation in the context of IoT and related paradigms, through various schemes such as resource allocation and architectural design and planning.

Generally, in a fog architecture, a large number of devices exist at the edge of the network, which collectively provides enormous amounts of computational power, that, if used, may help in curb the unnecessary data exchange between the IoT and the centralized cloud [71]. These devices are heterogeneous in terms of resources. This poses a number of challenges in the optimum design of architectures, protocols and hardware of future IoT based networks. Hence, proper resource management and network design solutions are needed [72]. These solutions should take into account important dimensions such as but not limited to energy efficiency, due to its impact on the environment [73], resilience, due to service criticality [2], [73], [74] and end-device cooperation, due to traffic bifurcation which leads to inter-service communication [75], [76]. Thus, fog solutions have been proposed to improve the aforementioned performance metrics through various approaches such as resource allocation [10], [14], [16], [25], [77]–[80] and architectural design and planning [29],[71], [81]–[84]. The reader is referred to the works in [73] and [64], for architectural design imperatives of fog networks and a detailed taxonomy of fog based solutions, respectively.

The focus in the literature has shifted towards making the whole IoT infrastructure more energy efficient [29] as opposed to optimizing only individual layers namely the device layer, access layer or the cloud. The works in [83] and [84] proposed the use of PONs to extend cloud and fog services closer to the user premises, respectively. Optical based networks are expected to become increasingly important to support edge and fog computing in the next decades. Although no particular algorithmic or optimization model was proposed, however, detailed discussions were provided on how the architecture in question can improve QoS and how different distributed fog resources located in the user premises can efficiently be managed.

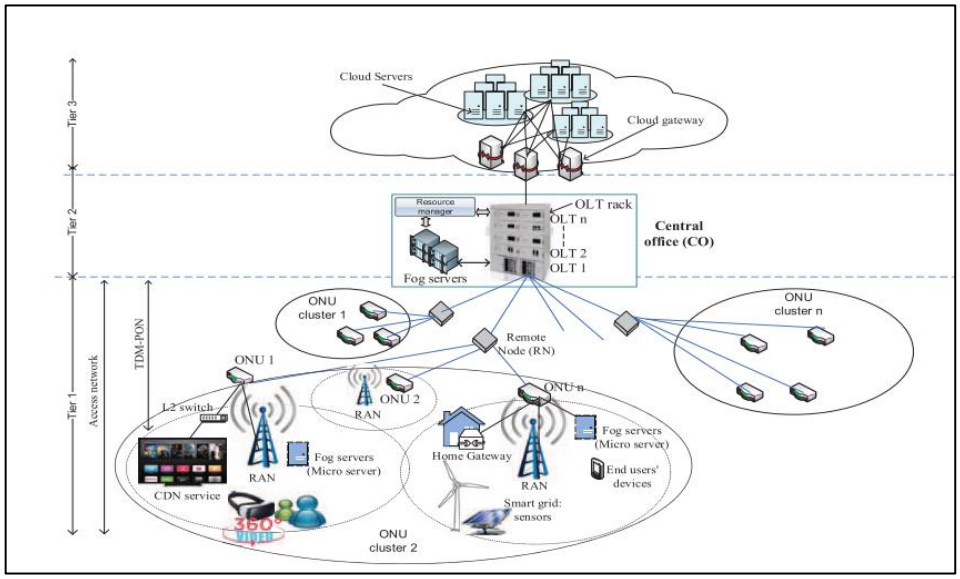


Figure 3.2 The Proposed Fog Architecture over PON by the Authors of [84].

The authors of [29] proposed an energy efficient IoT architecture in which sensors' sleep intervals are predicted based on their remaining battery level and as a result resources of the cloud can be better utilized by re-provisioning them when the sensory nodes are in sleep mode. The main contribution of the work is centred around developing a mechanism to predict

the sleep intervals of sensor nodes based upon certain sensor variables such as battery level and previous usage history.

The work in [25] mathematically models the entire fog network from the end terminals (TNs) to the cloud data centres located in the core network. The TN nodes sense data and transmit the same to the fog tiers, either to be processed by fog nodes or to be forwarded to the cloud for further analysis. The performance of the fog approach in provisioning for IoT applications is investigated by considering several dimensions such as power consumption, CO2 emissions and service latencies in the fog network compared to the baseline cloud system. Their results indicate that the fog computing approach is only beneficial when there is a high number of latency-sensitive applications. Although fog computing was comprehensively studied, the authors made no mention of the practical networking or processing hardware that was used in obtaining their results. In another work, the authors of [71] compare the efficiencies of highly distributed edge devices called Nano data centres that can host and distribute user contents in a P2P fashion. These edge servers are comprised of Raspberry Pi's that are low power single board computers. The authors investigate the system performance through time-based and a flow-based power consumption model. For devices that are highly shared by many users and services, the authors adopt a flow-based model whilst a time-based model is used for equipment that is close to end-users.

The work of [82] proposes a framework for cloudlet based network design and planning. The focus of the work is primarily centred around designing a network based on TDM-PON to optimize the network

infrastructure cost whilst meeting latency constraints only. The problem is formulated as a Mixed Integer Non Linear Programming (MINLP) model which is utilized to identify efficient cloudlet placement locations and optimal assignment of ONUs to cloudlets. The feasibility of the proposed model is evaluated against urban, suburban and rural scenarios, which guide the installation and maintenance costs. In another work [80], a generic fibre-wireless architecture is proposed which supports the coexistence of the centralized cloud and distributed mobile edge computing (MEC) for IoT connectivity. A distributed game-theoretic algorithm is developed to support collaborative computational offloading between the cloud and MEC. Numerical results show very low energy consumption is achieved compared to the baseline which is the optimal case that cannot be realized in practice, hence the distributed approach is used to reduce complexities. The authors of [81] put forth a capacity planning framework that improves the resource utilization of a hierarchical edge cloud network whilst simultaneously meeting QoS requirements in terms of response delay. They do this, by taking advantage of diverse demands for CPU, GPU and network resources.

The authors of [14] formulate the service distribution problem in an IoT-Cloud architecture using a linear program whose solution results in the optimum placement of IoT service functions and the routing of network flows across a multi-layer architecture consisting of devices, access and cloud layers. The total energy consumption is minimized whilst meeting the end-user latency demands. In another work [79], the service allocation problem is formulated as an integer programming optimization, whose objective function is to minimize the total latency experienced by IoT services, subject to capacity constraints at the various layers of the proposed fog architecture. The

IoT service requests are considered to be generic, ranging between 10 – 50 homogenous requests. The delay is minimized by placing the less demanding services as close as possible to the IoT devices whilst the medium and high demanding services are placed higher up the fog network. In their work, IoT devices have been excluded from hosting any type of data processing.

Similarly, the authors in [78] propose a generic algorithmic for the placement of IoT services in a fog-cloud framework. The IoT services are considered as multiple modules that are collectively used to deliver a full application. A specific algorithm is used to efficiently deploy application modules dynamically across the fog-cloud infrastructure close to the source terminals in the fog layer. The performance of the proposed solution is addressed through evaluation of latency and efficient resource utilization and it is claimed that it can be extended to include further design dimensions. In [77], an Integer Linear Program (ILP) is proposed to model the problem of resource provisioning from the perspective of service providers, in the context of the heterogeneous Internet of Things, where the objective function is to minimize the total monetary costs subject to capacity and latency budgets. The heterogeneity of IoT is modelled through unique profiling of applications and as such 4 different types of applications are considered. The topology considered comprises of a Metropolitan Area Network (MAN) and consists of two hierarchical levels of interconnected rings. The results indicated that the total operational cost is directly impacted by the application's computational complexity, compression factor, and latency budget, coupled with proportions of local traffic versus global traffic. The authors in [10] put forth a convex optimization model that addresses the delay-power trade-off in a cloud-fog architecture which consists of four subsystems. The work demonstrated that

compromising modestly on computational resources in order to save communication bandwidth and reduce transmission latency, fog computing platforms can significantly complement the performance of cloud computing. The proposed work has not considered the impact of local computation using the devices in the IoT layer.

The authors of [16], unlike the previous aforementioned works, model the IoT service placement in a practical testbed using an ILP formulation by considering several objective functions that address service latency, service migrations and energy efficiency. The optimization model is executed iteratively to allow for the retention of the objective values of previously executed models, thus, the feasibility region continuously decreases since iterations must satisfy previous results. The approach is generic and can be adapted to other resource placement problems. Their results show that for real-time services, latency becomes important and thus services are processed on the nearest fog, while the latency tolerant services can be offloaded to the distant cloud as energy consumption becomes the priority.

It is observed that each of the approaches proposed in all of the aforementioned studies does not consider fog solutions that offer network designers' insight into energy efficiency in short-term (capacitated) and long-term (un-capacitated) optical based fog networks. Moreover, our previous works considered energy efficient solutions in cloud and core networks, IoT and mobile networks using MILP techniques considering a variety of scenarios including big data processing in core networks [85], [86], design of energy efficient optical architectures [87]–[89], and data centres [90], content distribution [91] and caching [92], network coding [93], NFV and big data in

mobile networks [94], [95] and virtualization and process embedding in IoT based networks [96], [97].

In contrast, the work in this thesis aims first to model the entire IoT infrastructure in which all layers of the networking domains such as end devices, access, metro and core are taken into account from the moment an IoT service is launched until it is hosted on the ultimate destination which is the cloud DC, accessed via the core network. A Passive Optical Network (PON) has been proposed to support the fog infrastructure in the access domain as it is increasingly utilized due to its suitability for data intensive applications as they provide high bit rates, relatively low cost and high scalability [12]. An Ethernet based network is considered in the metro to aggregate traffic from the PON towards the cloud DCs in the core domain. An IP/WDM core network is considered to provide access to cloud DCs, in which a large number of servers are interconnected via a LAN network.

One of our main contributions in this thesis is the inclusion of the optical core network to provide access to the Cloud DC which is currently not supported by any of the aforementioned studies. Furthermore, several design characteristics that affect the power consumption of the fog approach are considered, including 1) granular power consumption profile of networking and processing devices; 2) Power Usage Effectiveness (PUE) to account for cooling [98] requirement in higher capacity devices found in the access, metro, core and cloud layers; 3) service splitting and the prospect of improved server packing in the fog layers; 4) deployment of special purpose DCs (SP-DCs) in the core network in addition to its general purpose DC (GP-DC) counterpart;

and 5) inter-service processing overhead to account for synchronization between sub-services.

3.3 Summary

This chapter has provided a brief overview of distributed processing paradigms, which have evolved since their inception into the modern day fog computing, specifically within the scope of IoT. Based on the literature, a high-level architecture of fog computing networks has been provided as well the type of devices that can act as fog nodes. Also, the merits of the fog IoT type services were compared to that of the conventional centralised cloud. Moreover, the cloud was briefly introduced along with the type of services it offers.

The chapter was concluded by the second subsection which consisted of an overview of the works in the literature pertaining to this thesis. First, the works that inspired the basis for the thesis were reviewed and the merits of the choices made were also discussed. Then the remaining parts of the related works were discussed. These were mainly within the scope of resource management and provisioning in the context of fog computing for IoT services. Chapter 4 introduces the first new IoT-Fog architecture in this thesis and optimises it for energy efficiency.

Chapter 4 Energy Efficient Distributed Processing with Non-Splittable IoT Services

4.1 Introduction

Due to the intensive resource requirements associated with visual processing applications, IoT based surveillance is a service with high energy consumption driven by the data rates and CPU it requires to deliver the final application. Processing all or parts of the collected data as close as to the end-device (e.g. source of the video) is seen as an effective strategy to reduce the total power consumption of such services. This chapter evaluates the impact of distributed processing on the reduction of the total power consumption, in an end-to-end IoT infrastructure based on the concept of fog computing. A Mixed Integer Linear Programming (MILP) model is developed to optimise the placement of service functions for a range of homogenous demands that are comprised of requests for networking and processing resources. In all the previous proposals, the problem of distributed processing in a practical network has not been considered yet in great detail. On the contrary, in this chapter, a very detailed and accurate approach towards energy efficient distributed processing from the end devices all the way to the cloud data centres attached to the core network elements is considered.

4.2 Case Study

As mentioned in previous chapters, IoT type devices do not only comprise of simple sensors and actuators but in addition, there will be a class of highly intelligent devices that will be expected to perform a substantial amount of

processing due to their abundant computational resources. In this thesis, we consider a visual analysis application for surveillance purposes. In the next generation smart cities, video surveillance is considered an important element since distributed cameras on a stretch of a busy road or a shopping centre could bring city security onto a higher level, thus providing the public with a strong sense of assurance [54]. China has already implemented a system called Skynet, which consists of a massive network of smart CCTV cameras that have AI incorporated into them and it is claimed to cover the whole of Beijing [99]. Although the project is faced with criticism from the Chinese public due to privacy concerns, it was demonstrated to a BBC journalist from a police control room how the system of networked cameras helped to catch the journalist within 7 minutes after an “escape” was staged [100].

Thus, we consider video surveillance applications as it has become clear that their widespread deployment is imminent in the future. The sheer data rates associated with video data being collected by a large scale network of intelligent cameras makes it virtually impractical to transport all of that data to the cloud for processing to obtain insights. In a news article published in 2013 by The Telegraph, it is reported that there is one surveillance camera for every 11 people in the UK [101]. Thus, we are motivated to investigate visual processing applications through the concept of fog computing to help reduce the implications of unnecessary data exchange with the centralised cloud by hosting parts of the service requests in the distributed layers of the fog framework.

4.3 The Proposed Distributed Processing Architecture

We begin by describing each layer of the proposed architecture depicted in Figure 4.1. It comprises of four main layers, namely the IoT Devices, Access Fog (AF), Metro Fog (MF) and the Cloud DC (DC). The following subsections will provide further details on the aforementioned layers.

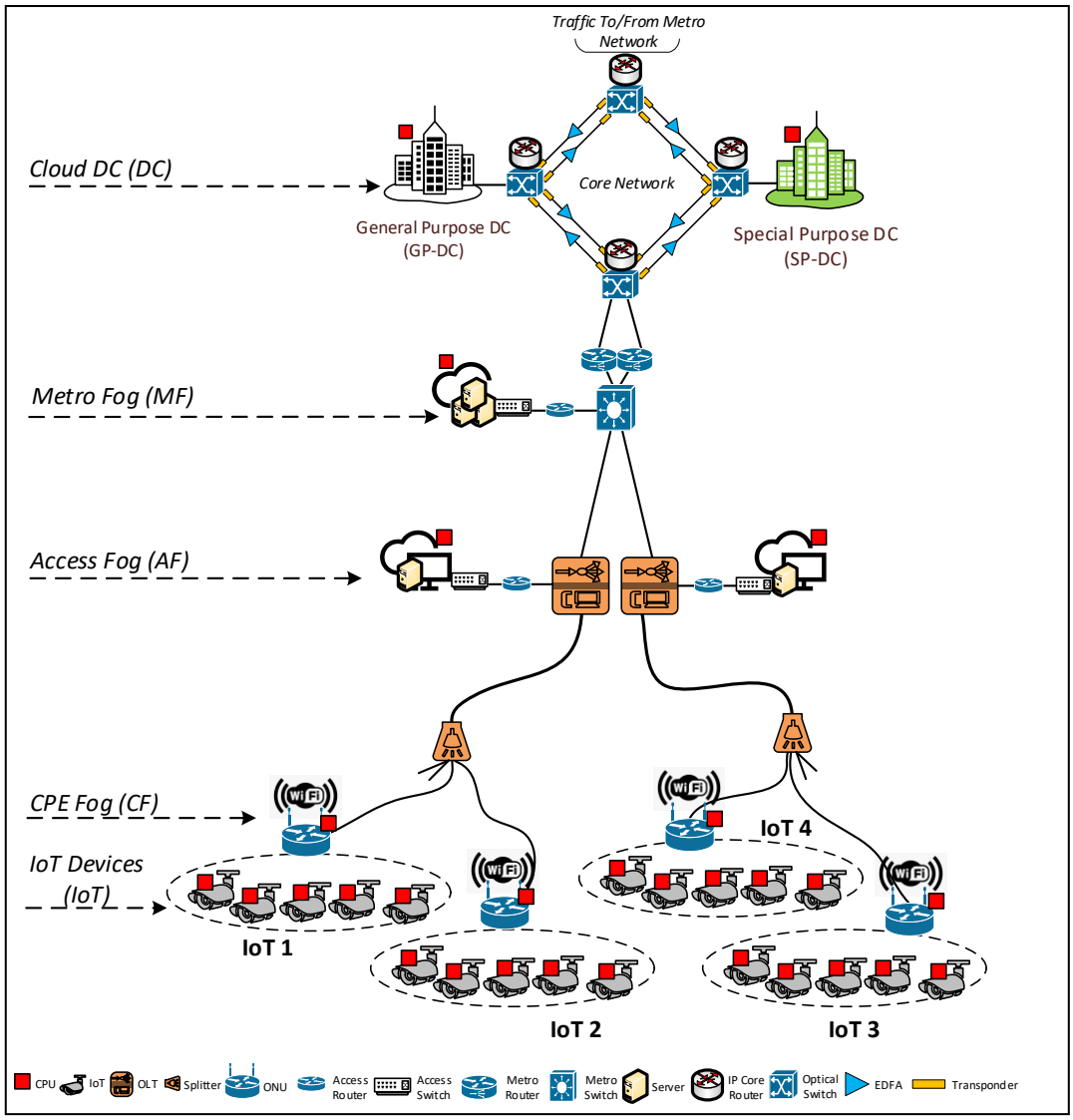


Figure 4.1: Proposed PON-based IoT Architecture Supported by Fog and Cloud Computing.

IoT Devices (IoT)

The bottom-most layer of the proposed architecture comprises of the IoT devices. These devices are smart, wireless nodes that are used to collect video data and transmit the same via the connected access point (ONU) to the next layer for processing and analysis, if local resources are insufficient. A WiFi link is considered between the devices and the ONU access points.

CPE Fog (CF)

The Customer Premises Equipment (CPE) Fog (CF) domain consists of a Passive Optical Network (PON), in which several clusters of ONU devices share a single fibre link to connect to the Optical Line Terminal (OLT) at the local exchange, via a passive optical splitter. The split ratio is commonly 1:32 or even 1:64, however, this depends solely on the planned network demand. Typically, PONs are connected in a star topology, with the link from the OLT being the root and ONUs being the leaves. This architecture is also known as point-to-multipoint (P2MP) [102]. PON is considered as one of the key access network technologies as it brings along several benefits including high scalability, abundant bandwidth, cost-effective services, and high energy efficiency compared to other access technologies [84]. Devices in this layer are predominately stationary and their processing capabilities are usually higher than those found in the IoT layer [103]. A typical PON distribution distance is usually up to 20km -60 km from the local exchange/ central office (CO).

Among the different flavours of PON (FDM-PON, OFDM-PON, Hybrid-PON), Time Division Multiplexing (TDM-PON) is considered a mainstream deployment across many regions of the world. The downstream bit rate in a

PON ranges from 1.244 to 2.488 Gbps, whilst in the upstream direction it ranges between 155 Mbps to 2.488 Gbps [104], [105]. Small organisations or even end-users can deploy their own fog infrastructures at locations such as APs, routers, gateways and etc.

Access Fog (AF)

The third layer is still part of the PON domain, however, it differs in terms of functionality and processing capability. Here, OLT nodes are placed at the local exchange point and they are used to aggregate service requests from all of the connected ONUs. In a typical scenario, a number of high-end servers are used to form a Fog collocated with the OLT [27], [48]. Thus, a substantial amount of service demands aggregated from the ONUs can be hosted and processed on the fog connected to the OLT Ethernet input. However, it still faces limitations and resource-intensive services are relayed to the next layer for processing.

Metro Fog (MF)

The metro network consists of a high-capacity Ethernet switch and a couple of edge routers that act as a gateway to the cloud data centres via the core network. The computational resources available to the metro fog is substantially higher in comparison to the lower fog due to the number of users and services it supports, however it still is incomparable to the cloud DC [106].

Cloud DC (DC)

The cloud layer comprises of a set of data centres that are accessed via the core network. Two types of data centres are considered: 1) a general purpose data centre (GP-DC), and 2) a special purpose data centre (SP-DC). Both data centres are a single hop away from the aggregation core router. As depicted in Figure 4.2, the local area network (LAN) elements inside both data centres consist of an edge router and a set of high speed switches to interconnect thousands of servers.

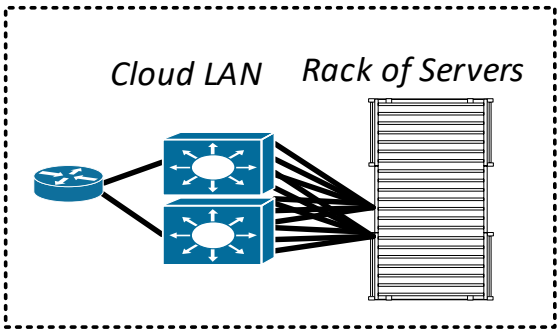


Figure 4.2 Network elements inside a cloud DC.

Core Network

The core network uses IP/WDM technology and it consists of two layers, the IP layer and the optical layer. In the IP layer, an IP core router is deployed at each node to aggregate network traffic from the metro routers. The optical layer is used to interconnect the IP core routers through optical switches and IP/WDM technologies such as EDFAs, transponders and regenerators.

Special Purpose Data Centres (SP-DCs)

Motivated by the sheer computational power of Graphical Processing Units (GPUs) as well as the breakthrough performances in terms of power consumption efficiencies for visual based deep learning algorithms, it is of

interest to investigate the implications of deploying such powerful servers that are dedicated and highly optimised to perform a specialised task, in the cloud data centre. We refer to such data centre as special purpose data centre because it only performs a specific service i.e. visual processing. On the contrary, the general purpose data centre (GP-DC) is designed to execute a range of generic services, hence, not as power efficient as the SP-DC. Nvidia, which is a leading manufacturer, has reported GPUs to be at least 10 times more efficient than CPUs.

The latest NVidia GPU (Tensor Cores T4) performance was benchmarked against a high-end CPU on the ResNet-50[1]. It is reported to be at least 27X more efficient than the CPU, using just 75 watts (W), making it an ideal solution for scale-out servers at the core or even the edge of the network. The ResNet-50 is a convolutional neural network that is trained on more than a million images and has the capability of classifying objects into 1000 categories[2].

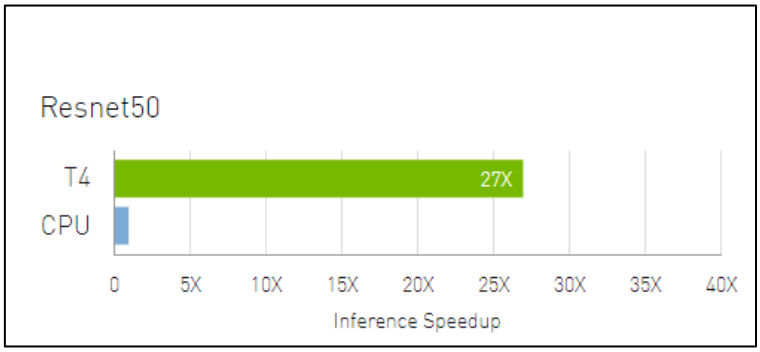


Figure 4.3 NVidia's Tensor T4 GPU Performance versus CPU.

[1] https://www.nvidia.com/en-gb/data-center/tensorcore/

[2] https://uk.mathworks.com/help/deeplearning/ref/resnet50.html

4.4 MILP Model for Energy Efficient Distributed Processing with Non-Splittable IoT Services

We begin to describe in detail, the granular power consumption modelling of all the equipment depicted in the proposed architecture in Figure 4.1. Since the devices involved in the considered architecture span multiple heterogeneous layers, it becomes a necessity to fairly represent the utilization characteristics of these devices. For example, high-capacity networking elements such as OLTs, metro/ core routers and switches are used by many other types of applications in addition to the IoT and it wouldn't make a fair evaluation if the total idle power consumption of these devices were wholly attributed to a small number of IoT services, hence the factor of δ is assumed for such devices.

While it is valid to assume that, the desirable power consumption profile should be a fully load dependent one, however in practical circumstances, this is not the case. It is reported in [107], that almost all devices adopt a linear power profile that consists of an idle and proportional part as depicted in Figure 4.4. With the former, power is consumed as soon as the device is activated however the latter dependents on various parameters such as frequency, voltage, or workload. In practice, idle power draws a large proportion of the maximum power of a networking/ processing device and hence it cannot be ignored. The total power consumption considering the linear profile with the idle power consumption of a networking/ processing device is calculated using equation (4.1)

$$Power\ Consumption\ = \left(\frac{Pmax - Pidle}{C}\right)\lambda + \ Pidle \tag{4.1}$$

where $Pidle$ is the idle power consumption of the device which is consumed as soon as the device is activated regardless of the load λ and ($Pmax$) is the maximum power consumption of the device, when it is 100% utilised at full capacity C (either in bps or MIPS). The proportional section of the power profile model for networking devices is expressed as energy per bit and likewise, for processing, it is expressed as energy per instruction.

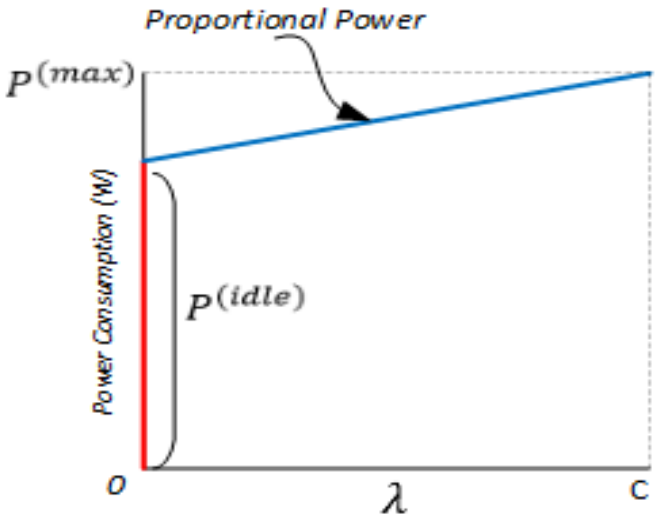


Figure 4.4 Linear Power Profile with Idle Power Consumption.

Before introducing the MILP model, the sets, parameters and variables used are defined as follows:

Sets:

N	Set of all nodes.
N_m	Set of neighbour nodes of node m in the proposed architecture.
C	Set of core nodes in the IP/WDM network, where $C \subset N$.
ONU	Set of ONUs in the PON network, where $ONU \subset N$.
OLT	Set of OLTs in the PON network, where $OLT \subset N$.
$M^{(R)}$	Set of metro routers, where $M^{(R)} \subset N$.
$M^{(Sw)}$	Set of metro switches, where $M^{(Sw)} \subset N$.
DC	Set of data centre nodes, where $DC \subset N$.
I	Set of all IoT devices, where $I \subset N$.
P	Set of nodes with processing devices, where $P \subset N$ and $P = I \cup ONU \cup OLT \cup M^{(Sw)} \cup DC$.
S	Set of IoT devices acting as source nodes where $S \subset I$.

Core Network Parameters:

$Pmax^{(r)}$	Maximum power consumption of an IP router port in the core network.
$Pmax^{(t)}$	Maximum power consumption of a transponder in the core network.

$Pmax^{(e)}$	Maximum power consumption of an EDFA in the core network.
$Pmax^{(o)}$	Maximum power consumption of an optical switch in the core network.
$Pmax^{(rg)}$	Maximum power consumption of a regenerator in the core network.
$Pidle^{(r)}$	Idle power consumption of an IP router port in the core network.
$Pidle^{(t)}$	Idle power consumption of a transponder in the core network.
$Pidle^{(e)}$	Idle power consumption of an EDFA in the core network.
$Pidle^{(o)}$	Idle power consumption of an optical switch in the core network.
$Pidle^{(rg)}$	Idle power consumption of a regenerator in the core network.
B	Maximum bit rate of single wavelength.
W	Number of wavelengths in a fibre in the core network.
$Eb^{(r)}$	Energy per bit of a router port, where $Eb^{(r)} = \left(\frac{Pmax^{(r)} - Pidle^{(r)}}{B}\right)$.
$Eb^{(t)}$	Energy per bit of a transponder, where $Eb^{(t)} = \left(\frac{Pmax^{(t)} - Pidle^{(t)}}{B}\right)$.

$Eb^{(e)}$ Energy per bit of the EDFA's, where $Eb^{(e)} = \left(\frac{Pmax^{(e)} - Pidle^{(e)}}{B}\right)$.

$Eb^{(o)}$ Energy per bit of the optical switches, where $Eb^{(o)} = \left(\frac{Pmax^{(o)} - Pidle^{(o)}}{B}\right)$.

$Eb^{(rg)}$ Energy per bit of regenerators, where $Eb^{(rg)} = \left(\frac{Pmax^{(rg)} - Pidle^{(rg)}}{B}\right)$.

D_{mn} Distance between two core nodes m and n, where $m, n \in C$.

$S^{(EDFA)}$ Span distance between two EDFAs.

$S^{(rg)}$ Span distance between two regenerators.

A_{mn} Number of EDFAs used on each fibre in the core network from node $m \in C$ to $n \in C, A_{mn} = \left\lfloor \left(\left(\frac{D_{mn}}{S^{(EDFA)}} \right) - 1 \right) \right\rfloor + 2$.

$\Re_{mn}$ Number of regenerators used between core node $m \in C$ and core node $n \in C, \Re_{mn} = \left\lfloor \left(\frac{D_{mn}}{S^{(rg)}} \right) - 1 \right\rfloor$.

$PUE^{(core)}$ Power Usage Effectiveness of IP/WDM core network node.

Cloud Data Centre Parameters:

$Pmax^{(DcSw)}$ Maximum power consumption of Cloud DC switch.

$Pidle^{(DcSw)}$ Idle power consumption of Cloud DC switch.

$BR^{(DcSw)}$ Bit rate of Cloud DC switch.

$Eb^{(DcSw)}$ Cloud DC switch energy per bit, where $Eb^{(DcSw)} = \left(\frac{Pmax^{(DcSw)} - Pidle^{(DcSw)}}{BR^{(DcSw)}}\right)$.

$Pmax^{(DcR)}$ Maximum power consumption of Cloud DC router.

$Pidle^{(DcR)}$ Idle power consumption of Cloud DC router.

$BR^{(DcSR)}$ Cloud DC router bit rate.

$Eb^{(DcR)}$ Energy per bit of a Cloud DC router, where $Eb^{(DcR)} = \left(\frac{Pmax^{(DcR)} - Pidle^{(DcR)}}{BR^{(DcR)}}\right)$.

$PUE^{(DC)}$ Power Usage Effectiveness of DC node, for processing and networking.

Metro Network and Fog Parameters:

$Pmax^{(MSw)}$ Maximum power consumption of a metro switch.

$Pidle^{(MSw)}$ Idle power consumption of a metro switch.

$BR^{(MSw)}$ Bit rate of a metro switch.

$Eb^{(MSw)}$ Metro switch energy per bit, where $Eb^{(MSw)} = \left(\frac{Pmax^{(MSw)} - Pidle^{(MSw)}}{BR^{(MSw)}}\right)$.

$Pmax^{(MR)}$	Maximum power consumption of a metro router.
$Pidle^{(MR)}$	Idle power consumption of a metro router.
$BR^{(MR)}$	Bit rate of a metro router.
$Eb^{(MR)}$	Metro router energy per bit, where $Eb^{(MR)} = \frac{Pmax^{(MR)} - Pidle^{(MR)}}{BR^{(MR)}}$.
$Pmax^{(MfSw)}$	Maximum power consumption of a metro fog switch.
$Pidle^{(MfSw)}$	Idle power consumption of a metro fog switch.
$BR^{(MfSw)}$	Bit rate of a metro fog switch.
$Eb^{(MfSw)}$	Metro fog switch energy per bit, where $Eb^{(MfSw)} = \left(\frac{Pmax^{(MfSw)} - Pidle^{(MfSw)}}{BR^{(MfSw)}}\right)$.
$Pmax^{(MfR)}$	Maximum power consumption of a metro fog router.
$Pidle^{(MfR)}$	Idle power consumption of a metro fog router.
$BR^{(MfR)}$	Bit rate of a metro fog router.
$Eb^{(MfR)}$	Metro fog router energy per bit, where $Eb^{(MfR)} = \left(\frac{Pmax^{(MfR)} - Pidle^{(MfR)}}{BR^{(MfR)}}\right)$.
$PUE^{(metro)}$	Power Usage Effectiveness of a metro node, for processing and networking.
ψ	Metro router port redundancy.

Access Network and Fog Parameters:

$Pmax^{(OLT)}$ Maximum power consumption of OLT in the PON network.

$Pidle^{(OLT)}$ Idle power consumption of OLT in the PON network.

$BR^{(OLT)}$ Bit rate of OLT in the PON network.

$Eb^{(OLT)}$ OLT router energy per bit, where $Eb^{(OLT)} = \left(\frac{Pmax^{(OLT)} - Pidle^{(OLT)}}{BR^{(OLT)}}\right)$.

$Pmax^{(ONU)}$ Maximum power consumption of an ONU in the PON network.

$Pidle^{(ONU)}$ Idle power consumption of an ONU in the PON network.

$BR^{(ONU)}$ Bit rate of the WiFi interface of an ONU in the PON network.

$Eb^{(ONU)}$ ONU energy per bit, where $Eb^{(ONU)} = \left(\frac{Pmax^{(ONU)} - Pidle^{(ONU)}}{BR^{(ONU)}}\right)$.

$Pmax^{(AfSw)}$ Maximum power consumption of an access fog switch.

$Pidle^{(AfSw)}$ Idle power consumption of an access fog switch.

$BR^{(AfSw)}$ Bit rate of an access fog switch.

$Eb^{(AfSw)}$ Access fog switch energy per bit, where $Eb^{(AfSw)} = \left(\frac{Pmax^{(AfSw)} - Pidle^{(AfSw)}}{BR^{(AfSw)}}\right)$.

$Pmax^{(AfR)}$ Maximum power consumption of an access fog router.

$Pidle^{(AfR)}$ Idle power consumption of an access fog router.

$BR^{(AfR)}$ Bit rate of an access fog router.

$Eb^{(AfR)}$ Access fog router energy per bit, where $Eb^{(AfR)} = \left(\frac{Pmax^{(AfR)} - Pidle^{(AfR)}}{BR^{(AfR)}}\right)$.

$Pmax^{(cpefSw)}$ Maximum power consumption of CPE fog switch.

$Pidle^{(cpefSw)}$ Idle power consumption of a CPE fog switch.

$BR^{(cpefSw)}$ Bit rate of a CPE fog switch.

$Eb^{(cpefSw)}$ CPE fog switch energy per bit, where $Eb^{(cpefSw)} = \left(\frac{Pmax^{(cpefSw)} - Pidle^{(cpefSw)}}{BR^{(cpefSw)}}\right)$.

$PUE^{(access)}$ Power Usage Effectiveness of an access fog node, for processing and networking.

IoT Devices' Parameters:

$Pmax^{(TxRx)}$ Maximum power consumption of an IoT transceiver.

$Pidle^{(TxRx)}$ Idle power consumption of an IoT transceiver.

$BR^{(TxRx)}$ Bit rate of the WiFi interface of an IoT device.

$Eb^{(TxRx)}$ IoT WiFi interface energy per bit, $Eb^{(TxRx)} = \left(\frac{Pmax^{(TxRx)} - Pidle^{(TxRx)}}{BR^{(TxRx)}}\right)$.

IoT Demand Parameters:

$D_s^{(CPU)}$ CPU demand of task originating at IoT source node $s \in S$, in Million Instructions Per Second (MIPS).

$D_s^{(Bw)}$ Bandwidth demand of task originating at IoT source node $s \in S$, in Mbps

Processing Devices' Parameters

$Pmax_d^{(CPU)}$ Maximum power consumption of processing device $d \in P$, in Watts.

$Pidle_d^{(CPU)}$ Idle power consumption of processing device $d \in P$, in Watts.

$C_d^{(CPU)}$ Maximum capacity of processing device $d \in P$ in Million Instructions Per Second (MIPS).

Ei_d Energy per instruction of processing device $d \in P$, where $Ei_d = \left(\frac{Pmax_d^{(CPU)} - Pidle_d^{(CPU)}}{C_d^{(CPU)}}\right)$.

Application Parameters:

δ Portion of the idle power of equipment attributed to the application.

K Number of subtasks an IoT task can be divided into.

Δ Number of MIPS required to process 1 Mb of traffic.

M Large enough number.

Variables:

λ^{sd} Traffic demand between IoT source node $s \in S$ and processing device $d \in P$.

λ_{mn}^{sd} Traffic flow between IoT source node $s \in S$ and processing device $d \in P$, traversing link (m, n), where $m \in N, n \in N_m$.

λ_d Volume of traffic aggregated by node $d \in N$.

$\mathcal{B}_m$ $\mathcal{B}_m = 1$, if network node $m \in N$ is activated, otherwise $\mathcal{B}_m = 0$.

θ_d Traffic in node $d \in P$ for processing.

Γ_{mn} $\Gamma_{mn} = 1$, if core network link m, n, where $m \in C, n \in (N_m \cap C)$ is activated, otherwise $\Gamma_{mn} = 0$.

ρ^{sd} Processing demand of IoT source node $s \in S$ hosted at processing device $d \in P$.

Ω^{sd} $\Omega^{sd} = 1$, if processing demand of IoT source node $s \in S$ is hosted at destination node $d \in P$, otherwise $\Omega^{sd} = 0$.

Ω^{d} $\Omega^{d} = 1$, if processing node $d \in P$ is activated, otherwise $\Omega^{d} = 0$.

$\mathcal{N}_d$ Number of processing servers activated at node $d \in P$.

W_{mn} Number of wavelengths used in fibre link (m, n) in the core network, where link $m, n \in C$.

F_{mn} Number of fibres used on link $m, n \in C$.

Ag_m Number of aggregation router ports activated at IP node $m \in C$.

The total power consumption of the entire IoT infrastructure depicted in Figure 4.1, is divided into two parts: 1) Network Power Consumption and 2) Processing Power Consumption. The following subsections contain a detailed breakdown of these power consumptions:

4.4.1 Network Power Consumption

Under the non-bypass light path approach [108], the IP/WDM total network power consumption is composed of:

1) The power consumption of router ports:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(r)}\lambda_m\right) + \sum_{m\in C}\left(Pidle^{(r)}\left(Ag_m + \sum_{n\in(N_m\cap C)} W_{mn}\right)\right)\right) \quad (4.2)$$

2) The Power consumption of transponders:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(t)}\lambda_m\right) + \sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(t)}W_{mn}\right)\right) \quad (4.3)$$

3) The power consumption of EDFAs:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(e)}\lambda_m A_{mn}F_{mn}\right) + \sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(e)}A_{mn}F_{mn}\right)\right) \tag{4.4}$$

4) The power consumption of optical switches:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(o)}\lambda_m\right) + \sum_{m\in C}\left(Pidle^{(o)}\mathcal{B}_m\right)\right) \tag{4.5}$$

5) The power consumption of regenerators:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(rg)}\lambda_m\Re_{mn}W_{mn}\right) + \sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(rg)}\,\Re_{mn}W_{mn}\right)\right) \tag{4.6}$$

The metro network's power consumption is composed of:

$$PUE^{(metro)}\left(\left(\sum_{m\in M^{(R)}}\left(Eb^{(MR)}\lambda_m\psi\right) + \sum_{m\in M^{(R)}}\left(\mathcal{B}_m Pidle^{(MR)}\psi\right)\right) + \left(\sum_{m\in M^{(Sw)}}\left(Eb^{(MSw)}\lambda_m\right) + \sum_{m\in M^{(R)}}\left(\mathcal{B}_m Pidle^{(MSw)}\right)\right)\right) \tag{4.7}$$

The access network's power consumption consists of the power consumption of OLT and ONU devices, which is given as:

$$PUE^{(access)}\left(\sum_{m\in OLT}\left(Eb^{(OLT)}\lambda_m\right)+\sum_{m\in OLT}\left(\mathcal{B}_m Pidle^{(OLT)}\right)\right) + \left(\sum_{m\in ONU}\left(Eb^{(ONU)}\lambda_m\right)+\sum_{m\in ONU}\left(\mathcal{B}_m Pidle^{(ONU)}\right)\right) \quad (4.8)$$

The IoT devices' communication interfaces power consumption is given as:

$$\sum_{m\in I}\left(Eb^{(TxRx)}\lambda_m\right)+\sum_{m\in I}\mathcal{B}_m Pidle^{(TxRx)} \quad (4.9)$$

4.4.2 Processing Power Consumption

The total power consumption of the processing devices (or servers) is composed of:

1) The processing power consumption of IoT devices:

$$\sum_{s\in S}\sum_{d\in I}\left(Ei_d\rho^{sd}\right)+\sum_{d\in I}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right) \quad (4.10)$$

2) The processing power consumption of CPE fog (CF) servers:

$$\sum_{s\in S}\sum_{d\in ONU}\left(Ei_d\rho^{sd}\right)+\sum_{d\in ONU}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right) \quad (4.11)$$

3) The processing power consumption of access fog (AF) servers:

$$PUE^{(access)}\left(\sum_{s\in S}\sum_{d\in OLT}(Ei_d\rho^{sd})+\sum_{d\in OLT}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right) \tag{4.12}$$

4) The processing power consumption of metro fog (MF) servers:

$$PUE^{(metro)}\left(\sum_{s\in S}\sum_{d\in M^{(Sw)}}(Ei_d\rho^{sd})+\sum_{d\in M^{(Sw)}}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right) \tag{4.13}$$

5) The processing power consumption of cloud DC servers

$$PUE^{(dc)}\left(\sum_{s\in S}\sum_{d\in DC}(Ei_d\rho^{sd})+\sum_{d\in DC}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right) \tag{4.14}$$

4.4.3 Power Consumption of Network inside Processing Nodes

The cloud DCs network power consumption is composed of the power consumption of cloud DC routers and switches:

$$PUE^{(DC)}\left(\left(\sum_{d\in DC}\left(Eb^{(DcSw)}\theta_d\right)+\sum_{d\in DC}\left(Pidle^{(DcSw)}\Omega^d\right)\right)+\left(\sum_{d\in DC}\left(Eb^{(DcR)}\theta_d\right)+\sum_{m\in DC}\left(Pidle^{(DcR)}\,\Omega^d\right)\right)\right) \tag{4.15}$$

The metro fog network power consumption of metro fog routers and switches is given as:

$$PUE^{(metro)}\left(\left(\sum_{d\in M^{(Sw)}}\left(Eb^{(MfR)}\theta_d\right)+\sum_{m\in M^{(Sw)}}\left(Pidle^{(MfR)}\Omega^d\right)\right)+\left(\sum_{d\in M^{(Sw)}}\left(Eb^{(MfSw)}\theta_d\right)+\sum_{d\in M^{(Sw)}}\left(Pidle^{(MfSw)}\right)\Omega^d\right)\right) \quad (4.16)$$

The access fog network power consumption of access fog routers and switches is given as:

$$PUE^{(access)}\left(\left(\sum_{d\in OLT}\left(Eb^{(AfR)}\theta_d\right)+\sum_{d\in OLT}\left(Pidle^{(AfR)}\Omega^d\right)\right)+\left(\left(\sum_{d\in OLT}\left(Eb^{(AfSw)}\theta_d\right)+\sum_{d\in OLT}\left(Pidle^{(AfSw)}\Omega^d\right)\right)\right) \quad (4.17)$$

The CPE fog network power consumption of CPE fog switches is given as:

$$\sum_{d\in ONU}\left(Eb^{(cpefSw)}\theta_d\right)+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\Omega_d\right) \quad (4.18)$$

The MILP model's objective function is given as follows:

Objective

Minimise the total power consumption:

$$PUE^{(core)}\left[\sum_{m\in C}\left(Eb^{(r)}\lambda_m\right)\right. \tag{4.19}$$

$$\left.+\sum_{m\in C}\left(Pidle^{(r)}\left(Ag_m+\sum_{n\in(N_m\cap C)}W_{mn}\right)\right)\right]+$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(Eb^{(t)}\lambda_m\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(t)}W_{mn}\right)\right)\right]+$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(Eb^{(e)}\lambda_m A_{mn}F_{mn}\right)\right.\right.$$

$$\left.\left.+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(e)}A_{mn}F_{mn}\right)\right)\right]+$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(Eb^{(o)}\lambda_m\right)+\sum_{m\in C}\left(Pidle^{(o)}\mathcal{B}_m\right)\right)\right]+$$

$$PUE^{(core)}\left[\sum_{m\in C}\left(Eb^{(rg)}\lambda_m Rg_{mn}W_{mn}\right)\right.$$

$$\left.+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(rg)}\Re_{mn}W_{mn}\right)\right]$$

$$+PUE^{(metro)}\left[\sum_{m\in M^{(R)}}\left(Eb^{(MR)}\lambda_m\psi\right)\right.$$

$$+\sum_{m\in M^{(R)}}\left(\mathcal{B}_m Pidle^{(MR)}\psi\right)+\sum_{m\in M^{(Sw)}}\left(Eb^{(MSw)}\lambda_m\right)$$

$$\left.+\sum_{m\in M^{(R)}}\left(\mathcal{B}_m Pidle^{(MSw)}\right)\right]$$

$$+\ PUE^{(access)}\left[\sum_{m\in OLT}\left(Eb^{(OLT)}\lambda_m\right)\right.$$

$$\left.+\sum_{m\in OLT}\left(\mathcal{B}_m Pidle^{(OLT)}\right)\right]+$$

$$PUE^{(core)}\left[\sum_{m\in ONU}\left(Eb^{(ONU)}\lambda_m\right)+\sum_{m\in ONU}\left(\mathcal{B}_m Pidle^{(ONU)}\right)\right]+$$

$$\left[\sum_{m\in I}\left(Eb^{(TxRx)}\lambda_m\right)+\sum_{m\in I}Pidle^{(TxRx)}\mathcal{B}_m\right]$$

$$+\left[\sum_{s\in S}\sum_{d\in I}(Ei_d\rho^{sd})+\sum_{d\in I}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right]$$

$$+\left[\sum_{s\in S}\sum_{d\in ONU}(Ei_d\rho^{sd})+\sum_{d\in ONU}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right]$$

$$+\ PUE^{(access)}\left[\sum_{s\in S}\sum_{d\in OLT}(Ei_d\rho^{sd})\right.$$

$$\left.+\sum_{d\in OLT}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right]$$

$$+\ PUE^{(metro)}\left[\sum_{s\in S}\sum_{d\in M^{Sw}}(Ei_d\rho^{sd})\right.$$

$$\left.+\sum_{d\in M^{(Sw)}}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right]$$

$$+PUE^{(dc)}\left[\sum_{s\in S}\sum_{d\in DC}(Ei_d\rho^{sd})+\sum_{d\in DC}\left(Pidle_d^{(cpu)}\,\mathcal{N}_d\right)\right]+$$

$$PUE^{(DC)}\left[\sum_{d\in DC}\left(Eb^{(DcSw)}\theta_d\right)\right.$$
$$+\sum_{m\in DC}\left(Pidle^{(DcSw)}\Omega^d\right)+\sum_{d\in DC}\left(Eb^{(DcR)}\theta_d\right)$$
$$\left.+\sum_{d\in DC}\left(Pidle^{(DcR)}\,\Omega^d\right)\right]$$
$$+PUE^{(metro)}\left[\sum_{d\in M^{(Sw)}}\left(Eb^{(MfR)}\theta_d\right)\right.$$
$$+\sum_{d\in M^{(Sw)}}\left(Pidle^{(MfR)}\Omega^d\right)+\sum_{d\in M^{(Sw)}}\left(Eb^{(MfSw)}\theta_d\right)$$
$$\left.+\sum_{d\in M^{(Sw)}}\left(Pidle^{(MfSw)}\Omega^d\right)\right]$$
$$+PUE^{(access)}\left[\sum_{m\in OLT}\left(Eb^{(AfR)}\lambda_m\right)\right.$$
$$+\sum_{m\in OLT}\left(\mathcal{B}_m Pidle^{(AfR)}\right)$$
$$\left.+\sum_{m\in OLT}\left(Eb^{(AfSw)}\theta_m\right)+\sum_{m\in OLT}\left(\mathcal{B}_m Pidle^{(AfSw)}\right)\right]$$
$$+\left[\sum_{d\in ONU}\left(Eb^{(cpefSw)}\theta_d\right)+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\Omega^d\right)\right]$$

Subject to:

$$\sum_{n\in N_m}\lambda_{mn}^{sd}-\sum_{n\in N_m}\lambda_{nm}^{sd}=\begin{cases}\lambda_{sd} & m=s\\ -\lambda_{sd} & m=d\\ 0 & otherwise\end{cases} \tag{4.20}$$

$$\forall s\in S, d\in P, m\in N: s\neq d.$$

Constraint (4.20) conserves traffic from the source node to the destination node in the considered topology depicted in Figure 4.1. It ensures that the

total incoming traffic at a node is equal to the total outgoing traffic of that node; unless the node in question is either the source node or the destination node.

$$\sum_{d \in P} \rho^{sd} = D_s^{(CPU)} \tag{4.21}$$

$$\forall s \in S$$

Constraint (4.21) ensures that processing task per IoT source node $s \in S$ is met at a given destination node.

$$\rho^{sd} \geq \Omega^{sd} \tag{4.22}$$

$$\forall s \in S, d \in P$$

$$\rho^{sd} \leq M\Omega^{sd} \tag{4.23}$$

$$\forall s \in S, d \in P$$

Constraints(4.22) and (4.23) are used in the conversion of ρ^{sd} into its binary equitant. When $\rho^{sd} = 1$, the source node $s \in S$ is processes its CPU task request at destination node $d \in P$.

$$\sum_{d \in P} \Omega^{sd} \leq K \tag{4.24}$$

$$\forall s \in S$$

Constraint (4.24) ensures that processing tasks are placed at a single location only, hence, no task splitting is allowed.

$$\mathcal{N}_d \geq \sum_{s \in S} \frac{\rho^{sd}}{C_d^{(CPU)}} \tag{4.25}$$

$$\forall d \in P$$

Constraint (4.25) determines the number of servers required at processing node $d \in P$.

$$\mathcal{N}_d \leq \mathcal{V}_d \tag{4.26}$$

$$\forall d \in P$$

Constraint (4.26) ensures that, the number of servers activated at a processing node $d \in P$, does not exceed the maximum available number of servers in that node.

$$\sum_{s \in I} \Omega^{sd} \geq \Omega^d \tag{4.27}$$

$$\forall d \in P$$

$$\sum_{s \in I} \Omega^{sd} \leq M\, \Omega^d \tag{4.28}$$

$$\forall d \in P$$

Constraints (4.27) and (4.28) are used to ensure that, the binary variable $\Omega^d = 1$ if processing node $d \in P$ is activated, otherwise $\Omega^d = 0$.

$$\lambda_m = \sum_{\substack{s \in S: \\ m=s}} \sum_{d \in P} \sum_{n \in N_m} \lambda_{mn}^{sd} \tag{4.29}$$

$$\forall m \in S$$

$$\lambda_m = \sum_{\substack{s \in S: \\ m \neq s}} \sum_{\substack{d \in P: \\ s \neq d}} \sum_{n \in N_m} \lambda_{nm}^{sd}$$

$$\forall m \in (I \cup OLT \cup M^{(Sw)} \cup M^{(R)} \cup DC) \tag{4.30}$$

$$\lambda_m = \sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} \sum_{\substack{n \in N_m: \\ n \in (N_m \cap C)}} \lambda_{mn}^{sd} \tag{4.31}$$

$$\forall m \in C$$

Constraint (4.29) ensures that the total aggregate traffic on node $m \in S$ is accounted for only when the source node is transmitting. Whilst constraint (4.30) ensures that, the aggregate traffic on node $m \in N,$ where $m \notin C,$ is only accounted for if the transmitting node $m \neq s$ is not the source of the traffic. Finally constraint (4.31) determines the aggregate traffic in the core network, given that the transmitting node $m \in C$ is not equal to the source of the traffic node $s \in S$.

$$\theta_d \leq M\Omega^d \tag{4.32}$$

$$\forall d \in P$$

$$\theta_d \leq \lambda_d \tag{4.33}$$

$$\forall d \in P$$

$$\theta_d \geq \lambda_d - (1 - \Omega^d)M \tag{4.34}$$

$$\forall d \in P$$

Constraints (4.38), (4.29) and (4.30) are used to linearise the product of the binary Ω^d, where $d \in P$ and the continuous variable Ω^d, where $d \in P$. This

ensures that traffic on a processing node $d \in P$ is only accounted for if it is destined to that node for processing.

$$\lambda_m \geq \mathcal{B}_m \tag{4.35}$$

$$\forall m \in N$$

$$\lambda_m \leq M\mathcal{B}_m \tag{4.36}$$

$$\forall m \in N$$

Constraint (4.35) and (4.36) convert the continuous variable λ_m, where $m \in N$ into its binary equivalent.

$$\lambda^{sd} = D_s^{(BW)}\Omega_{sd} \tag{4.37}$$

$$\forall s \in S, d \in P$$

Constraint (4.37) ensures that the total traffic demand for each source node is met. The binary variable Ω_{sd} ensures that traffic is only directed to the destination node that is hosting a processing task.

$$\sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} \lambda_{mn}^{sd} \leq C_{mn} \tag{4.38}$$

$$\forall m \in \left(I \cup ONU \cup OLT \cup M^{(Sw)} \cup M^{(R)} \cup DC\right): n \in N_m$$

Constraint (4.38) ensures that the total traffic carried on link m, n, in the metro and access layer only does not exceed its capacity in Mbps.

$$Ag_m \geq \frac{\lambda_m}{B} \tag{4.39}$$

$$\forall m \in C$$

Constraint (4.39) gives the number of aggregation router ports at each IP/WDM node.

$$\sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} \lambda_{mn}^{sd} \leq W_{mn} B \qquad \forall m \in C: n \in (C \cap N_m) \tag{4.40}$$

$$W_{mn} \leq WF_{mn} \qquad \forall m \in C: n \in (C \cap N_m) \tag{4.41}$$

Constraints (4.40) and (4.41) represent the physical link capacity of the IP/WDM optical links. Constraint (4.40) ensures that the total traffic on a link does not exceed the capacity of a single wavelength while constraint (4.41) ensures the total number of wavelength channels does not exceed the capacity of a single fibre link.

4.5 Input Data for the MILP Model

To evaluate the performance of the proposed model, the IoT network is modelled as a graph $G(N, L)$, where N is the set of all nodes and L is the set of bidirectional links connecting those nodes. A subset $I \subset N$ represents the set of all the IoT devices in the considered network, whilst only a subset $S \subseteq N$ act as demand source nodes. A subset of processing nodes, where $P \subset N$ and $P \subset \left(I \cup ONU \cup OLT \cup M^{(Sw)} \cup DC\right)$ act as processing nodes. The processing node $d \in P$ has a maximum computational capacity $C_d^{(CPU)}$ measured in Million Instructions Per Second (MIPS). Also, each link

(m, n), where $m \in N$ and $n \in N_m$, has a maximum bit rate (BR) measured in Gbps.

Some or all of the IoT devices are the sources of the workload. Each demand is characterised by a tuple $D(CPU, BW)$, where CPU is the amount of processing required in $kMIPS$ and BW is the amount of bandwidth required in Mb/s. These nodes consist of 20 devices in total and these

devices are divided into 4 groups equally, hence each group is connected to the PON network via a single ONU as shown in the proposed architecture in Figure 3.2.

4.5.1 Workload Intensity Definition

In our evaluations, we have made CPU requirement proportional to traffic (BW), such that, for every bit of traffic 1000 MIPS is required. Although, it is beyond the scope of the work in this thesis, measuring CPU efficiency by MIPS is not an accurate benchmark, since different CPUs have different architectures, hence varied performances for the same task. Nevertheless, this does not stop us from making a starting point by consulting the literature to obtain realistic values. In [109], the authors have reported that for a specific visual processing algorithm referred to as Analyse Then Compress (ATC), for a file of 10KB (0.08Mb), 69.23 MIPS are required for processing for visual object recognition. Thus, through simple calculations we derived how many MIPS are required (Δ) to process 1Mb of traffic in 1 second as follows, using equation (4.42):

$$\Delta = \frac{69.23}{0.08} \cong 865.4. \qquad (4.42)$$

For the sake of simplifying analysis and being conservative, we assume that each 1Mb of traffic requires approximately 1000 MIPS of processing, in 1 second.

As for the traffic requirement, we use an online tool to estimate the required data rates for different resolutions and this was estimated to be between 1 – 10 Mbps, which covers video resolutions between 1024×720 to 1600×1200 at 30 frames per second[3]. The CPU workload intensity is then calculated by multiplying the Δ by the amount of traffic. Thus, this makes the CPU demand proportional to the size of the traffic due to the assumption that the higher the traffic, the more features a video file will hold, thus more CPU instructions are required to process that file.

Compression Type	Video Format	Resolution	Bandwidth (Mbps)
H.264	PAL D1	720 x 576	0.5
	0,8 MPx	1024 x 768	0.9
	HD 720p	1280 x 720	1
MPEG	0,2 MPx	640 x 360	1.3
	PAL D1	720 x 576	2.4
	HD 720p	1280 x 720	5.3
	1,2 MPx	1280 x 960	7.1
	1,9 MPx	1600 x 1200	11.1
	Full HD	1920 x 1080	12

Table 1 Data rate of various video files used as guide.

4.5.2 Equipment Idle Power Consumption Attributed to IoT Application

In our evaluations, we have made use of equipment datasheets where ever possible to report the power consumption of the devices. However, it is not always feasible to obtain this information from device datasheets, hence, we

[3] https://www.cctvcalculator.net/en/calculations/bandwidth-calculator/.

make realistic assumptions based on the literature. In terms of idle power consumption, based on [71], most high capacity networking equipment such as metro/core routers and switches consume 90% of the equipment's maximum power. As for processing servers' idle power consumption, based on [110], we assume it is 60% of the maximum power consumption of the CPU.

Moreover, we assume that IoT applications are only responsible for a portion of $Pidle$ of high capacity networking equipment. This assumption is valid, for instance, metro switches are used to serve thousands of different users simultaneously, thus it would not make a fair analysis if all of $Pidle$ was attributed to a specific application like the one considered in this thesis. Thus, we make use of Cisco's visual networking index for the years 2017-2022 to estimate the total traffic of surveillance type applications similar to the one considered in this work. It is reported that, globally, 3% of all video traffic on the internet is due to surveillance services, hence the portion of idle power δ attributed to the application in question is 3% [111].

4.5.3 Power Usage Effectiveness (PUE)

In our evaluations, PUE is not considered for the IoT and ONU devices, as there is generally no cooling requirements for them [112]. The power usage effectiveness (PUE) is the ratio of the total power consumed by a facility (i.e. ISP networks, data centres) to the total power consumption of the equipment within the facility (i.e. servers, switches, routers, etc). In 2018, Google reported that one of its data centres is currently operating at a PUE of 1.15. We make use of a report published in 2016 which estimates the PUE values of various

data centres base on “Space Type” [98]. Within the report, it is shown that PUE values progressively decrease with the increase in the “Space Type”.

Thus, in a similar fashion, we increase PUE progressively in the proposed network architecture since the largest “Space Type” is generally hyper-scale data centres connected to the core network. It is assumed that the PUE at any layer applies to both processing and networking equipment. The PUE value of the core network is referenced from one of our previous works, and this is assumed to be 1.5 [114]. Table 2 is a summary of the PUE values used in the model.

Network Layer	**PUE**
IoT Devices	1
CPE Fog (CF)	1
Access Fog ($PUE^{(access)}$)	1.5
Metro Fog ($PUE^{(metro)}$)	1.4
Cloud DC ($PUE^{(DC)}$)	1.12 [113]
Core Network (($PUE^{(core)}$))	1.5 [114]

Table 2 PUE values of all the layers of the proposed architecture.

The network parameters used in the MILP model are summarised in Table 3. The parameters consist of the maximum and idle power consumption of the devices in question. The IoT and ONU devices’ power consumption is attributed to their WiFi transceivers and their value of δ and PUE set is to 1. Whilst the other remaining equipment have a PUE and a value of δ associated to them, due to their high capacity.

Device	**Pmax (W)**	**Pidle (W)**	$\boldsymbol{\delta}$	**BR (Gb/s)**	**PUE**
IoT (WiFi)	0.56	0.34	-	0.1	1
ONU (WiFi)	15	9	-	0.3	1
OLT	1940	60	3%	8600	1.5
Metro Router Port	30	27	3%	40	1.4
Metro Ethernet Switch	470	423	3%	600	1.4
Metro Router Port Redundancy (ψ)	2				

Table 3 Network Parameters for the MILP Model

The core network consists of a number of core routers interconnected by IP/WDM technologies. We have considered the DC nodes to be only a single hop from the user traffic and the average distance between two neighbouring core nodes is 2010km (based on the AT&T US network topology). The power consumption values of the core network in the MILP model are summarised in Table 4.

Distance between two neighbouring EDFAs ($S^{(EDFA)}$	80 (km) [115]
Number of wavelengths in a fibre (W)	32 [115]
Bitrate of a wavelength (B)	40 Gb/s
Distance between two neighbouring core nodes D_{mn}	2500km
Maximum power consumption of a router port $Pmax^{(r)}$	638 (W) [115]
Idle power consumption of a router port $Pidle^{(r)}$	574.2 (W)
Energy per bit of a router port $Eb^{(r)}$	1.6 W/Gb/s
Maximum power consumption of a transponder $Pmax^{(t)}$	129 (W) [115]
Idle power consumption of a transponder $Pidle^{(t)}$	116 (W)
Energy per bit of a transponder $Eb^{(t)}$	0.32 (W/Gb/s)
Maximum power consumption of an optical switch $Pmax^{(o)}$	85 (W) [115]
Idle power consumption of a transponder $Pidle^{(o)}$	77 (W)
Energy per bit of a transponder $Eb^{(o)}$	0.2 (W/Gb/s)
Maximum power consumption of an optical switch $Pmax^{(e)}$	85 (W) [115]
Idle power consumption of a transponder $Pidle^{(e)}$	11 (W)
Energy per bit of a transponder $Eb^{(e)}$	0.02 (W/Gb/s)
Maximum power consumption of an optical switch $Pmax^{(rg)}$, reach 2500km	71.4 (W) [115]
Idle power consumption of a transponder $Pidle^{(rg)}$	64 (W)
Energy per bit of a transponder $Eb^{(rg)}$	0.19 (W/Gb/s)
Portion of the aggregate idle powers attributed to the application (δ)	3% [111]

Table 4 Input data of the core network for the MILP Model

The processing devices' input parameters are summarised in Table 5. In order to estimate the processing capacity of the servers in MIPS, we have made use of a technical benchmark, in which, it is reported that Intel high-end servers process 4 instructions/ cycle (i/c) [116]. Thus, to determine the maximum capacity of a processing device we have used the following

$$MIPS = clock \times \frac{Is}{Cycle} \qquad 4.43$$

Where $\frac{Is}{Cycle}$ is the number of instructions a CPU can execute per clock cycle which is given in GHz in Table 5. To differentiate between the types CPUs and their efficiencies, we set the Is/Cycle of the Metro Fog (MF) as a reference point. The efficiency of the processing decreases as one moves down the hierarchy (from the core to the IoT device).

Node	Device Model	Pmax (*W*)	Pidle (*W*)	C (*GHz*)	C (*MIPS*)	Ei (*W/kMIPS*)	Is/Cycle	PUE
SP-DC Server	NVidia T4 GPU	75 [117]	45	1.25[117]	1080k	27μ	864	1.12
GP-DC Server	Intel Xeon E5-2680	130 [118]	78	2.7[118]	108k	481μ	5	
MF Server	Intel X5675	95 [119]	57	3.06[119]	73.44k	517μ	4	1.4
AF Server	Intel Xeon E5-2420	95 [120]	57	1.9[120]	34.2k	1111μ	3	1.5
CF Server	RPi 3 Model B	12.5 [121]	2	1.2 [122]	2.4k	4375μ	2	2.5
IoT Device	RPi Zero W	3.96 [121]	0.5	1 [123]	1k	3460μ	1	1

Table 5 Input data of processing servers for the MILP model.

At those layers where multiple servers can be deployed, a networking infrastructure becomes a necessity to interconnect the multiple active servers. Hence, we have used routers and switches accordingly to achieve this. We have used realistic values for the processing networking equipment to differentiate between the many layers of the proposed architecture

in Figure 3.2. Generally, lower layers have been assigned lower specification devices where applicable, for instance, a metro switch is a much more power consuming equipment than an L2 switch at the access. Table 6 summarises the networking equipment used inside processing nodes.

Device	Pmax (W)	Pidle (W)	BR (Gb/s)	Eb (W/Gb/s)	PUE
CF Switch	1.78W [124]	0.36[124]	1.6[124]	0.89	1
AF Router	13W[125]	11.7	40[125]	0.03	1.4
AF Switch	210W[126]	189	240[126]	0.08	1.4
MF Router	13W[125]	11.7	40[125]	0.03	1.4
MF Switch	210W [126]	189	600[126]	0.04	1.4
DC LAN Router	30[125]	27	40[125]	0.08	1.5
DC LAN Switch	470[126]	423	600[126]	0.08	1.5

Table 6 Processing network input data for the model.

4.6 Power Consumption Evaluation

This section presents by results, the outcomes of the proposed energy efficient distributed MILP model for IoT with non-splittable services. The power consumption of the considered architecture depicted in Figure 4.1 is evaluated and the optimum processing locations of a range of representative workloads are found. We approach the optimisation problem using two design strategies: 1) an un-capacitated design problem, and 2) a capacitated design problem. It is worthy of mention that, IoT devices are in all cases capacitated in terms of processing only. For each design problem, there is a further breakdown in scenarios, a) having GP-DCs only, and b) having access to SP-DC as well.

Moreover, the complexity of the MILP models in the current and subsequent chapters grows exponentially with the increase in the number of IoT requests. Therefore, the total number of IoT nodes considered in a service request is between 1 and 20 requests which are distributed among 4 IoT groups uniformly with some degree of clustering depending on the distribution scenario (e.g. Scenario # 1 – Scenario #4). The four request scenarios are aimed at capturing the impact of the number of requests coupled with the location of those requests. For instance, Scenario #1 and Scenario #4 capture the extreme ends of the request distribution whilst Scenario #2 and Scenario #3 capture cases that lie between the two aforementioned scenarios. The total number of IoT devices in each IoT group is based on a representative home LAN network which typically connects a single to few users [41]. Choosing to minimise energy or power consumption is dependent on the type of system one is working with. For instance, if the system is such that only a finite amount of load is expected at a given time period and for the rest of the time the

system is not consuming any power then it would be more practical to minimise energy. However if on the other hand, the system is expected to work continuously over a given time period then minimising power would be more appropriate, which currently is the case in this thesis because it is assumed that the source nodes (IoT video cameras) are operating at homogenous bit rates and hence consume the same amount of power throughout the evaluation period.

AMPL software with CPLEX 12.5 solver is used as the platform for solving the MILP models and all the models were executed on a PC with an Intel Core i5-4460 CPU, running at 3.20 GHz, with 16 GB of RAM.

4.7 Un-Capacitated Design Problem with GP-DCs Only

In the un-capacitated design approach, it is assumed that the number of processing devices deployed at each node is unrestricted except in the devices located in the IoT layer due to their limited features. This approach aims to determine for a given demand volume, the optimum resources needed to host a given service if there are no restrictions on the network equipment capacity and no restrictions on the number of servers that can be hosted at each site. Note that in our evaluations, the network capacity is always sufficient to carry the traffic. Therefore the ‘capacitated restriction’ applies to the number of processing servers available at each location in our case. The goal is also to determine whether it is the optimal choice to build large numbers of devices at a given location in the proposed architecture. Generally, such design problems occur in medium to long term network design planning [127].

4.7.1 Scenario #1: A single active IoT

In this scenario, out of the total 20 IoT devices in the model, we take one end of the extremes and assume that only a single IoT device is active at any time instance and the rest of the IoT devices are in the idle mode. As expected and shown in Figure 4.5, for low workload values such as 1000 MIPS, significant savings (98%) can be achieved compared to the baseline solution, where the baseline solution is a scenario where processing is always carried out at the GP-DC. This is due to the local computational resource of the IoT device, hence, the costly overhead of the network and high idle powers of DC servers are avoided. However, as the workload increases and violates the capacity of the IoT device, we begin to see the intervention of the CF node as it is only a single hop from the IoT device. The general trend in this scenario always favours the activation of additional servers attached to the CF node due to its low idle power consumption compared with the servers located in the upper layers of the fog architecture. Moreover, the results indicate promising power savings of at least 70%, at the extreme end of the workloads (10,000 MIPS).

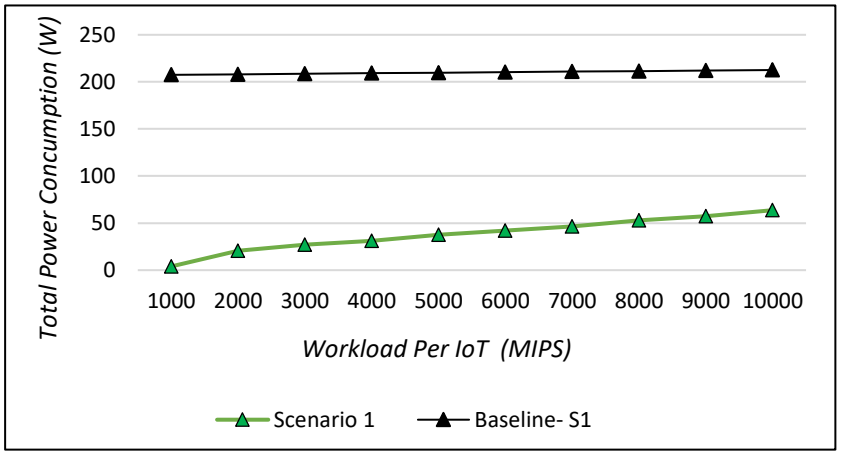


Figure 4.5 Total Power Consumption of Distributed Approach in Scenario #1.

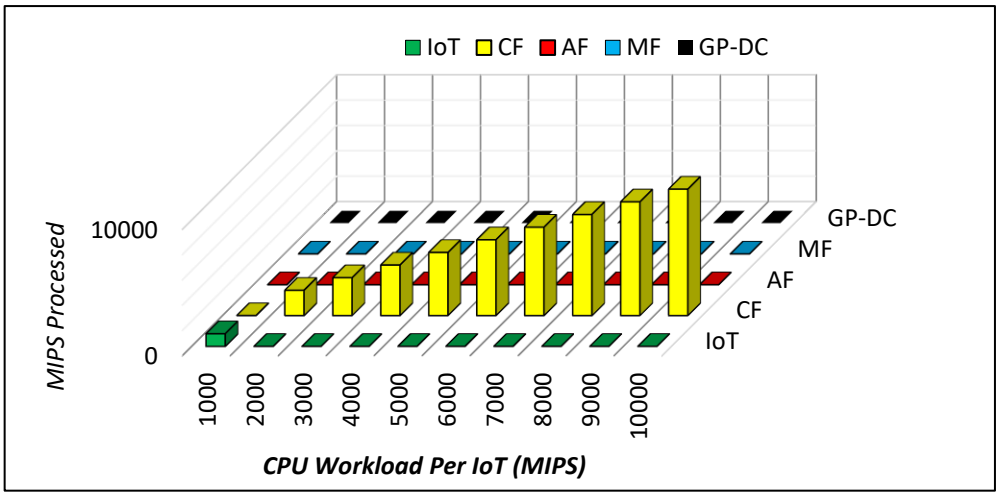


Figure 4.6 Workload Distribution in Scenario #1.

4.7.2 Scenario #2: Five active IoTs in the same group

In this scenario, the number of IoT devices demanding computational resources has increased to five devices residing in the same group and connected to the same CF. The trends in this scenario remain the same as scenario #1, except for workload values of 5000 MIPS and beyond. As can be seen in Figure 4.8, the model decides to allocate all the demands to the metro fog that is connected via the metro network. Although the IoT devices are collocated in the same group and can be allocated to a single CF, the results indicate that activating a large server with high idle power and other associated overheads such as networking and PUE, is still the optimal choice. This can be explained by observing the processing inefficiency of the CF servers. For instance, taking 4000 MIPS as an example, we have 25,000 MIPS in total as there are five IoT devices generating demands. The proportional power consumption for the total MIPS to be processing on CF servers amounts to 109 W alone, compared to the 57W idle power of the more efficient server attached the metro network. Hence, this gives interesting insights about the potential large scale deployments of such servers at the

edge of the network which may not be as energy efficient as larger fog nodes concentrated higher up in the network hierarchy. Although CF servers produce savings of up to 69% for lower ends of the workload, this diminishes as soon as the workload intensity of the services increase. With the demands allocated to the MF node, power consumptions savings of up to 46% can be achieved, compared to the baseline.

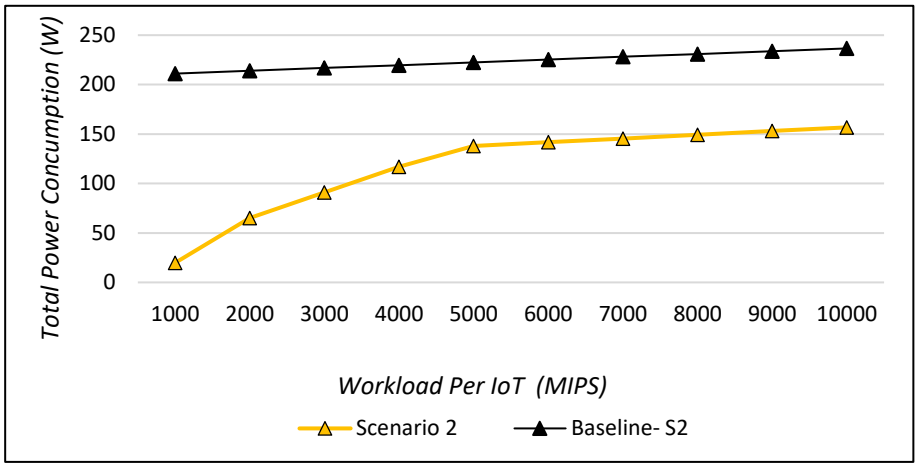


Figure 4.7 Total Power Consumption of Distributed Approach in Scenario #2.

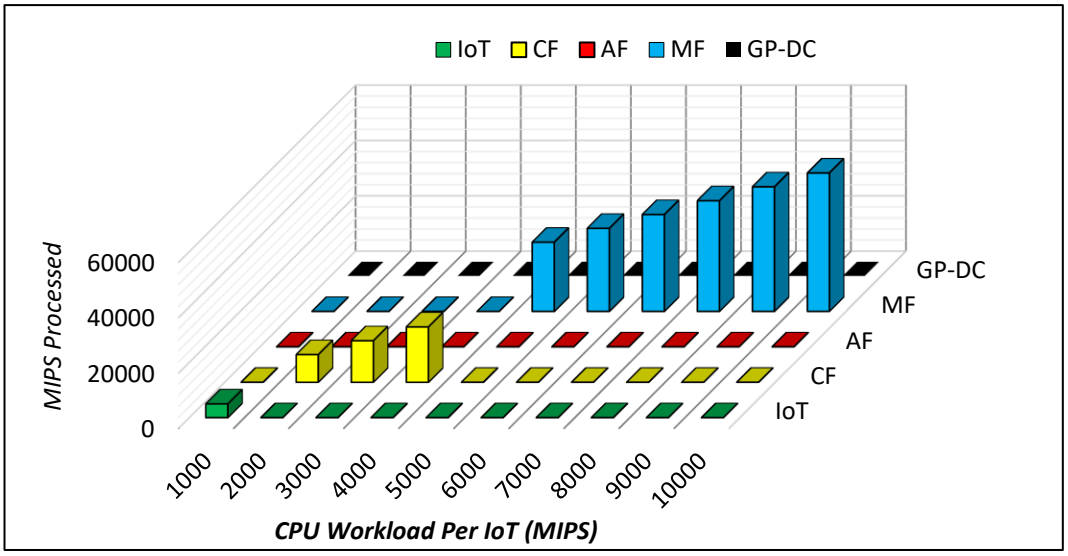


Figure 4.8 Workload Distribution in Scenario #2.

As can be seen in Figure 4.8, the model never utilises the AF server despite its close proximity in terms of distance from the IoT device and the fact that the OLT power consumption is minimal compared to the high capacity Ethernet switch attached to the MF server. The main cause for not choosing to utilise the processing resources of the AF is primarily linked with the high PUE value because the AF and MF have both identical servers in terms of power consumption.

4.7.3 Scenario #3: Four active IoTs, one per group

In this scenario, we aim to investigate the effect the location of the IoT devices has on the optimal allocation of services, hence, each request is connected to a separate network. Interestingly, the trends remain unchanged as in Scenario #2, except that accessing the MF is delayed to the case of 5000 MIPS. This agrees with the explanations provided for Scenario #1, as the results here indicate that, for IoT devices located in different parts of the network, activating additional CF servers at the four different locations coupled with the networking overhead at the CF layer (e.g. 4 ONU devices activated) is still the optimal choice. The distributed approach still produces promising power savings compared to the baseline scenario, as can be seen in Figure 4.9. When all demands are hosted at the CF layer, savings of up to 66% can be achieved whilst this drops down to 39% when the services are allocated to the MF node.

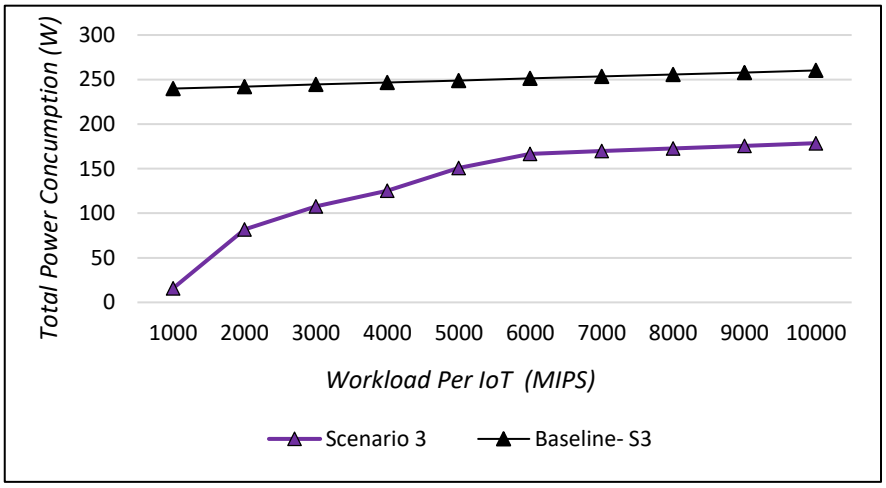


Figure 4.9 Total Power Consumption of Distributed Approach in Scenario #3.

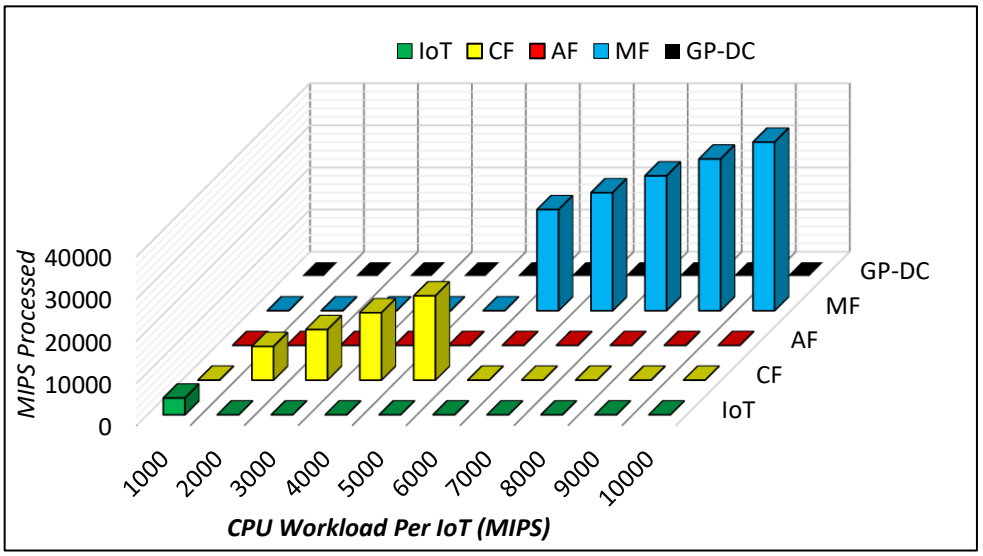


Figure 4.10 Workload Distribution in Scenario #3.

4.7.4 Scenario #4: Twenty active IoTs

In this scenario, we take the other end of the extremes and assume that all of the IoT devices generate requests for resources simultaneously. With the increase in the number of IoT devices, the volume of demands also increases, hence trends are expected to change. As can be seen in Figure 4.11, the distributed processing approach still yields total savings of up to 17% at 7000 MIPS, compared to the baseline. However, when the workload volume

reaches a certain level, e.g. at 5000 MIPS, the model decides to allocate all of the workloads to the centralised cloud data centre and bypass the fog layers all together. This is attributed to the idle power consumption of the GP-DC because at 6000 MIPS and 7000 MIPS the model switches back to the MF as at those particular workload levels, additional servers have to be activated. Hence, the computational efficiency of the GP-DC is traded off with the costly overhead of the network. However, once the workload has increased to 8000 MIPs and beyond, processing everything at the GP-DC proves to be the optimal choice. The only time the CF server is utilised is at workload 4000 MIPS in combination with MF. This is because it saves activating an additional server MF node.

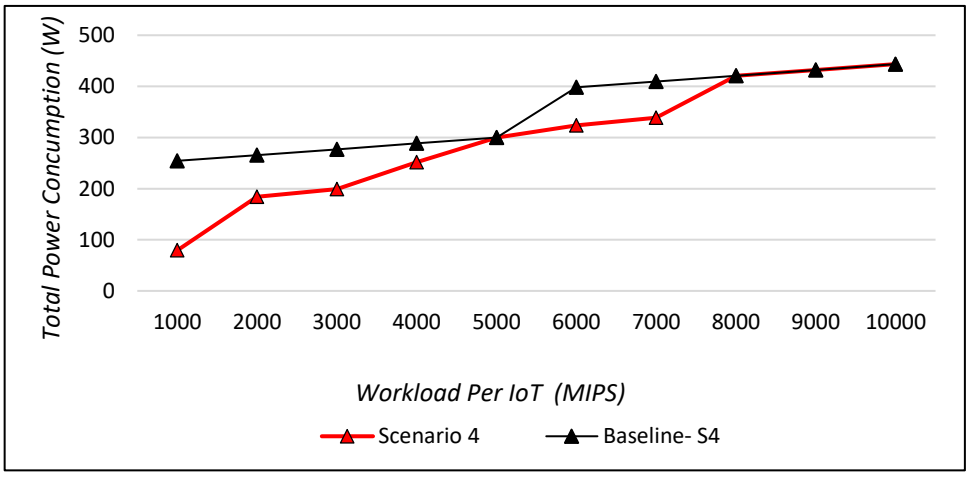


Figure 4.11 Total Power Consumption of Distributed Approach in Scenario #4.

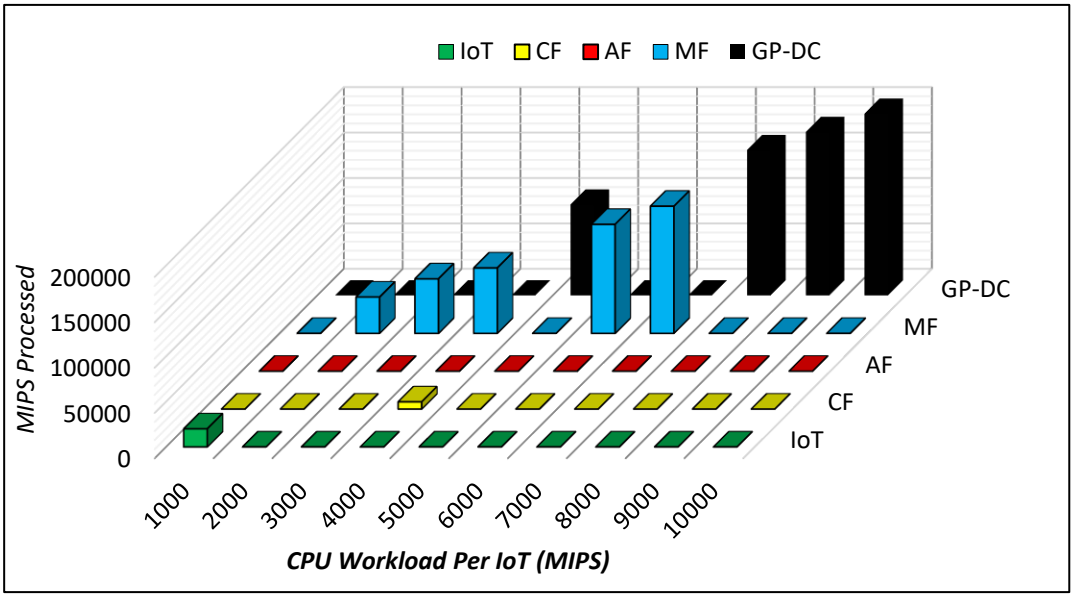


Figure 4.12 Workload Distribution in Scenario #4.

4.8 Capacitated Design Problem with GP-DCs

In this section, we consider the case where extra capacity cannot be added to the processing nodes in question, hence the problem is known as capacitated. Such design problems are faced in the short term when the network and in particular the processing nodes are already designed and are in place.

4.8.1 Scenario #1: *A Single active IoT*

In the capacitated design problem, different trends are expected because the prospect of adding extra processing capacity is no longer the case. As can be seen in Figure 4.14, unlike the trends observed in the scenarios of the un-capacitated problem, the AF server is chosen as the next best choice after the IoT local computation and CF capacities have become violated. We have already observed that the AF server is never a good choice in the un-capacitated case and this is primarily down to the value of the PUE associated

with this node. Although a bad choice, the distributed approach still yield savings with AF as can be seen in Figure 4.14.

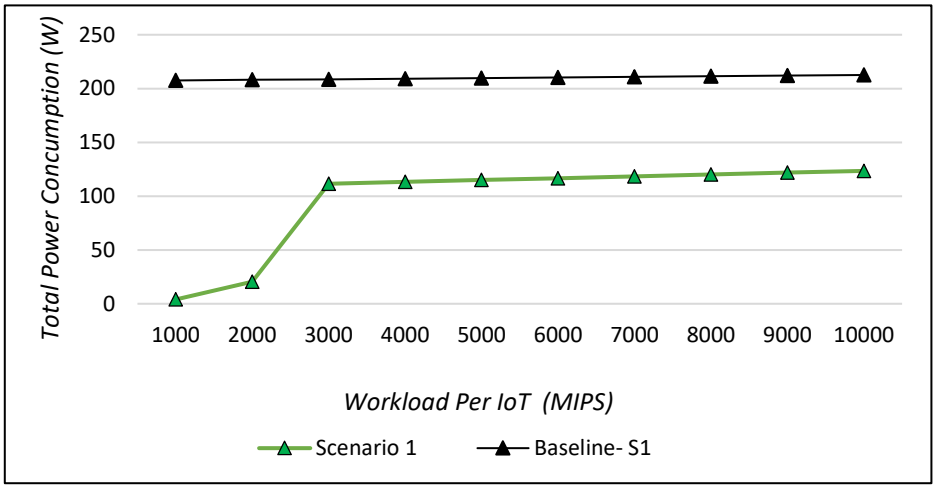


Figure 4.14 Total Power Consumption of Distributed Approach in Scenario #1, in the Capacitated Case.

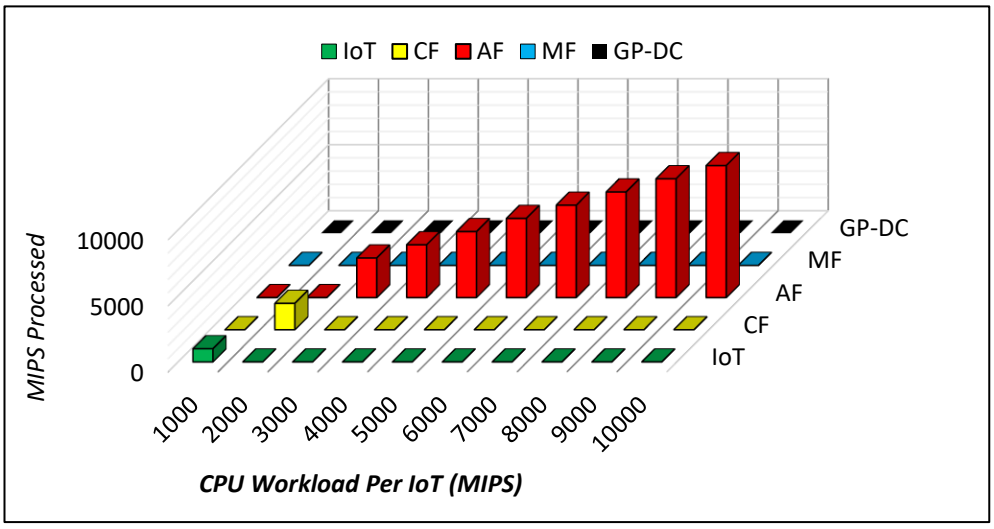


Figure 4.14 Workload Distribution in Scenario #1, in the Capacitated Case.

4.8.2 Scenario #2: Five active IoTs in the same group

In this scenario, we begin to observe the disappearance of the AF node as anticipated due to its lower processing efficiency and higher PUE compared with the MF node, as shown in Figure 4.16. The total power consumption savings drop down to 41% from 69% for workload volumes of 2000 MIPS in the un-capacitated case. This is mainly the difference between hosting the demands in the CF layer compared to the AF layer. As shown in Figure 4.15, still a significant amount of power saving is achieved compared to the baseline solution. Although the CF servers had enough capacity to host 9600 MIPS of the total 10,000 MIPS (2000 MIPS/IoT), the model is forced to consolidate processing at the AF layer due to the service splitting constraint forcing processing to take place in a single location because the AF server would need to intervene anyway to process at least 400 MIPS thus packing a single AF server is the optimal choice in this case. This is consistent with previous observations in the un-capacitated case, for lower workload volumes (i.e. 2000 MIPS), the model tends to serve the demands in the lower layers of the fog such as the AF node primarily due to the level of workload since the processing efficiency of the MF server and its lower PUE does not justify the networking overhead for accessing the MF. However, as the workload increases (i.e. 3000 MIPS and higher), the processing efficiency coupled with the lower PUE of the MF server compensates for the networking overhead, hence MF node is chosen as the optimal location to serve the demands as can be seen in Figure 4.16.

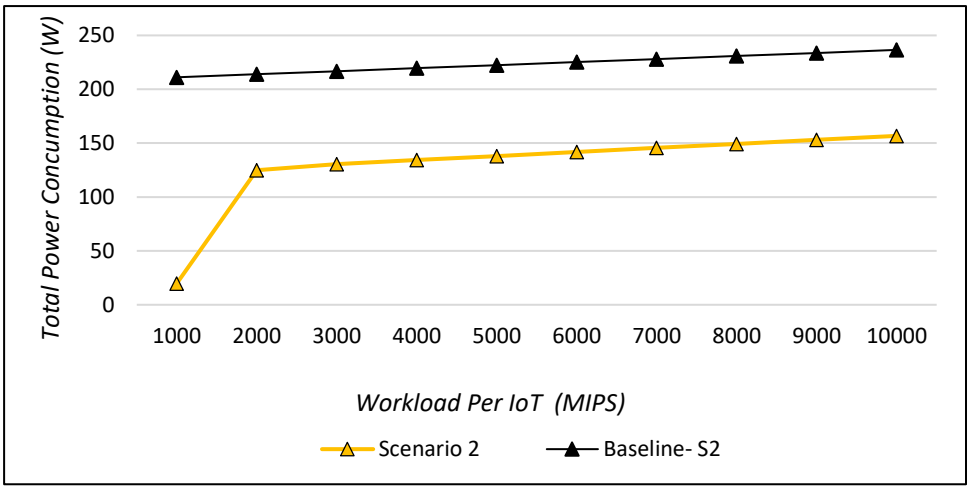


Figure 4.15 Total Power Consumption of Distributed Approach in Scenario #2, in the Capacitated Case.

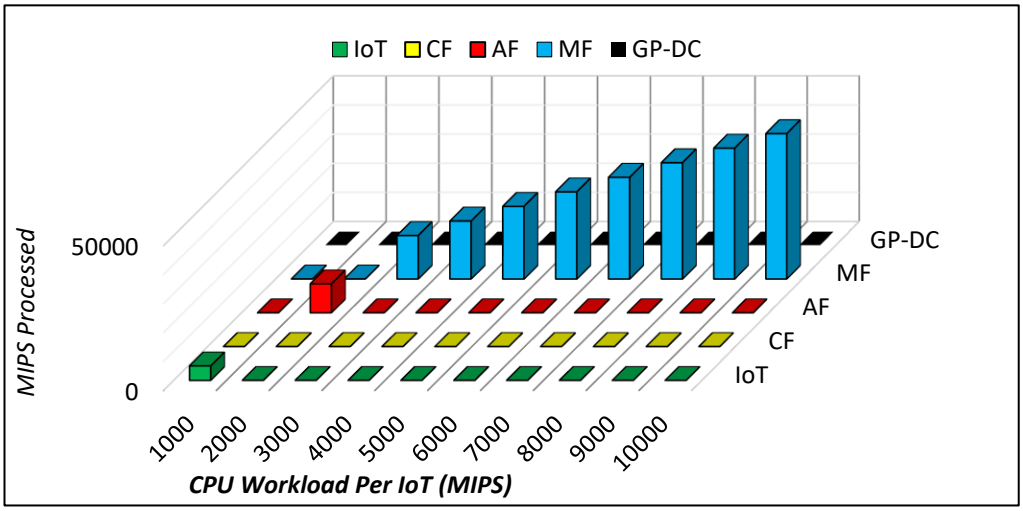


Figure 4.16 Workload Distribution in Scenario #2, in the Capacitated Case.

4.8.3 Scenario #3: *Four active IoTs, one per group*

The trends in this scenario are relatively comparable to Scenario #2, except for the case at 2000 MIPS where instead of the AF server, the CF servers are utilized. This is mainly due to the geographical distribution of the IoT source nodes as in this scenario, each CF server has enough capacity to serve its source node and the number of source nodes happen to match the number of CF servers available, hence the high idle power and associated PUE of the

higher fog layers like the AF and the MF can be avoided in this case, unlike Scenario #2 at 2000 MIPS. A total saving of up to 66% is achieved at 2000 MIPS and up to 55% saving at higher workloads is achieved, as shown in Figure 4.17.

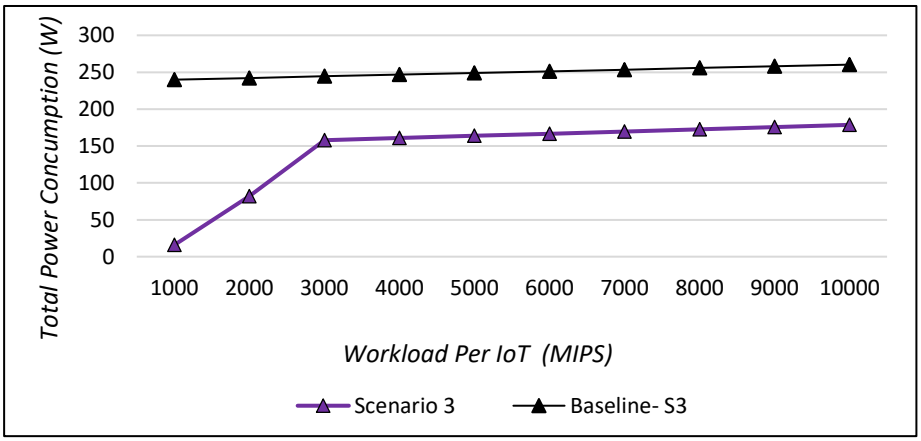


Figure 4.17 Total Power Consumption of Distributed Approach in Scenario #3, in the Capacitated Case.

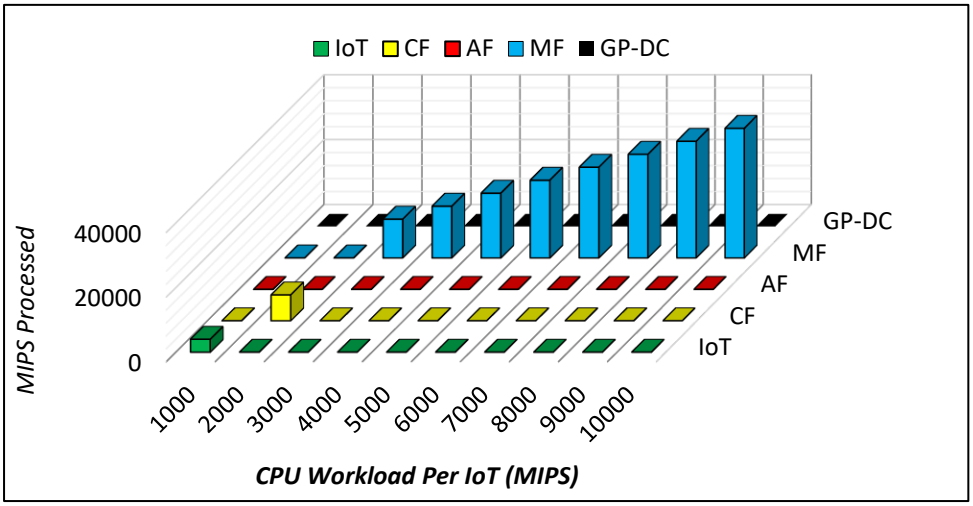


Figure 4.18 Workload Distribution in Scenario #3, in the Capacitated Case.

4.8.4 Scenario #4: *Twenty active IoTs*

In this scenario, we begin to observe the same trends found in scenario #4 in the un-capacitated case except that the intervention of the cloud occurs earlier in this scenario at 4000 MIPS. This result proves the consistency of the model

since the extra capacity needed to host all the demands at 4000 MIPS, requires multiple servers at the MF node, thus it becomes more efficient to migrate all services to the GP-DC to better pack the already activated servers as it is much more efficient and has a better PUE value. As can be seen in Figure 4.19, utilisation of the MF server is only beneficial at certain workload values, otherwise once a certain number of servers are required, the network overhead to get to the GP-DC justifies the activation of the MF server. Figure 4.20 shows that there are still substantial savings (about 17%) at 7000 MIPS despite the activation of multiple servers at the MF and its high PUE, compared to the GP-DC.

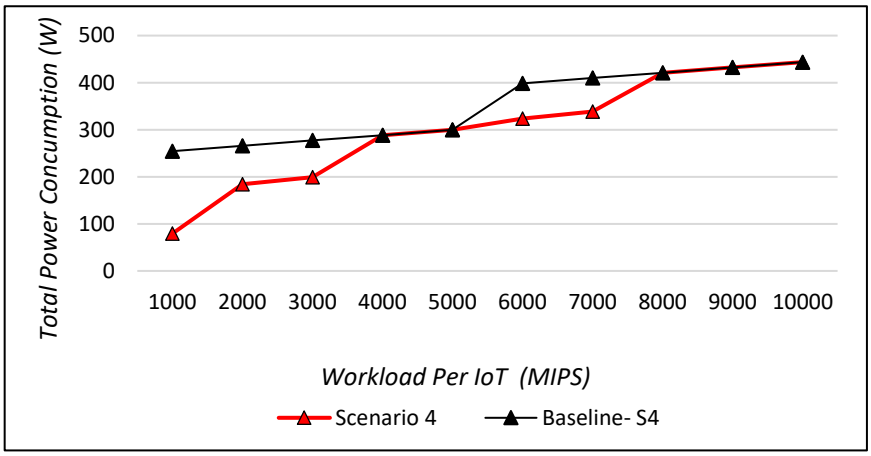


Figure 4.20 Total Power Consumption of Distributed Approach in Scenario #4, in the Capacitated Case.

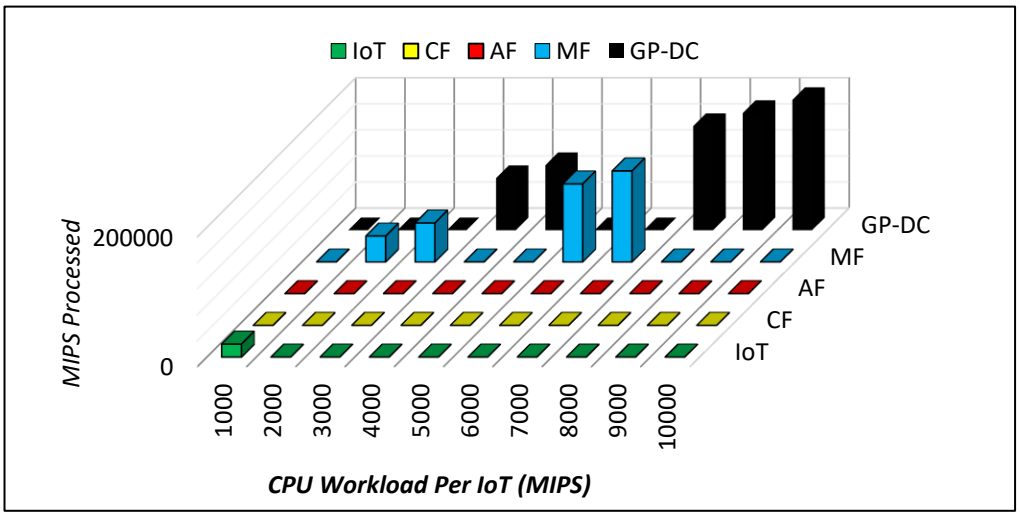


Figure 4.19 Workload Distribution in Scenario #4, in the Capacitated Case.

4.9 Impact of the SP-DC in all Cases

Given the high energy efficiency of the SP-DC servers, it was worth investigating its impact on improving the energy efficiency of the proposed distributed processing model. The results indicated that all the trends in both the capacitated and un-capacitated cases and from scenario #1 to scenario #3, remained unchanged. However, at scenario #4, different trends were observed when a highly efficient server like the SP-DC is deployed. The impact of the SP-DC is observed at and beyond 4000 MIPS. Interestingly, as shown in Figure 4.21, the SP-DC yields total savings of up to 50%, whilst the maximum saving obtained in the same scenario was up to 30% even in the un-capacitated case where multiple CF servers could be deployed. This is a promising performance from the SP-DC and these results demonstrate that for scenarios where the computational workload is extremely high, deploying mini DCs in the fog layers that are associated with high PUEs and are less efficient in terms of processing per instruction, hosting services on them brings no benefits when a highly efficient centralised DC is available at the core network.

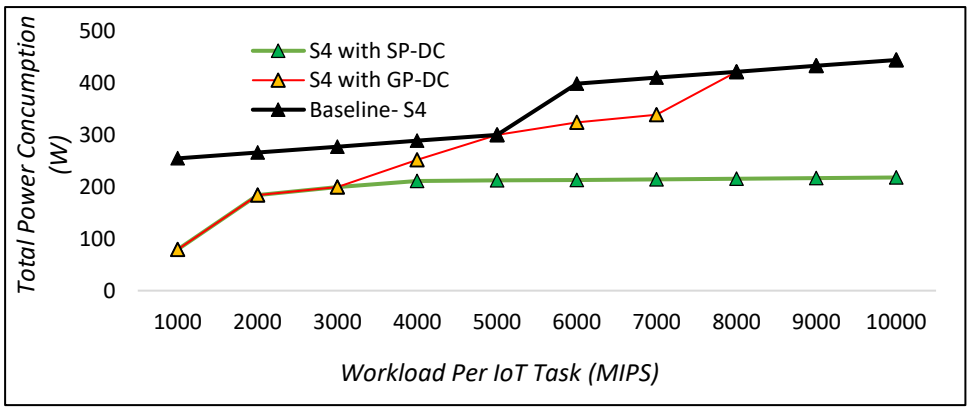


Figure 4.21 Total Power Consumption of Distributed Approach in Scenario #4, with and without SP-DC.

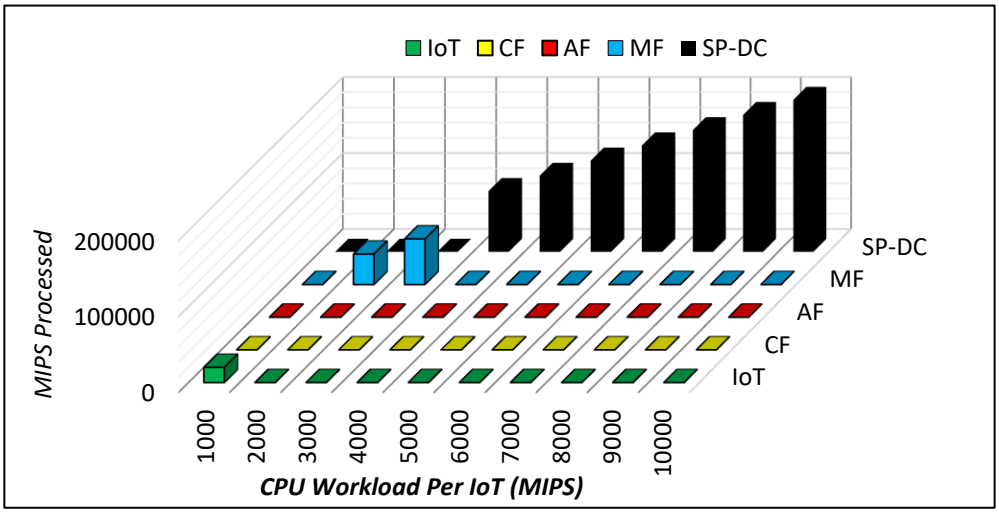


Figure 4.22 Workload Distribution in Scenario #4, when SP-DC is Deployed.

4.10 MILP Model Verification

This subsection considers boundary cases whose optimum server placement solution is known and hence provides verification of the MILP through simple analytic closed-form expressions. This is used to confirm with confidence the MILP results obtained in this thesis are valid. The current model forms the basis of the works in the subsequent chapters, hence it is sufficient and crucial to check this model. The checkpoints consist of various scenarios such as the case where all the services are processed by (i) the cloud (GP-DC), (ii) the access fog (AF), (iii) the metro fog (MF) and (iv) finally the IoT devices. Although not exhaustive, however, the aforementioned checkpoints ensure that all the network and server elements between the IoT and the cloud are verified in terms of networking and processing power consumption. The total power consumptions are evaluated by referring to appropriate figures in the power consumption evaluation subsection. The verifications are summarised in Table 7,

Table 8, and Table 9. In each table, reference is made to the scenario that is being evaluated. The column headings represent the power consumption at each layer of the proposed architecture while the row headings consist of network, processing and total power consumption. It is important to note that, the proportional power consumption of the network is ignored (since it's negligible compared to the idle power consumption of the networking equipment (and the servers processing power consumption)), hence the analytic checking calculations only account for idle network powers and processing power consumption. Also, for the sake of simplicity, in the processing column, the first value is the network power consumption of the processing node, the second is the server idle power and the third is the proportional power per instruction. All figures are calculated after PUE.

Checkpoint 1							
Capacitated, Scenario #4, All processing at Metro Fog (MF) at 5000 MIPS per IoT, 2 idle servers.							
Layer	P^{IoT} **(W)**	P^{CPE} **(W)**	P^{AF} **(W)**	P^{MF} **(W)**	P^{core} **(W)**	P^{DC} **(W)**	**Total (W)**
Network Power	6.8	36	3.6	17.76	-	-	64.16
Processing Power	-	-	-	7.98 + (2*79.8)+ (72.38 =239.96	-	-	239.96
Total Power	**304.1 W**						
Capacitated, Scenario #4, All processing at Cloud DC (GP-DC) at 5000 MIPS per IoT, 1 idle server.							
Network Power	-	-	-	2.2	11.58 + 5.1+3.4+51 =**71.08**	-	64.16+71.08=135.24
Processing Power	-	-	-	15.12 + 87.36 + 53.87	-	-	156.2
Total Power	**291.5 W**						
MILP Result	≅ 299 ➔ refer to Figure 4.20.						

Table 7 Analytic Verification of the Optimal Choice in Scenario #4 at 5000 MIPS.

Table 7 is a summary of the verification checks done for Scenario #4 at a workload of 5000 MIPS per IoT. The results confirm that the optimal choice corroborates with that of the MILP as can be seen in Figure 4.20. In Table 7, the total power consumption was 291.5 W for the approximate analytic calculations versus 299 W for the accurate MILP optimisation which includes power consumption components ignored by the analytic approximation and its calculations. It is also worth noting that the choice of Metro fog to process the tasks results in higher power consumption (304.1W), and hence the metro fog is correctly not chosen by the MILP to place the processing tasks.

Checkpoint 3							
Capacitated, Scenario #2, All processing at Access Fog (AF) at 2000 MIPS per IoT, 1 idle server.							
Layer	P^{IoT} **(W)**	P^{CPE} **(W)**	P^{AF} **(W)**	P^{MF} **(W)**	P^{core} **(W)**	P^{DC} **(W)**	**Total (W)**
Network Power	1.7	9	2.7	-	-	-	13.4
Processing Power	-	-	-	8.55+85.5+15.5	-	-	109.55
Total Power	**122.95 W**						
Capacitated, Scenario #2, All processing at Metro Fog (MF) at 2000 MIPS per IoT, 1 idle server.							
Network Power	-	-	-	17.76	-	-	13.4+ 17.76= 31.16
Processing Power	-	-	-	7.98+79.8+7.23	-	-	95
Total Power	**126 W**						
MILP Result	≅ 125 ➔ refer to Figure 4.15						

Table 8 Analytic Verification of the Optimal Choice in Scenario #2 at 5000 MIPS.

Table 8 also confirms the optimal choice of the MILP at 2000 MIPS, when all processing is done at the AF and MF separately. The optimal case is where processing is done at the OLT and the total power consumption is calculated as 122.95W. In the MILP this figure is roughly 125W as can be seen in Figure 4.15. Table 9 is the final checkpoint that examines local processing at the IoT layer for Scenario #4, with similar verification conclusions.

Checkpoint 4							
Capacitated, Scenario #4, All processing at IoT at 1000 MIPS per IoT, 20 idle servers.							
Layer	P^{IoT} **(W)**	P^{CPE} **(W)**	P^{AF} **(W)**	P^{MF} **(W)**	P^{core} **(W)**	P^{DC} **(W)**	**Total (W)**
Network Power	-	-	-	-	-	-	-
Processing Power	$10 + (20000\ MIPS \times 3460\mu W)$	-	-	-	-	-	79.2
Total Power	**79.2 W**						
MILP Result	= **79.2** ➔ refer to Figure 4.21.						

Table 9 Analytic Verification of the Optimal Choice in Scenario #4 at 1000 MIPS.

4.11 Summary

Energy efficiency has become a key factor in the design and implementation of communication networks in general. This chapter has evaluated distributed processing based on the paradigm of fog computing, in the context of IoT A MILP model has been developed with the objective of minimising the total power consumption (networking and processing) by optimal placement of IoT services. Un-capacitated and capacitated design problems were considered for a range of scenarios that consisted of different numbers of active IoT source nodes which reflected the extreme cases and the moderate ones in between. It has been demonstrated by results that, in most cases, distributed processing helps to improve energy efficiency by processing data in close proximity to the source nodes and hence substantial savings are made. Moreover, two types of data centre servers were examined (GP and SP). The results indicated that in scenarios where the workload volume is low, having a SP-DC in the core network made no difference, as the model would always prefer to allocate services to the fog nodes.

However, when the number of active IoTs increase and subsequently the workload volume increased (scenario #4), the results indicated that deploying SP-DCs could save up to 50% compared to 30%. It became apparent by the results that bringing local computation such as the IoT devices, produce substantial savings, however, due to capacity limitations this could not have been achieved. In this direction, this chapter motivates the basis for investigating the impact of service splitting on improving energy efficiency in the next chapter. Finally, the chapter was concluded by a section of analytic verification of the considered MILP model.

Chapter 5 Energy Efficient Distributed Processing for IoT with Service Splitting

5.1 Introduction

Future IoT services will consist of multiple components, coordinating and communicating over the network to achieve a common task, similar to applications design in Service-Oriented Architectures (SOA) [3]. Extensive research has been carried out to tackle the problem of abstracting workflow of multiple services associated with end devices to provide a proper architecture that incorporates service management and composition capabilities. Each IoT device holds a limited amount of computational resources, given the scale of IoT, each device may be called on to provide a variety of services. In this direction, inspired by the work in [72], this chapter evaluates a scenario in which, service tasks can be split into multiple subtasks, hence multiple processing nodes can be utilised to complete a single service [128].

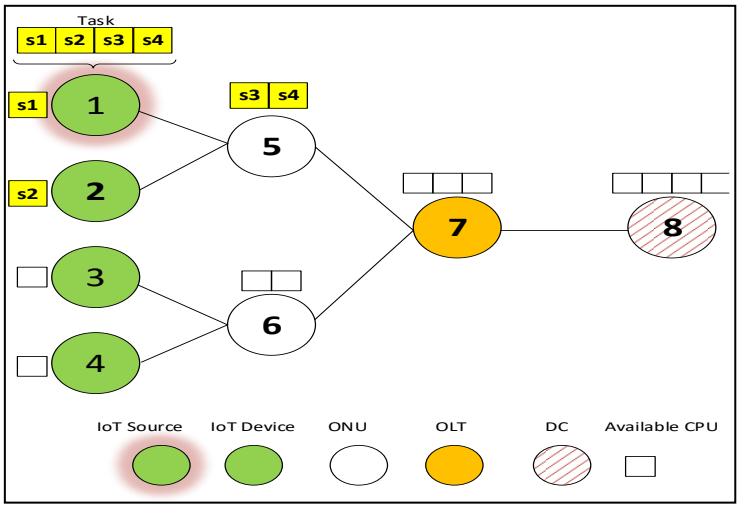


Figure 5.1 An illustrative example of service splitting

Figure 5.1 is a simple illustrative example of service splitting in the considered architecture. An IoT source node has a task for processing that can be subdivided into multiple components (s1-s4). The main goal of this chapter is to determine in cases where IoT devices' available CPU capacity is not sufficient to process a task, whether service splitting among numerous processing nodes becomes beneficial in terms of total power reduction, given the added power consumption associated with network traffic after splitting, especially when network equipment have idle power consumption. In our evaluations, service splitting is only said to have occurred if different subtasks of an application service are processed in geographically distributed servers, otherwise, if subtasks are all processed on the same processing node, then this is not classed as service splitting mainly due to the same network latency for the subtasks.

5.2 Modification to the MILP Model

The MILP model in Chapter 4 remains intact in the current chapter, except for a minimal modification to the processing location constraint (4.24) in Chapter 4, in order adapt to the variation introduced by service splitting. Previously, it was assumed that the parameter $K = 1$, in the current chapter K will adopt values from 1 – 5 to investigate a range of service splitting scenarios and their impact on the reduction of the total power consumption of the distributed processing approach. Therefore we introduce the constraint:

$$\sum_{d \in P} \Omega^{sd} \leq K \qquad (5.1)$$

Constraint (5.1) remains unchanged, except that, the value of K is increased from 1 to 5 for each iteration of the MILP model.

5.3 Power Consumption Evaluation

Similar to chapter 4, we consider the same scenarios to gauge the impact of service splitting on improving the total power consumption.

5.4 Un-Capacitated Design Problem with GP-DCs Only

5.4.1 Scenario #1: *A single Active IoT*

In this scenario, the results obtained show early indications that during low workloads, service splitting introduces comparable savings to the case where $K = 1$, as can be seen in Figure 5.2. Interestingly, for all workload volumes at K = 3, the model still decides to always at least fully utilise the IoT source node as can be seen in Figure 5.3. This is understandable as local computation incurs zero network overhead plus the low idle power of the IoT node compared to the CF server makes it always beneficial to utilise in this case as it can avoid the activation of further CF servers which are associated with high idle power.

For example, the case where K = 3 and the workload 4000 MIPS, the model does the service splitting in the ratio of 2:2. This is only done due to the restriction enforced by the value of K, otherwise, it would save more power to split the 4000 MIPS among the other IoT devices that are in the same group as the source node. This is confirmed in the scenario where the value of K was increased to 5. It is shown that splitting the 4000 MIPS in the ratio of 4:0 is the best choice and hence the CF server is never utilised in this case due to the high idle power of the ONUs and the CF servers. The general trend in this scenario as can be observed in Figure 5.3, K=5 at 5000 MIPS, the optimal choice for the IoT source node is to process 1000 MIPS locally and offload the

remaining 4000 MIPS to the peer IoT nodes since up to five service splits can be performed, confirming the inefficiency of the CF servers due to high idle power of ONU.

However, when the workload is increased to 6000 MIPS for the same scenario, we can observe that 4000 MIPS are kept on the IoT layer while the remaining 2000 MIPS are processed on the CF servers. At this instance, the CF server introduces savings and is not related to the value of K although the model is restricted by it, however, a single IoT group can provide an aggregate 5000 MIPS and if the remaining 1000 MIPS was to be processed on another IoT from another group, then the demand must be offloaded via the OLT to another ONU of that group, hence, instead, it would introduce more savings to pack the CF servers connected to the source node group. It becomes clear that when the total workload is within the aggregate capacity of the IoT devices located in the same group as the source node, making use of service splitting improves power efficiencies due to the very low idle power of the IoT devices.

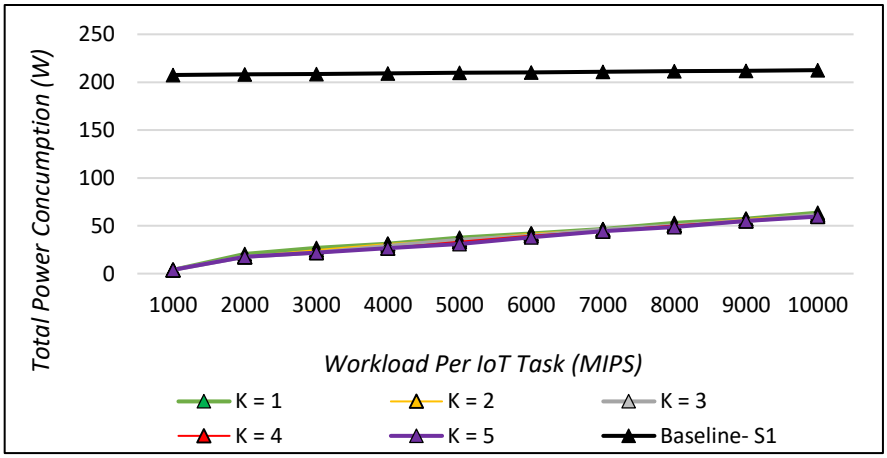


Figure 5.2 Total Power Consumption of the Distributed Approach at Various Values of K.

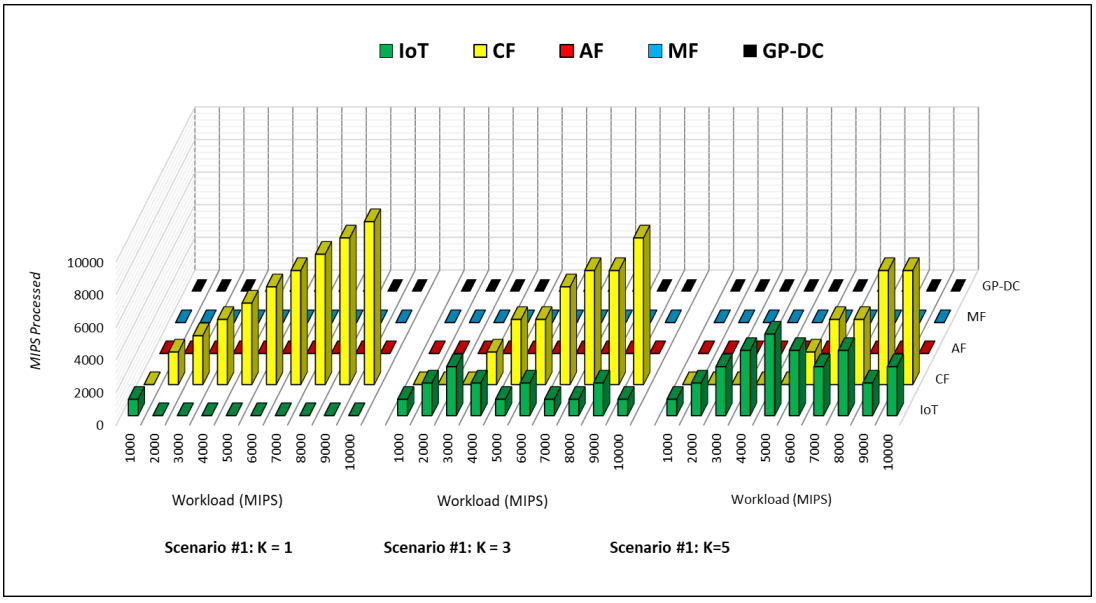


Figure 5.3 Workload Distribution in Scenario #1 at Different Values of K.

5.4.2 Scenario #2: *Five active IoTs in the same group*

In this scenario, the number of IoT source nodes has increased to 5 which is the total number of IoT devices in one group. The savings of service splitting are still comparable to the case where K=1, as can be seen in Figure 5.4. The maximum savings of service splitting compared to K=1 does not exceed 2.73%. The trends in this scenario have changed compared to scenario #1. This is primarily due to the increase in the number of source nodes and subsequently the volume of demand. In Figure 5.5, it can be observed that service splitting (i.e. K=3) only becomes relevant for the lower end of the workload such as at 2000 MIPS to 5000 MIPS. This can be explained by noting that service splitting among the IoT device(s) and multiple CF server connected to the ONU is only beneficial if the volume of workload is below a certain threshold, in this case, 25000 MIPS in total which is 5000 MIPS per IoT source node. This can be confirmed by the case where the workload increases. Once the workload is increased to 6000 MIPS per IoT (30,000 in total), it can be confirmed in Figure 5.5, that service splitting is irrelevant

mainly due to the processing inefficiency of the CF servers and IoT devices. For instance, the proportional power consumption to process just the 30,000 MIPS and this would result in 131.3 W power consumption had this workload been processed on the CF servers that are a single hop from the IoT devices. This compared to the high idle power of the MF server which is 109W after PUE, still makes the MF server more efficient than splitting the demand between the IoTs and the CF server.

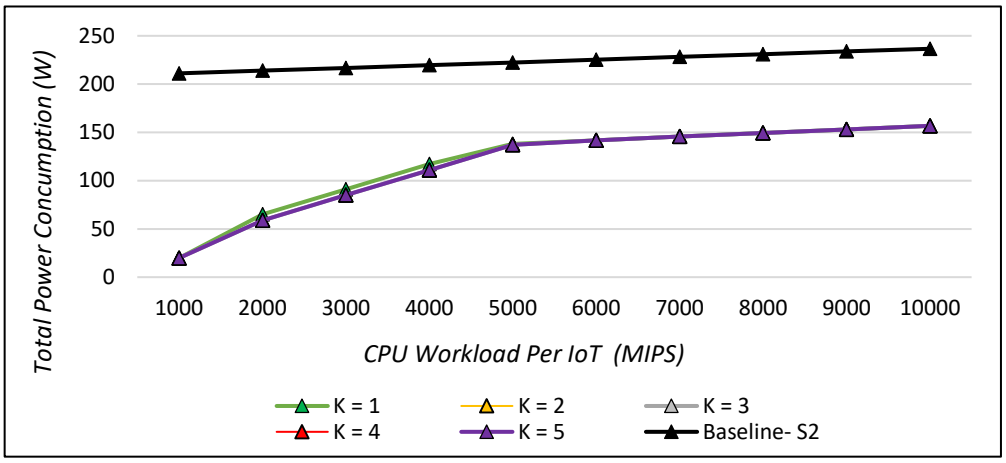


Figure 5.4 Total Power Consumption of the Distributed Approach at Various Values of K.

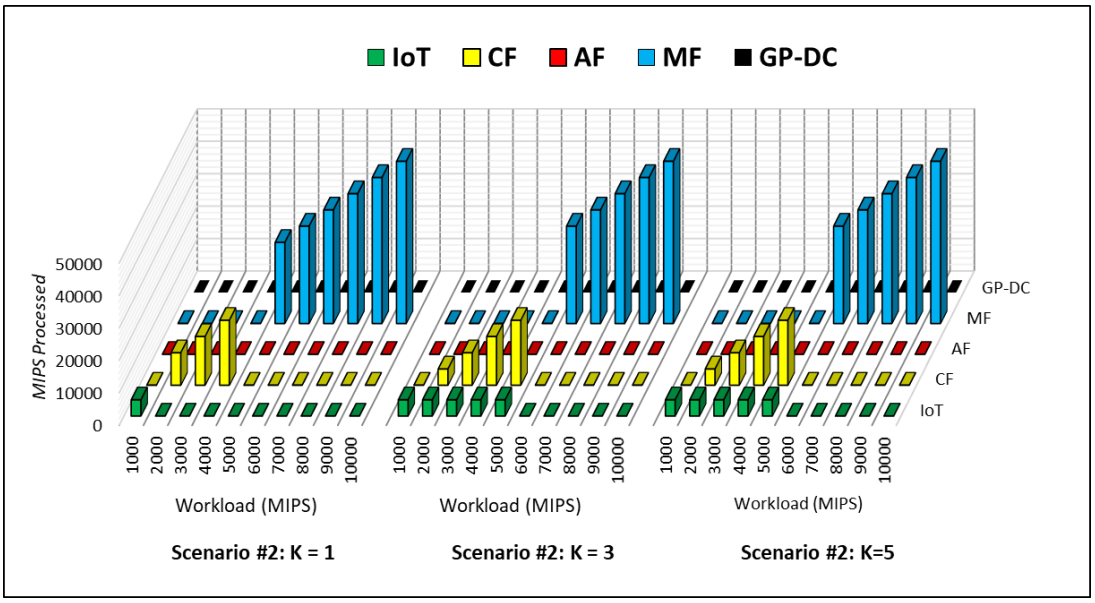


Figure 5.5 Workload Distribution in Scenario #2 at Different Values of K.

5.4.3 Scenario #3: *Four active IoTs, one per group*

It is worthy of mention that in this scenario, the four active IoT nodes are located in different parts of the network such that each one is connected to a different ONU. The model in this instance makes use of service splitting (K =5) as it introduces up to 18% savings compared with no service splitting (K=1), as highlighted in Figure 5.6 at the workload 5000 MIPS. This is understandable because CF servers have 4x times more idle power consumption than their IoT counterparts, hence, making use of the maximum service splitting value K=5 produces more savings compared with the case where K=1 at 5000 MIPS. Moreover, it can be seen in Figure 5.7, at 6000 MIPS, service splitting is only utilised to allow the CF server intervene since the total demand is 24,000 MIPS (4×6000MIPS) compared to the 20,000 available capacity offered by all the 20 IoT devices. We begin to notice that service splitting becomes irrelevant when the total workload has increased i.e. at 6000 MIPS, the metro fog (MF) is utilised to host all the workload. This is primarily due to the processing inefficiencies of the small IoT type devices CPE fog (CF) servers as activating a single MF server with a much better processing efficiency outperforms many IoT and CF servers activated together.

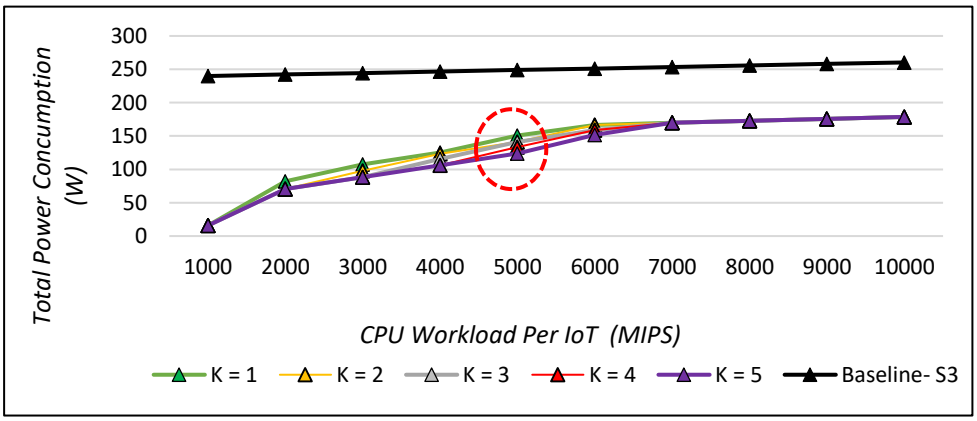


Figure 5.6 Total Power Consumption of the Distributed Approach at Various Values of K.

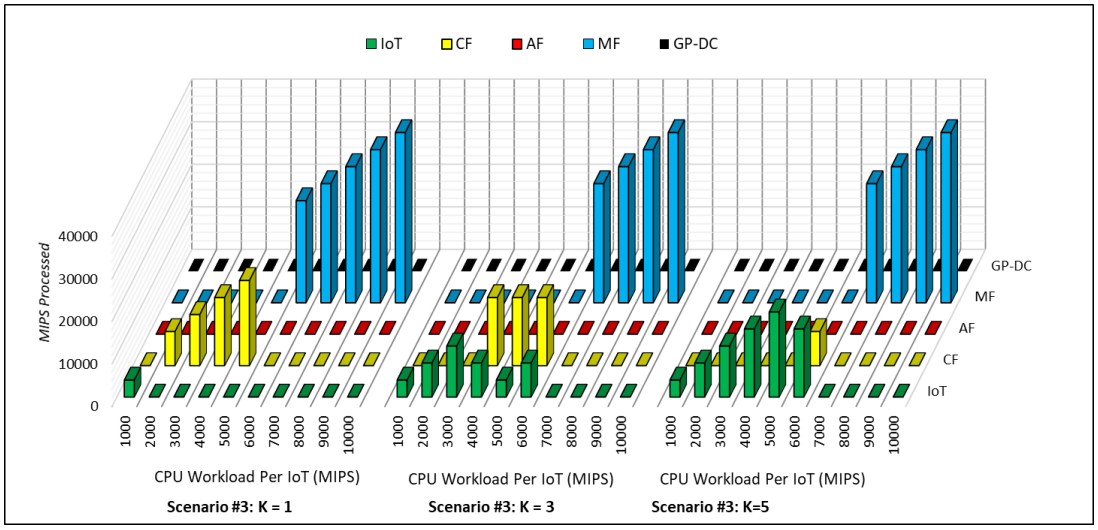


Figure 5.7 Workload Distribution in Scenario #3 at Different Values of K.

5.4.4 Scenario #4: Twenty active IoTs

This scenario offers additional insights on whether service splitting improves the performance of the proposed distributed processing approach. In the previous scenarios, the intervention of the cloud (GP-DC) was always avoided. However as shown in the workload distribution in Figure 5.9, the cloud has come into the picture primarily due to the processing inefficiency of the metro fog server compared to its cloud counterpart. If we take as an example the case where the demand per IoT is 5000MIPS (100,000 MIPS in total) for all three cases K=1, K=3 and K=5, the model chooses to avoid service splitting and instead the workload is offloaded to the cloud altogether, bypassing the metro fog server. This is interesting because the processing efficiency of the cloud DC compensates for the additional power overhead of the core network and the network within the cloud. However, if services could be split to avoid the activation of multiple servers in the metro fog, then the cloud is also avoided in this scenario. For instance, at 8000 MIPS when K=1,

the workload is wholly processed in the cloud whilst for the same scenario where K = 3 or K = 5, the model fully packs a single server in the metro fog and splits the remaining workload among the IoT devices, hence as has been highlighted in Figure 5.8, total savings of up to 6% was achieved by service splitting compared to the case where service splitting is not an option (K = 1).

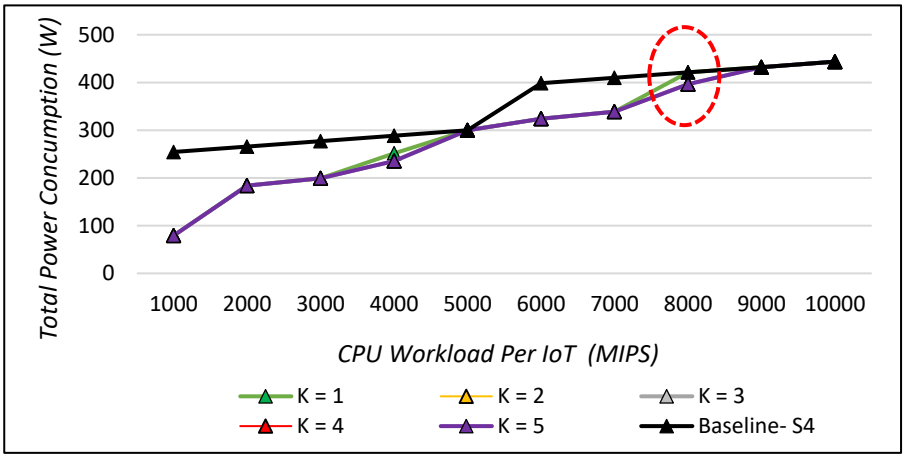


Figure 5.8 Total Power Consumption of the Distributed Approach at Various Values of K.

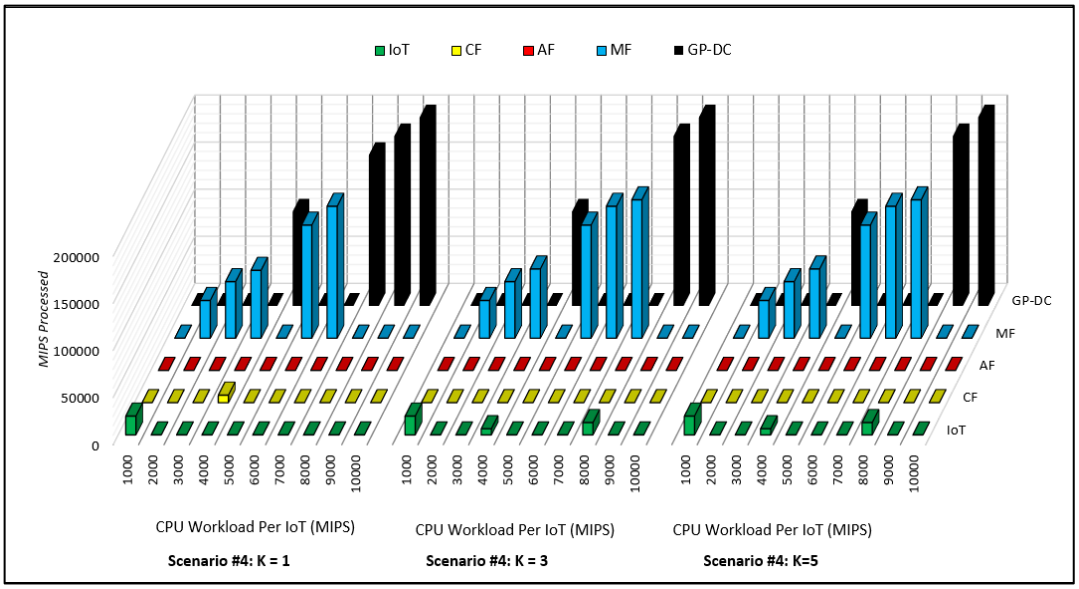


Figure 5.9 Workload Distribution in Scenario #4 at Different Values of K.

5.5 Capacitated Design Problem with GP-DCs Only

In this section, it is of interest to investigate whether service splitting in the short term network design, introduces additional savings on top of the distributed processing approach. The following subsections comprise of the same scenarios that were also considered previously. This way we can make fair comparisons between the various observations in the results.

5.5.1 Scenario #1: *A single active IoT*

Figure 5.10 shows, in the case where processing nodes are limited by capacity, with the increase in the number of service splits (i.e. K>1), substantial savings can be made as opposed to the case with no service splits (K=1). The savings are due to the fact that the access fog's server idle power is avoided since application services will be processed locally between the IoT devices and the CPE fogs despite the network overhead incurred in getting access to these devices. The total savings achieved by the distributed processing approach with non-splittable services (K=1) was up to 46% compared to the baseline, however, this figure increased to 86% when the value of K was changed to 2 (i.e. K=2), as highlighted in Figure 5.10.

Moreover, when the workload volume increases to 10,000 MIPS at K = 5, we begin to see a drastic drop in savings as can be seen in Figure 5.10. The savings due to service splitting dropped from 86% to only 7%. This can be understood by noting that at 10,000 MIPS, 4 CPE fog servers are activated to process 9000 MIPS whilst the remaining 1000 MIPS is processed at the IoT source node itself. Although there were 600 MIPS left for processing on the

CPE fog due to the power consumption of the fog switch and the processing inefficiency, the model fully packed the IoT source node instead.

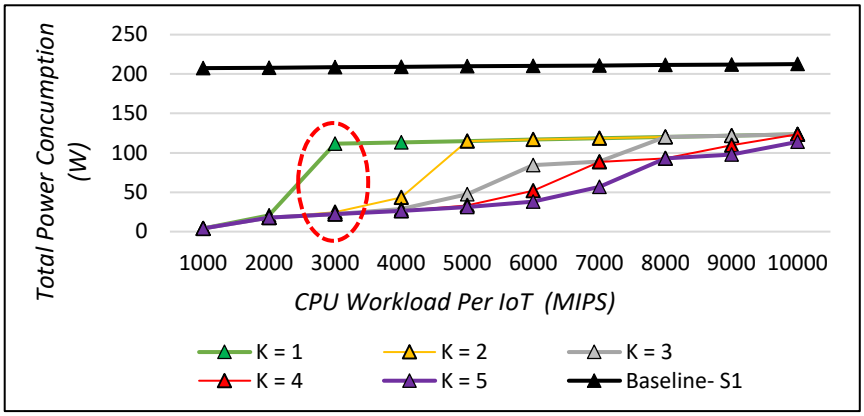


Figure 5.10 Total Power Consumption of the Distributed Approach at Various Values of K.

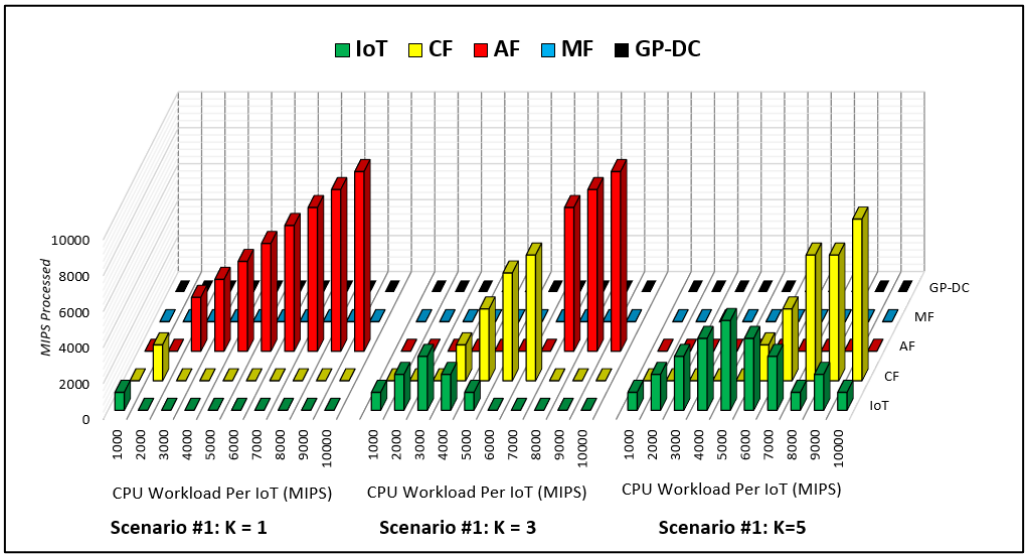


Figure 5.11 Workload Distribution in Scenario #1 at Different Values of K.

5.5.2 Scenario #2: Five active IoTs in the same group

In this scenario, the power savings introduced by service splitting is very limited as shown in Figure 5.12. This is largely due to the capacity limitations placed on the CPE fog coupled with the inflexibility posed by the restriction of service splitting. For example at 4000 MIPS, the total demand is 20,000,

although the IoT devices' capacity in total can accommodate the total workload however this would mean the value of K to be increased to 12, provided that 9000 MIPS was host at the CPE fogs and the remaining 11,000 MIPS was subdivided among the IoTs, hence K = 12.

Interestingly, as shown in Figure 5.13, after the total capacity of the IoT source nodes' group is depleted (4000 MIPS and beyond), the case for service splitting becomes irrelevant as the model always allocates the workload to the metro fog server as activating multiple CPE fogs would incur high costs due to high power consumption of ONU devices.

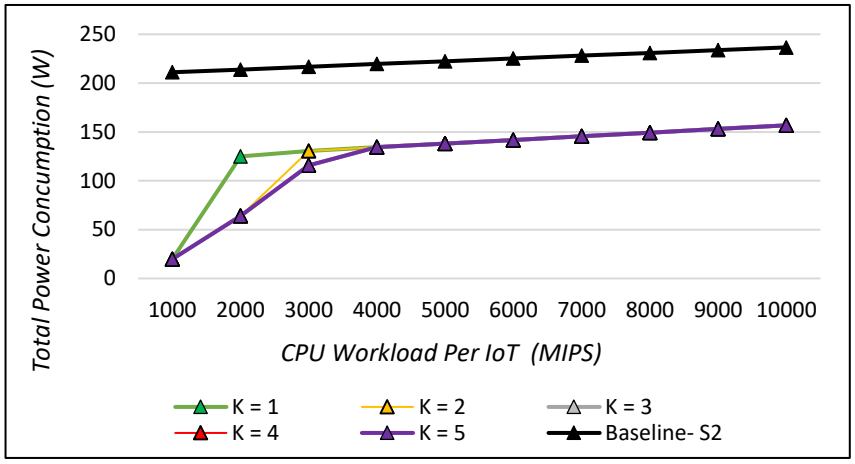


Figure 5.12 Total Power Consumption of the Distributed Approach at Various Values of K.

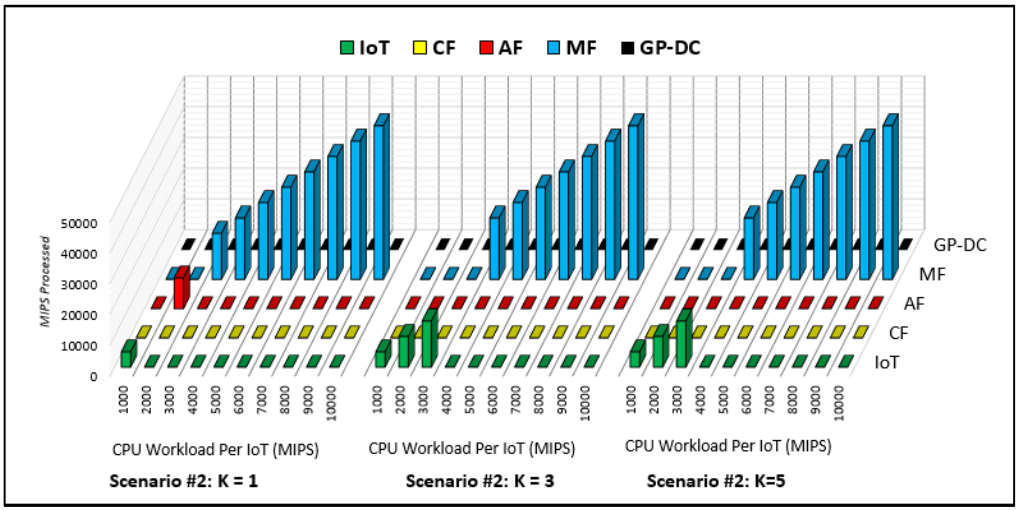


Figure 5.13 Workload Distribution in Scenario #2 at Different Values of K.

5.5.3 Scenario #3: *Four active IoTs, one per group*

The trends in this scenario remain relatively unchanged except for the fact that service splitting is utilised only because the IoT source nodes are from different groups, and the ONU devices would need to be turned on anyway to get to the metro fog, hence CPE fogs attached to the ONUs are used due to their low idle power compared to the metro fog server. This observation was established in the previous scenario in Figure 5.13 at 4000 MIPS, where only the metro fog server was used, compared to 4000 MIPS in this scenario where the workload is processed between the IoT nodes and CPE fogs. In this scenario, a total saving of 56% was achieved with service splitting value K >3 as opposed to 33% with no service splits K = 1, as highlighted in Figure 5.14. As mentioned previously, this large saving is the difference between the idle power of the metro fog server and the smaller devices like the ONUs and the CPE fogs. However, we have already established that, when the workload is increased, the processing per instruction at the metro fog compensates for the idle power of its server, hence all workloads are processed at the metro fog as can be seen in Figure 5.14.

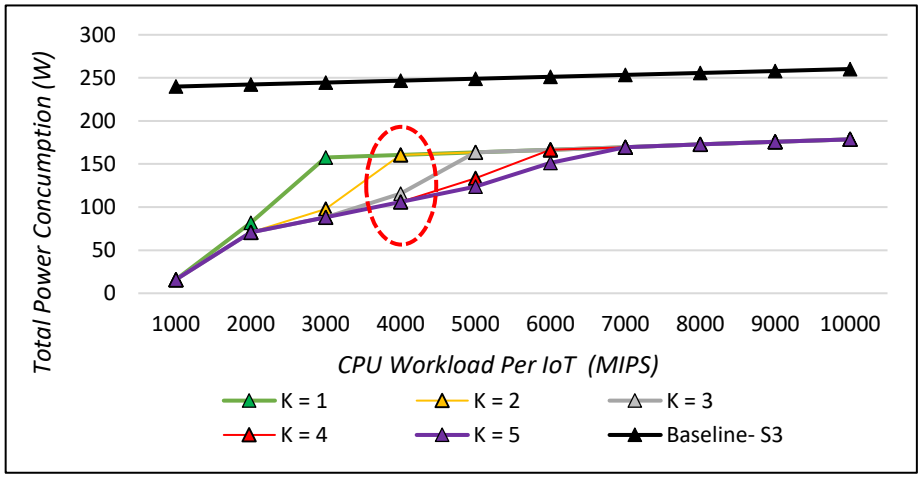


Figure 5.14 Total Power Consumption of the Distributed Approach at Various Values of K.

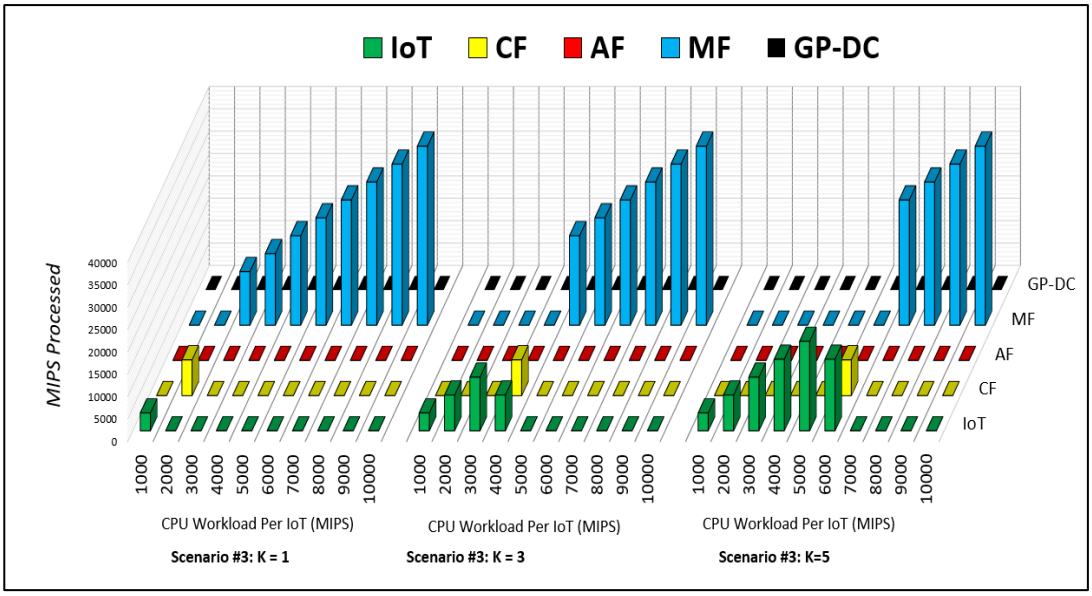


Figure 5.15 Workload Distribution in Scenario #3 at Different Values of K.

5.5.4 Scenario #4: *Twenty active IoTs*

In this scenario, similar to the case K = 1, service splitting is predominantly irrelevant, except in rare circumstances such the scenario at 4000 MIPS at K>1, a total of 80,000 MIPS is demanded by the source nodes and if all of this was to be processed on the metro fog, it would require two servers, hence, in this case, the metro fog server is fully packed and the remaining workload (6560 MIPS) is processed on source nodes' local CPUs. Thus, as shown in Figure 5.16, service splitting at K>1 introduces total savings of up to 18% compared to 0% with no service splitting (K = 1) as the solution was the same as the baseline in this instance. Similar to the observations obtained in the un-capacitated case, the metro fog and the cloud are largely the best choices, respectively, when the workload is too high.

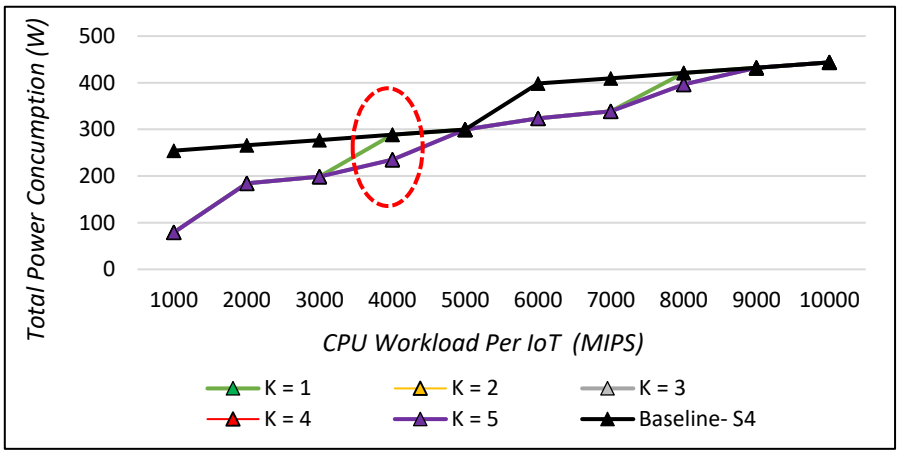


Figure 5.16 Total Power Consumption of the Distributed Approach at Various Values of K.

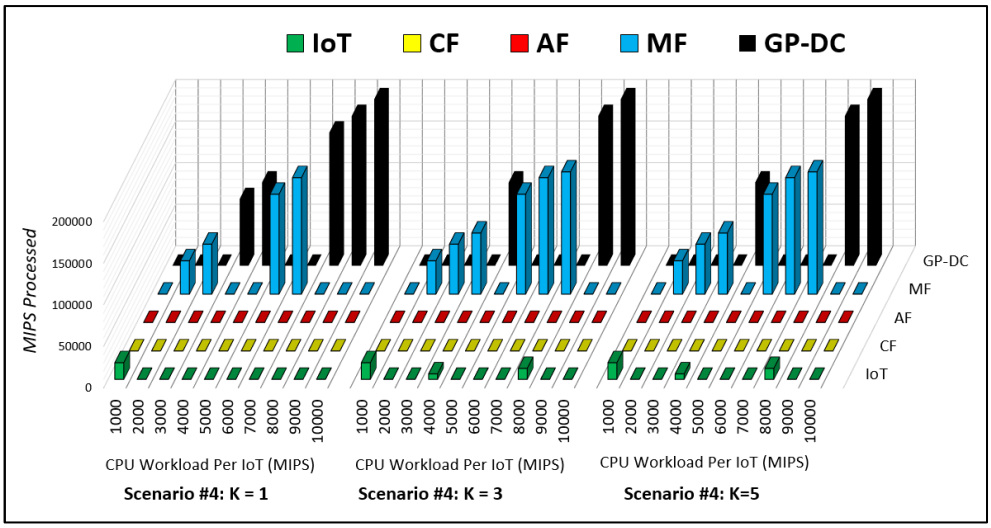


Figure 5.17 Workload Distribution in Scenario #4 at Different Values of K.

5.6 Impact of the SP-DC in Un/Capaciated Design

It is of interest to investigate whether service splitting influences the decision of utilising the highly energy efficient SP-DC. The results indicated that service splitting did not have any influence on the decision of whether to utilise the SP-DC or not, in both cases the short term and long term design problems, i.e. capacitated and un-capacitated, respectively. This is consistent with the findings of chapter 4, for very high demand volumes, the cloud SP-DC produces significant savings as shown in Figure 5.18.

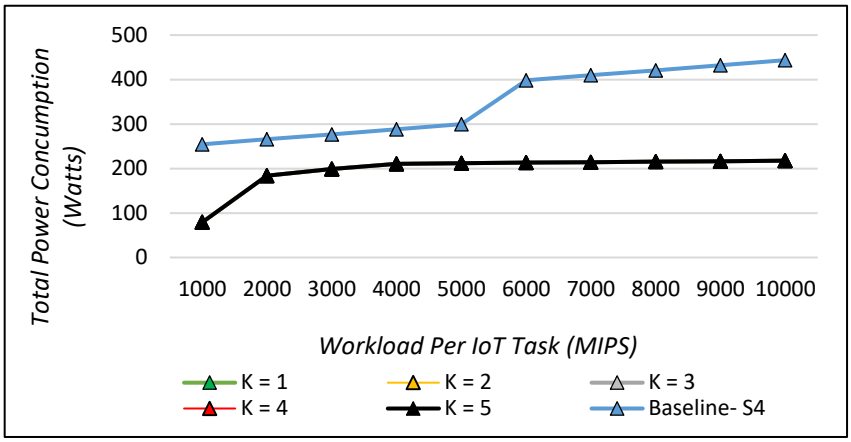


Figure 5.18 Total Power Consumption of the Distributed Approach at Various Values of K.

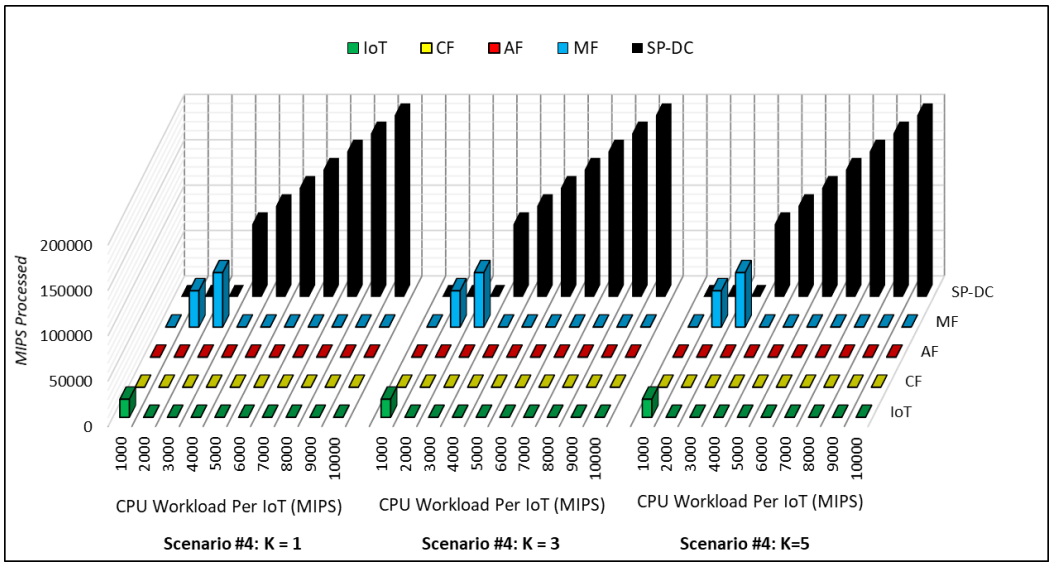


Figure 5.19 Workload Distribution in Scenario #4 at Different Values of K, when SP-DC deployed.

5.7 Inter-Service Synchronisation Processing Overhead

This section considers a scenario in which service splits incur an extra processing overhead due to synchronisation between the subtasks of the service in question, as depicted in Figure 5.20. The extra overhead only considers processing since the communication traffic power consumption is almost negligible in terms of its influence on decision making as network equipment idle power is 60% - 90% of the maximum power consumption.

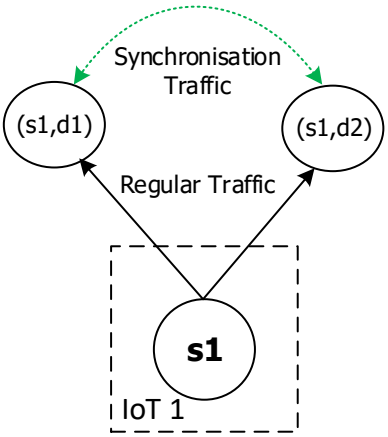


Figure 5.20 an example of synchronisation traffic between subtasks of an IoT service

The power consumption evaluations within this section are based on several processing overheads due to synchronisation and these are ratios such as 1%, 5% and 10%. From Chapter 5, the previously considered scenarios of active IoT devices such as Scenario #1, Scenario #2 and Scenario #3 were considered in the capacitated case, since the largest number of splits occurred in these scenarios where synchronisation overhead was not accounted for. Therefore, it is of interest to investigate the extent to which the synchronisation overhead impacts the decision in terms of making service splits.

Before introducing the MILP model, the additional parameters and variables are defined as follows:

Application Parameters:

ϕ	Synchronisation traffic overhead ratio
$\phi^{(p)}$	Synchronisation processing overhead ratio.
PUE_d	PUE of processing node $d \in P$.

Variables:

λ_{d1d2}^{s}	$\lambda_{d1d2}^{s} = 1$, if there is synchronisation traffic overhead of service $s \in S$ between processing node $d1 \in P$ and $d\ 2 \in P: d2 \neq d1$, otherwise $\lambda_{d1d2}^{s} = 0$.
λ_{d1d2}	Synchronisation traffic between processing nodes $d1 \in P$ and $d\ 2 \in P: d2 \neq d1$.
λ_{mn}^{d1d2}	Synchronisation traffic processing node $d1 \in P$ and $d2 \in P: d2 \neq d1$, traversing link m, n, where $m \in N$ and $n \in N_m$.
$\lambda_{i}^{(sync)}$	Synchronisation traffic on node $i \in N$.
ρ_{d1d2}^{s}	Synchronisation processing demand of service $s \in S$ between processing node $d1 \in P$ and $d2 \in P$.
ρ^{sd}	Service processing demand of IoT source node $s \in S$ hosted at processing device $d \in P$.
Ω^{sd}	$\Omega^{sd} = 1$, if service processing demand of IoT source node $s \in S$ is hosted at destination node $d \in P$, otherwise $\Omega^{sd} = 0$.
$\mathcal{N}_{d}^{(sync)}$	Number of processing servers activated for regular service request and synchronisation processing overhead at node $d \in P$.

The total power consumption equations remain intact except for an additional equation which accounts for synchronisation processing overhead and this is defined as follows:

Power Consumption of Synchronisation Overhead:

$$\sum_{s \in S} \sum_{\substack{d2 \in P: \\ d2 \neq d1}} (\rho_{d1d2}^{s} PUE_{d2} Ei_{d}) \qquad (5.2)$$

Additional Constraints:

$$\sum_{n\in N_m} \lambda_{mn}^{d1d2} - \sum_{n\in N_m} \lambda_{nm}^{d1d2} = \begin{cases} \lambda_{d1d2} & m = d1 \\ -\lambda_{d1d2} & m = d2 \\ 0 & otherwise \end{cases} \tag{5.3}$$

$$\forall d1 \in P, d2 \in P, m \in N: d1 \neq d2.$$

Constraint (5.3) conserves synchronisation traffic from source node to destination node in the topology depicted in Figure 3.2. It ensures that, the total incoming traffic at a node is equal to the total outgoing traffic of that node; unless the node in question is either the source node or the destination node

$$\lambda_{d1d2}^{s} \leq \Omega^{sd1} \tag{5.4}$$

$$\forall s \in S, d1 \in P, d2 \in P: d1 \neq d2$$

$$\lambda_{d1d2}^{s} \leq \Omega^{sd2} \tag{5.5}$$

$$\forall s \in S, d1 \in P, d2 \in P: d1 \neq d2$$

$$\lambda_{d1d2}^{s} \geq (\Omega^{sd1} + \Omega^{sd2}) - 1 \tag{5.6}$$

$$\forall s \in S, d1 \in P, d2 \in P: d1 \neq d2$$

Constraints (5.4) to (5.6) are used in the linearization of the product of binary variables Ω^{sd1} and Ω^{sd2} , where $s \in S$ $d1 \in P$ and $d2 \in P: d2 \neq d1$.

$$\rho_{d1d2}^{s} = \lambda_{d1d2}^{s} D_{s}^{(CPU)}\, \phi \tag{5.7}$$

$$\forall s \in S, d1 \in P, d2 \in P: d1 \neq d2$$

Constraint (5.7) ensures that the total synchronisation processing overhead of the service source node $s \in S$, between processing $d1 \in P$ and $d2 \in P: d2 \neq d1$ is realised.

$$\lambda_i^{(sync)} = \sum_{d1 \in P} \sum_{\substack{d2 \in P: \\ d2 \neq d1}} \lambda_{mn}^{d1d2} + \sum_{d1 \in P} \sum_{\substack{n \in N_m: \\ d1 \neq m, m \in P}} \lambda_{nm}^{d1m} \tag{5.8}$$

$$\forall m \in N$$

Constraint (5.8) ensures that egress and ingress synchronisation traffic on node $i \in N$ is accounted for.

$$\mathcal{N}_d^{(sync)} \geq \frac{\left(\sum_{s \in S} \rho^{sd} + \sum_{s \in S} \sum_{\substack{d1 \in P: \\ d1 \neq d}} \rho_{d1d2}^{s}\right)}{C_d^{(CPU)}} \tag{5.9}$$

Constraint (5.9) determines the number of servers required at processing node $d \in P$.

$$\mathcal{N}_d^{(sync)} \leq \mathcal{V}_d \tag{5.10}$$

$$\forall d \in P$$

Constraint (5.9) ensures that the number of servers activated at a processing node $d \in P$, does not exceed the maximum available number of servers in that node.

$$\lambda_{d1d2} = \sum_{s \in S} \left(\lambda_{d1d2}^{s} D_s^{(Bw)}\right) \tag{5.12}$$

$$\forall d1 \in P, d2 \in P: d1 \neq d2$$

Constraint (5.12) ensures that the total communication demand between $d1 \in P$ and $d2 \in P$, where $d2 \neq d1$ is achieved.

5.7.1 Scenario #1: *A single active IoT*

As can be seen in previous subsections in Figure 5.10, the current scenario incurred the largest number of splits for a number of reasons: 1) due to the number of idle resources in the IoT and 2) the idle power consumption of the ONU devices. After having considered the processing overhead due to synchronisation, the results in Figure 5.21 indicate that, for low demand volumes such as 3000 MIPS as shown in Figure 5.22(a), service splitting is still favourable which can be related to the relatively higher idle power of the CPE fog server compared to that of the IoT devices. However, as shown in Figure 5.21(a), at 5000 MIPS and beyond, the trends observed indicate that synchronisation has a significant impact on the service placement decisions. For instance, at 5000 MIPS, the original solution with no overhead decided to split the total workload among the same group of IoTs, since there was enough capacity offered by IoT devices whereas the current solution has done the same number of splits but due to capacity limitations of the IoT devices (collectively), processing the extra workload due to overhead is done at the CPE fog (CF) as is one hope away. The general trend shows that even at very small overhead ratios (e.g. 1%), service splitting, in the long run, is not an efficient choice as can be seen, with the increase in workload, the services are placed higher and higher up the network hierarchy.

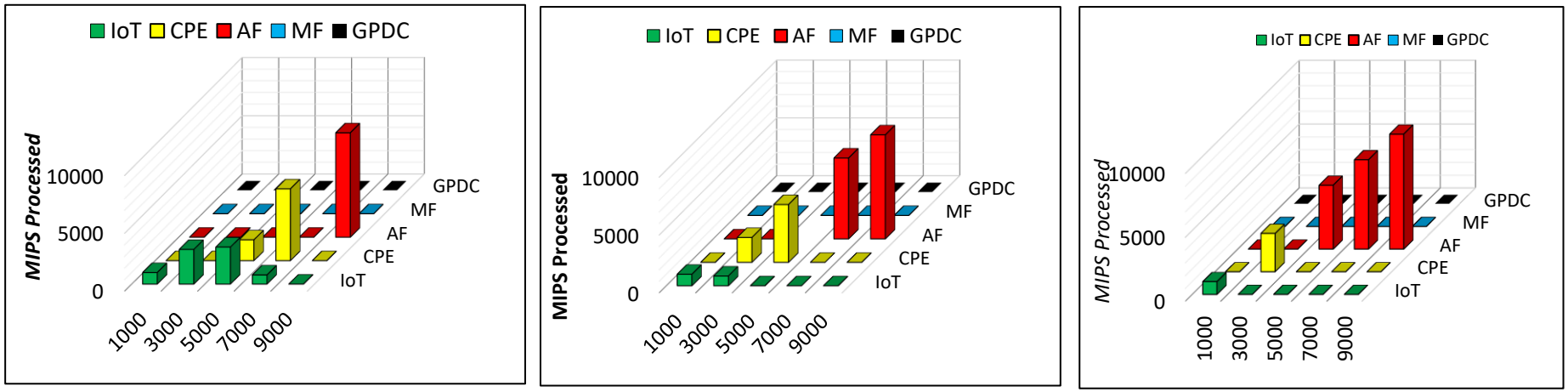


Figure 5.21 Workload distribution at scenario #1 during (a) 1% overhead, (b) 5% overhead and (c) 10% overnead.

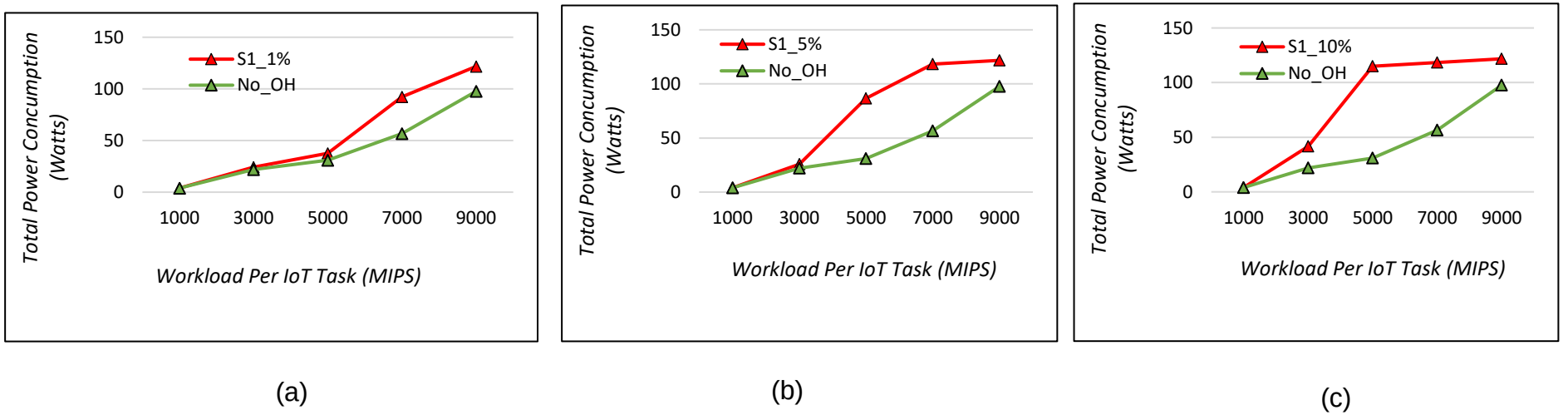


Figure 5.22 Total power consumption overhead at scenario #1 compared to the solution with no overhead (No_OH), during (a) 1% overhead, (b) 5% overhead and (c) 10% overhead.

5.7.2 Scenario #2: Five active IoTs in the same group

In this scenario, the number of active IoT's have increased, hence the total workload has also increased. As can be seen in Figure 5.23(a), the number of service splits have dropped from 15 to 14 at an early stage, i.e. at 3000 MIPS. This confirms the previous observations in scenario #1 that despite some overhead, for very low demands, service splitting, although marginal, it does still introduce savings in all cases as shown in Figure 5.24. Consistent with previous observations, the metro fog becomes the dominant choice due to its processing efficiency as this was the case before synchronisation overhead, if anything, synchronisation overhead will provide even further incentives to utilise the metro fog (MF).

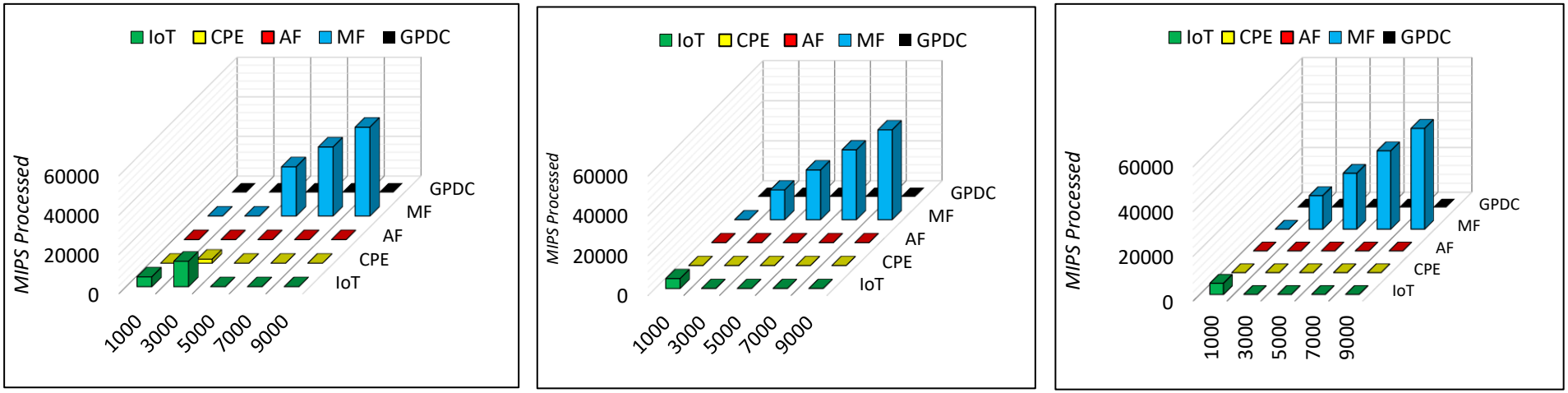


Figure 5.23 Workload distribution at scenario #2 during (a) 1% overhead, (b) 5% overhead and (c) 10% overhead.

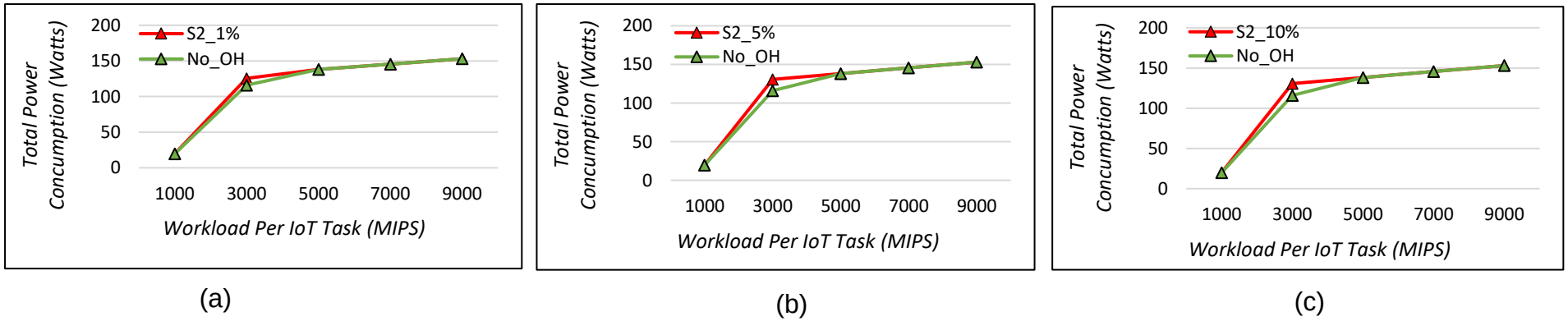


Figure 5.24 Total power consumption overhead at scenario #2 compared to the solution with no overhead (No_OH), during (a) 1% overhead, (b) 5% overhead and (c) 10% overhead

5.7.3 Scenario #3: Four active IoTs, one per group

In this scenario, due to the distribution of the IoT source nodes, for low demand volumes with overheads of 5% and 10%, splitting is still favourable although the number of splits has decreased compared to the case with no synchronisation overheads as can be observed in Figure 5.26(a) and Figure 5.26(b). This is largely due to the fact that, for an IoT source node to access another IoT device to process its request, an ONU device must be activated, hence utilising the CPE fog (CF) servers with larger capacity would be a better packing option as it will drop the number of splits.

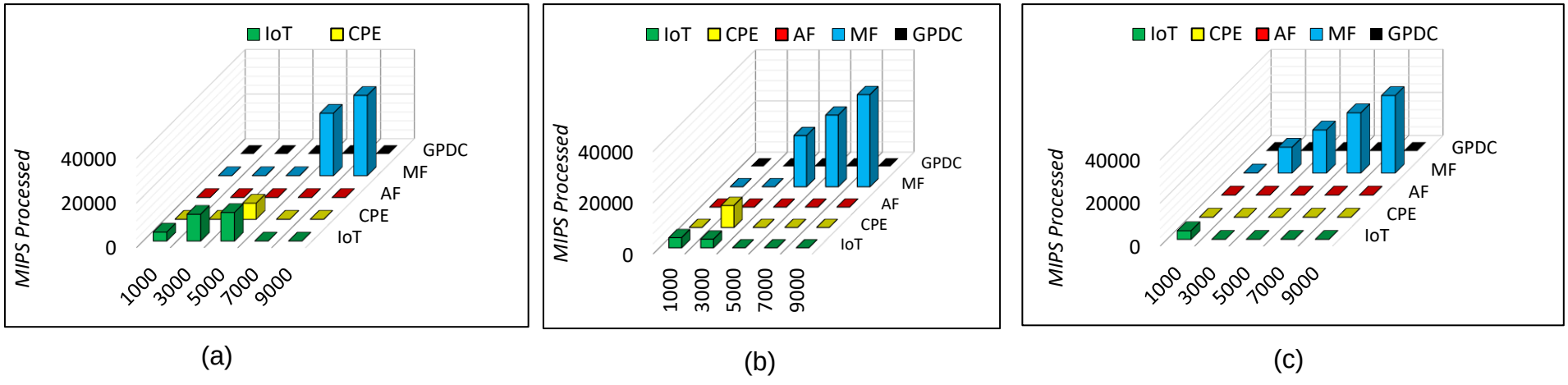


Figure 5.26 Workload distribution at scenario #2 during (a) 3% overhead, (b) 5% overhead and (c) 10% overhead.

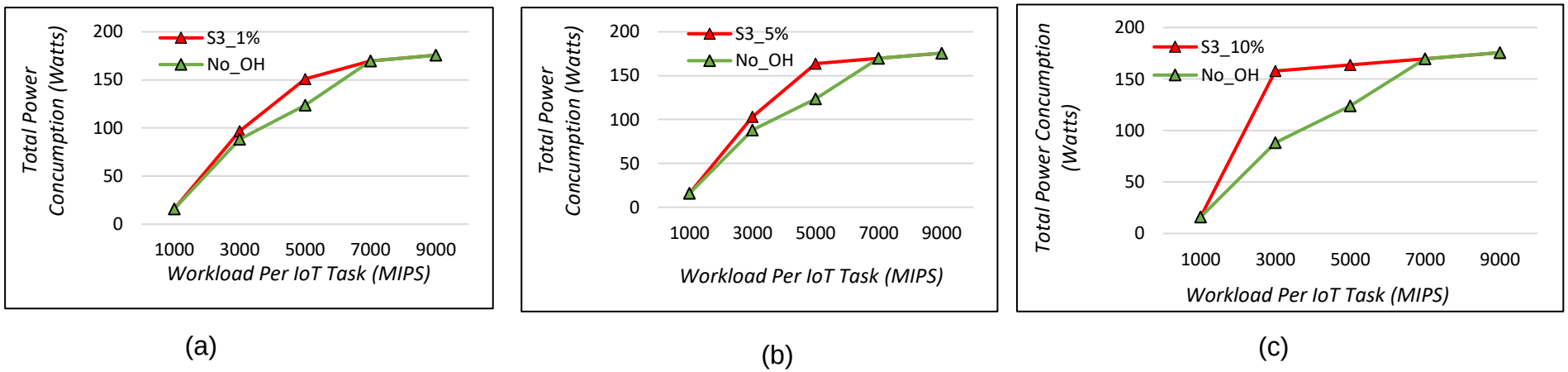


Figure 5.25 Total power consumption overhead at scenario #3 compared to the solution with no overhead (No_OH), during (a) 1% overhead, (b) 5% overhead and (c) 10% overhead.

5.8 Summary

This chapter has extended the work carried out in Chapter 4 by investigating the impact service splitting introduces in terms of the improvement in the energy efficiency of the distributed processing approach. Two design problems were considered, 1) un-capacitated, and 2) capacitated, so that insight can be obtained for the problems that require short and long term solutions, respectively. In the un-capacitated case, it was found that service splitting did not improve the performance of the distributed approach largely due to the idle power consumptions associated with the ONUs and that of the CPE fog servers. It was observed that service splitting produced favourable savings for low workloads, given that subtasks were divided among the IoT devices that resided in the same group as the source(s), as this meant avoiding high idle power of the ONU devices.

In general, for the un-capacitated case, we can conclude that, during very high workload volumes, processing on the CPE fog servers is not a good choice due to the idle power and processing inefficiency of these low power devices, hence, concentrating the workload in the more efficient metro fog was a better choice and this meant service splitting had no impact in this context. We also looked at the solution for a capacitated case (i.e. short term solution). The results showed that, during low workloads (i.e. scenario #1), service splitting achieved significant savings of up to 86% compared to 46% in the case where service splitting was not allowed (i.e. K=1); all compared to the baseline case where processing is all done in the cloud. However, for moderate workloads (i.e. scenario #3), savings due to service splitting

dropped down to 7%. This was due to the locations in which the source nodes where located, i.e. several ONU's had to be activated to process the total demand. It was also confirmed that for very high workloads (i.e. scenario #4, at 4000 MIPS and beyond) service splitting did not have any influence on avoiding the distant SP-DC located in the core network, due to its highly energy efficient server. Having examined the aspect of distributed processing with and without service splitting, we can conclude that the fog approach improves energy efficiency significantly regardless of service splitting, however, the results have shown that the cloud is still relevant and may not be replaced entirely due to the processing efficiency of the cloud servers.

Moreover, the results have shown that the inter-service synchronisation overhead between IoT subtasks has a great influence on the total number of service splits. However much insignificant the ratio of the processing overhead, the results showed that synchronisation processing overhead is not a trivial matter and hence much attention needs to paid to this area in order in order to make the best use of the resources that are available in the edge of the network. The impact of synchronisation processing overhead was evident in Scenario #1 which considered an overhead of only 1%. For this scenario, in the case with no overheads, the most number of splits occurred, however, when 1% overhead was considered the maximum number of splits reduced from 3 to 2 for the lowest ends of the demands and for the higher demands this reduced to zero as processing demands in the Access Fog (AF) server was more energy efficient than splitting among the lower layers

Chapter 6 Resilient IoT Processing

6.1 Introduction

With the growing pervasiveness of the connected “objects” in our surroundings, the Internet of Things (IoT) becomes increasingly pertinent to our daily lives as it is expected to provide a myriad of applications ranging from manipulating simple sensors in a smart home to large scale industrial automation of industries and public utilities. Nevertheless, as these deployments are growing in scale and scope, our dependence on their proper performance is also increasing [115]. Oftentimes, the IoT objects are simple embedded systems that are inexpensive and disposable which can result in frequent failures and operational malfunctions. Thus, this poses several questions in terms of how to design such deployments that are resilient to failures.

So far, in previous chapters, we have considered the IoT infrastructure to be in an ideal operational state, i.e, no failure consideration is taken into account, neither for links nor for nodes. The network survivability approaches for tackling failures are mainly grouped under two schemes: 1) protection and 2) restoration. The former involves pre-emptive actions that are taken by network designers to be prepared for failures before they occur whilst the second approach deals with recovery options after a failure has already occurred, hence the term restoration [113]. This chapter is concerned with studying the power consumption overheads of a range of server protection approaches and draws comparisons with the baseline approach that does not take resilience into account.

Provisioning of resiliency is crucial, especially in an IoT-based surveillance application as any disruption in a node or link can lead to significant loss of intelligence that may be deemed too valuable by law enforcement agencies. In this direction, we aim to introduce resilience to server protection through the approach of node disjoint protection. This is where the primary and backup servers are placed at two nodes that are geographically apart, hence application services can be retained in case of any failure in the primary node. We have utilised the same architecture depicted in Figure 4.1 in our evaluations. Typically, there are three levels of server protection which are widely known as 1+1, 1:1 and 1:N. In the 1+1 case, it is assumed that for every primary server, an additional backup server is activated. Whereas in the 1:1 case, the primary server is activated whilst the backup is on standby (i.e. consuming idle power only). Last but not least is the 1:N case in which a single secondary server can be shared as protection for multiple primary servers. These schemes all provide some level of resilience to a single node failure, however, the 1+1 approach provides the highest level of resilience as each task is replicated on a backup server and as soon as the primary server fails the intended application is redirected to the secondary server in real-time, hence, in this chapter 1+1 server protection is considered. In the following sections, the considered protection approach is evaluated in different scenarios in terms of the number and distribution of active IoTs.

6.2 Modification to the MILP Model

To evaluate resilience, the MILP model introduced in Chapter 4 is considered. To apply the concept of resilience, the MILP model is modified to adapt to account for the additional variables that need to be incorporated.

Here the modified variables appended to the original MILP model are introduced and redefined as follows:

Variables:

λp^{sd} Primary server traffic demand between IoT source node $s \in S$ and processing device $d \in P$.

λp_{mn}^{sd} Primary server traffic flow between IoT source node $s \in S$ and processing device $d \in P$, traversing link (m,n), where $m \in N, n \in N_m$.

$\lambda_d^{(pb)}$ Primary and backup servers' traffic aggregated by node $d \in N$.

$\mathcal{B}_m^{(pb)}$ $\mathcal{B}p_m = 1$, if network node $m \in N$ is activated for primary or secondary server traffic, otherwise $\mathcal{B}p_m = 0$.

$\theta_d^{(pb)}$ Primary server traffic in node $d \in P$ for processing.

ρp^{sd} Primary server processing demand of IoT source node $s \in S$ hosted at processing device $d \in P$.

Ωp^{sd} $\Omega p^{sd} = 1$, if primary server processing demand of IoT source node $s \in S$ is hosted at destination node $d \in P$, otherwise $\Omega p^{sd} = 0$.

Ωp^d $\Omega p^d = 1$, if primary server processing node $d \in P$ is activated, otherwise $\Omega p^d = 0$.

$\mathcal{N}_d^{(pb)}$ Number of primary and backup processing servers activated at node $d \in P$.

$\mathcal{N}b_d$ Number of backup processing servers activated at node $d \in P$.

λb^{sd} Backup server traffic demand between IoT source node $s \in S$ and processing device $d \in P$.

λb_{mn}^{sd} Backup server traffic flow between IoT source node $s \in S$ and processing device $d \in P$, traversing link (m, n), where $m \in N, n \in N_m$.

$\Gamma_{mn}^{(pb)}$ $\Gamma_{mn} = 1$, if core network link m, n, where $m \in C, n \in (N_m \cap C)$ is activated for back server traffic, otherwise $\Gamma_{mn} = 0$.

γb_d $\gamma b_d = 1$, if network for primary server at processing node $d \in P$ is deactivated, otherwise $\gamma b_d = 0$.

ρb^{sd} Backup server processing demand of IoT source node $s \in S$ hosted at processing device $d \in P$.

Ωb^{sd} $\Omega \mathrm{b}^{sd} = 1$, if backup server processing demand of IoT source node $s \in S$ is hosted at destination node $d \in P$, otherwise $\Omega \mathrm{b}^{sd} = 0$.

Ωb^d $\Omega \mathrm{b}^d = 1$, if backup server processing node $d \in P$ is activated, otherwise $\Omega^d = 0$.

$W_{mn}^{(pb)}$ Number of wavelengths used in fibre link (m, n), for primary and backup servers traffic in the core network, where link $m, n \in C$.

$F_{mn}^{(pb)}$ Number of fibres used on link $m, n \in C$, for primary and backup servers traffic in the core network

$Ag_{m}^{(pb)}$ Number of aggregation router ports activated at IP node $m \in C$, for primary and backup servers traffic in the core network

The total power consumption of the entire IoT infrastructure depicted in Figure 4.1 is divided into two parts: 1) Network Power Consumption and 2) Processing Power Consumption. Following subsections contain a detailed breakdown of these power consumptions:

Under the non-bypass light path approach, the IP/WDM total network power consumption is composed of:

6.2.1 Network Power Consumption

1) The power consumption of router ports:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(r)}\lambda_{d}^{(pb)}\right) + \sum_{m\in C}\left(Pidle^{(r)}\left(Ag_{m}^{(pb)} + \sum_{n\in(N_m\cap C)} W_{mn}^{(pb)}\right)\right)\right) \quad (6.1)$$

2) The power consumption of transponders:

$$PUE^{(core)}\left(\sum_{m\in C}\left(Eb^{(t)}\lambda_d^{(pb)}\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(t)}W_{mn}^{(pb)}\right)\right) \tag{6.2}$$

1) The power consumption of EDFAs:

$$PUE^{(core)}\left(\sum_{m\in C}\left(\lambda_m^{(pb)}A_{mn}F_{mn}^{(pb)}\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(e)}A_{mn}F_{mn}^{(pb)}\right)\right) \tag{6.3}$$

3) The power consumption of optical switches:

$$PUE^{(core)}\left(\sum_{m\in C}\left(\lambda_m^{(pb)}\right)+\sum_{m\in C}\left(Pidle^{(o)}\mathcal{B}_m^{(pb)}\right)\right) \tag{6.4}$$

4) The power consumption of regenerators:

$$PUE^{(core)}\left(\sum_{m\in C}\left(\lambda_m^{(pb)}Rg_{mn}W_{mn}^{(pb)}\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(rg)}\,Rg_{mn}W_{mn}^{(pb)}\right)\right) \tag{6.5}$$

The metro network's power consumption is composed of:

$$PUE^{(metro)}\left(\left(\sum_{m\in M^{(R)}}\left(Eb^{(MR)}\lambda_m^{(pb)}\mathcal{R}\right)\right.\right. \\ \left. + \sum_{m\in M^{(R)}}\left(\mathcal{B}_m^{(pb)}Pidle^{(MR)}\mathcal{R}\right)\right) \\ + \left(\sum_{m\in M^{(Sw)}}\left(Eb^{(MSw)}\lambda_m^{(pb)}\right)\right. \\ \left.\left. + \sum_{m\in M^{(R)}}\left(\mathcal{B}_m^{(pb)}Pidle^{(MSw)}\right)\right)\right) \tag{6.6}$$

The access network's power consumption consists of the power consumption of OLT and ONU devices, which is given as:

$$PUE^{(access)}\left(\sum_{m\in OLT}\left(Eb^{(OLT)}\lambda_m^{(pb)}\right) + \sum_{m\in OLT}\left(\mathcal{B}_m^{(pb)}Pidle^{(OLT)}\right)\right) \\ + \left(\sum_{m\in ONU}\left(Eb^{(ONU)}\lambda_m^{(pb)}\right)\right. \\ \left. + \sum_{m\in ONU}\left(\mathcal{B}_m^{(pb)}Pidle^{(ONU)}\right)\right) \tag{6.7}$$

The IoT devices' communication interfaces power consumption is given as:

$$\sum_{m\in I}\left(Eb^{(TxRx)}\lambda_m^{(pb)}\right) + \sum_{m\in I}\mathcal{B}_m^{(pb)}Pidle^{(TxRx)} \tag{6.8}$$

6.2.2 Processing Power Consumption

The total power consumption of the processing devices (or servers) is composed of:

1) The processing power consumption of IoT devices:

$$\sum_{s\in S}\sum_{d\in I}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in I}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right) \quad (6.9)$$

2) The processing power consumption of CPE fog (CF) servers:

$$\sum_{s\in S}\sum_{d\in ONU}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in ONU}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right) \quad (6.10)$$

3) The processing power consumption of access fog (AF) servers:

$$PUE^{(access)}\left(\sum_{s\in S}\sum_{d\in OLT}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in OLT}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right) \quad (6.11)$$

4) The processing power consumption of metro fog (MF) servers:

$$PUE^{(metro)}\left(\sum_{s\in S}\sum_{d\in M^{(Sw)}}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in M^{(Sw)}}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right) \quad (6.12)$$

5) The processing power consumption of cloud DC servers

$$PUE^{(dc)}\left(\sum_{s\in S}\sum_{d\in DC}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in DC}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right) \quad (6.13)$$

6.2.3 Power Consumption of Network inside Processing Nodes

The cloud DCs network power consumption is composed of the power consumption of cloud DC routers and switches:

$$PUE^{(DC)}\left(\left(\sum_{d\in DC}\left(Eb^{(DcSw)}\theta_d^{(pb)}\right)+\sum_{d\in DC}\left(Pidle^{(DcSw)}\Omega p^{d}\right)\right)+\left(\sum_{d\in DC}\left(Eb^{(DcR)}\theta_d^{(pb)}\right)+\sum_{m\in DC}\left(Pidle^{(DcR)}\,\Omega p^{d}\right)\right)+\sum_{m\in DC}\left(Pidle^{(DcR)}\,\gamma p^{d}\right)+\sum_{m\in DC}\left(Pidle^{(DcSw)}\,\gamma p^{d}\right)\right) \quad (6.14)$$

The metro fog network power consumption of metro fog routers and switches is given as:

$$PUE^{(metro)}\Bigg(\Bigg(\sum_{d\in M^{(Sw)}}\left(Eb^{(MfR)}\theta_d^{(pb)}\right) + \sum_{m\in M^{(Sw)}}\left(Pidle^{(MfR)}\Omega p^d\right)\Bigg) + \Bigg(\sum_{d\in M^{(Sw)}}\left(Eb^{(MfSw)}\theta_d^{(pb)}\right) + \sum_{d\in M^{(Sw)}}\left(Pidle^{(MfSw)}\right)\Omega p^d\Bigg) + \sum_{m\in M^{(Sw)}}\left(Pidle^{(MfR)}\,\gamma p^d\right) + \sum_{m\in M^{(Sw)}}\left(Pidle^{(MfSw)}\,\gamma p^d\right)\Bigg) \tag{6.15}$$

The access fog network power consumption of access fog routers and switches is given as:

$$PUE^{(access)}\Bigg(\sum_{d\in OLT}\left(Eb^{(AfR)}\theta_d^{(pb)}\right) + \sum_{d\in OLT}\left(Pidle^{(AfR)}\Omega p^d\right) + \sum_{d\in OLT}\left(Eb^{(AfSw)}\theta_d^{(pb)}\right) + \sum_{d\in OLT}\left(Pidle^{(AfSw)}\gamma p^d\right) + \sum_{d\in OLT}\left(Pidle^{(AfR)}\gamma p^d\right)\Bigg) \tag{6.16}$$

The CPE fog network power consumption of CPE fog switches is given as:

$$\sum_{d\in ONU}\left(Eb^{(cpefSw)}\theta_d^{(pb)}\right)+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\Omega p^d\right)+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\gamma p^d\right) \tag{6.17}$$

The MILP model's objective function is given as follows:

Objective

Minimise total power consumption:

$$PUE^{(core)}\left[\sum_{m\in C}\left(\lambda_m^{(pb)}\right)+\sum_{m\in C}\left(Pidle^{(r)}\left(Ag_m^{(pb)}+\sum_{n\in(N_m\cap C)}W_{mn}^{(pb)}\right)\right)\right]+ \tag{6.18}$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(\lambda_m^{(pb)}\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(t)}W_{mn}^{(pb)}\right)\right)\right]+$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(\lambda_m^{(pb)}A_{mn}F_{mn}^{(pb)}\right)+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(e)}A_{mn}F_{mn}^{(pb)}\right)\right)\right]+$$

$$PUE^{(core)}\left[\left(\sum_{m\in C}\left(\lambda_m^{(pb)}\right)+\sum_{m\in C}\left(Pidle^{(o)}\mathcal{B}_m^{(pb)}\right)\right)\right]+$$

$$PUE^{(core)}\left[\sum_{m\in C}\left(\lambda_m^{(pb)}Rg_{mn}W_{mn}^{(pb)}\right)\right.$$

$$\left.+\sum_{m\in C}\sum_{n\in(N_m\cap C)}\left(Pidle^{(rg)}\,Rg_{mn}W_{mn}^{(pb)}\right)\right]$$

$$+PUE^{(metro)}\left[\sum_{m\in M^{(R)}}\left(Eb^{(MR)}\lambda_m^{(pb)}\mathcal{R}\right)\right.$$

$$+\sum_{m\in M^{(R)}}\left(\mathcal{B}_m^{(pb)}Pidle^{(MR)}\mathcal{R}\right)+\sum_{m\in M^{(Sw)}}\left(Eb^{(MSw)}\lambda_m^{(pb)}\right)$$

$$\left.+\sum_{m\in M^{(R)}}\left(\mathcal{B}_m^{(pb)}Pidle^{(MSw)}\right)\right]$$

$$+\;PUE^{(access)}\left[\sum_{m\in OLT}\left(Eb^{(OLT)}\lambda_m^{(pb)}\right)\right.$$

$$\left.+\sum_{m\in OLT}\left(\mathcal{B}_m^{(pb)}Pidle^{(OLT)}\right)\right]+$$

$$\sum_{m\in ONU}\left(Eb^{(ONU)}\lambda_m^{(pb)}\right)+\sum_{m\in ONU}\left(\mathcal{B}_m^{(pb)}Pidle^{(ONU)}\right)$$

$$+\sum_{m\in I}\left(Eb^{(TxRx)}\lambda_m^{(pb)}\right)+\sum_{m\in I}Pidle^{(TxRx)}\mathcal{B}_m^{(pb)}$$

$$+\sum_{s\in S}\sum_{d\in I}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)$$

$$+\sum_{d\in I}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)$$

$$+\sum_{s\in S}\sum_{d\in ONU}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)$$

$$+\sum_{d\in ONU}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)$$

$$+ PUE^{(access)}\left[\sum_{s\in S}\sum_{d\in OLT}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)+\sum_{d\in OLT}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right]$$

$$+\ PUE^{(metro)}\left[\sum_{s\in S}\sum_{d\in M^{Sw}}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)\right.$$

$$\left.+\sum_{d\in M^{(Sw)}}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right]$$

$$+ PUE^{(DC)}\left[\sum_{s\in S}\sum_{d\in DC}\left(Ei_d(\rho p^{sd}+\rho b^{sd})\right)\right.$$

$$\left.+\sum_{d\in DC}\left(Pidle_d^{(cpu)}\mathcal{N}_d^{(pb)}\right)\right]$$

$$+ PUE^{(DC)}\left[\sum_{d\in DC}\left(Eb^{(DcSw)}\theta_d^{(pb)}\right)\right.$$

$$+\sum_{m\in DC}\left(Pidle^{(DcSw)}\Omega p^d\right)+\sum_{d\in DC}\left(Eb^{(DcR)}\theta_d^{(pb)}\right)$$

$$+\sum_{d\in DC}\left(Pidle^{(DcR)}\,\Omega p^d\right)+\sum_{m\in DC}\left(Pidle^{(DcR)}\,\gamma p^d\right)$$

$$\left.+\sum_{m\in DC}\left(Pidle^{(DcSw)}\,\gamma p^d\right)\right]$$

$$+ PUE^{(metro)}\left[\sum_{d\in M^{(Sw)}}\left(Eb^{(MfR)}\theta_d^{(pb)}\right)\right.$$

$$+\sum_{d\in M^{(Sw)}}\left(Pidle^{(MfR)}\Omega p^d\right)+\sum_{d\in M^{(Sw)}}\left(Eb^{(MfSw)}\theta_d^{(pb)}\right)$$

$$\left.+\sum_{d\in M^{(Sw)}}\left(Pidle^{(MfSw)}\Omega p^d\right)\right]+\sum_{m\in M^{(Sw)}}\left(Pidle^{(MfR)}\,\gamma p^d\right)$$

$$+\sum_{m\in M^{(Sw)}}\left(Pidle^{(MfSw)}\,\gamma p^d\right)$$

$$+PUE^{(access)}\left[\sum_{m\in OLT}\left(\lambda_m^{(pb)}\right)\right.$$
$$+\sum_{m\in OLT}\left(\mathcal{B}_m^{(pb)}Pidle^{(AfR)}\right)$$
$$+\sum_{m\in OLT}\left(Eb^{(AfSw)}\theta_m^{(pb)}\right)$$
$$+\sum_{m\in OLT}\left(\mathcal{B}_m^{(pb)}Pidle^{(AfSw)}\right)$$
$$\left.+\sum_{d\in OLT}\left(Pidle^{(AfSw)}\gamma p^d\right)+\sum_{d\in OLT}\left(Pidle^{(AfR)}\gamma p^d\right)\right]$$
$$+\sum_{d\in ONU}\left(Eb^{(cpefSw)}\theta_d^{(pb)}\right)+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\Omega d^d\right)$$
$$+\sum_{d\in ONU}\left(Pidle^{(cpefSw)}\,\gamma p^d\right)$$

Subject to:

$$\sum_{n\in N_m}\lambda p_{mn}^{sd}-\sum_{n\in N_m}\lambda p_{nm}^{sd}=\begin{cases}\lambda p_{sd} & m=s\\ -\lambda p_{sd} & m=d\\ 0 & otherwise\end{cases} \tag{6.19}$$

$$\forall s\in S, d\in P, m\in N: s\neq d.$$

Constraint (*6.19*) conserves primary server traffic from the source node to the destination node in the considered topology depicted Figure 4.1. It ensures that the total incoming traffic at a node is equal to the total outgoing traffic of that node; unless the node in question is either the source node or the destination node.

$$\sum_{d \in P} \rho p^{sd} = D_s^{(CPU)} \tag{6.20}$$

$$\forall s \in S$$

Constraint (*6.20*) ensures that the primary processing task per IoT source node $s \in S$ is met at a given destination node.

$$\rho p^{sd} \geq \Omega p^{sd} \tag{6.21}$$

$$\forall s \in S, d \in P$$

$$\rho p^{sd} \leq M \Omega p^{sd} \tag{6.22}$$

$$\forall s \in S, d \in P$$

Constraints (*6.21*) and (*6.22*) are used in the conversion of ρp^{sd} , where $s \in S$ and $d \in P$ into its binary equivalence. When $\rho p^{sd} = 1$, it indicates that source node $s \in S$ is processing its primary server CPU task at destination node $d \in P$.

$$\sum_{d \in P} \rho b^{sd} = D_s^{(CPU)} \tag{6.23}$$

$$\forall s \in S$$

Constraint (4.23) ensures that backup processing task per IoT source node $s \in S$ is met at a given destination node.

$$\rho b^{sd} \geq \Omega b^{sd} \qquad (6.24)$$

$$\forall s \in S, d \in P$$

$$\rho b^{sd} \leq M\Omega b^{sd} \qquad (6.25)$$

$$\forall s \in S, d \in P$$

Constraints (4.24) and (4.25) are used in the conversion of ρb^{sd} , (where $s \in S$ and $d \in P$) into its binary equivalent. Here $\rho b^{sd} = 1$, indicates that source node $s \in S$ is processing its primary server CPU task at destination node $d \in P$.

$$\sum_{n \in N_m} \lambda b_{mn}^{sd} - \sum_{n \in N_m} \lambda b_{nm}^{sd} = \begin{cases} \lambda b^{sd} & m = s \\ -\lambda b^{sd} & m = d \\ 0 & otherwise \end{cases} \qquad (6.26)$$

$$\forall s \in S, d \in P, m \in N : s \neq d.$$

Constraint (6.26) conserves backup server traffic from the source node to the destination node in the considered topology. It ensures that the total incoming traffic at a node is equal to the total outgoing traffic of that node; unless the node in question is either the source node or the destination node.

$$\lambda b^{(sd)} = D_s^{(BW)} \Omega b^{sd} \qquad (6.27)$$

$$\forall s \in S, d \in P$$

Constraint (6.27) ensures the total traffic demand of the primary server for each source node is met. The binary variable Ωp^{sd} ensures that traffic is only directed to the destination node that is hosting a processing task.

$$(\Omega p^{sd} + \Omega b^{sd}) \leq 1 \tag{6.28}$$

$$\forall s \in S, d \in P$$

Constraint (4.28) ensures primary and backup servers are geographically located in different processing node $d \in P$.

$$\sum_{d \in P} \Omega p^{sd} \leq 1 \tag{6.29}$$

$$\forall s \in S$$

Constraint (4.29) ensures that primary servers' processing tasks are placed at a single location only, hence, no service splitting is allowed.

$$\sum_{d \in P} \Omega b^{sd} \leq 1 \tag{6.30}$$

$$\forall s \in S$$

Constraint (4.30) ensures that backup servers' processing tasks are placed at a single location only, hence, no service splitting is allowed.

$$\mathcal{N}_d^{(pb)} \geq \sum_{s \in S} \frac{(\rho p^{sd} + \rho b^{sd})}{C_d^{(CPU)}} \tag{6.31}$$

$$\forall d \in P$$

Constraint (4.31) determines the number of servers required at processing node $d \in P$, in order to process the primary and back servers' demands.

Constraint (4.32) determines the number of backup servers required at

$$\mathcal{N}b_d \geq \sum_{s \in S} \frac{(\rho b^{sd})}{C_d^{(CPU)}} \tag{6.32}$$

$$\forall d \in P$$

processing node $d \in P$.

$$\mathcal{N}_d^{(pb)} \leq \mathcal{V}_d \tag{6.33}$$

$$\forall d \in P$$

Constraint (4.33) ensures that the number of primary and backup servers activated at a processing node $d \in P$, does not exceed the maximum available to that node.

$$\sum_{s \in I} \Omega p^{sd} \geq \Omega p^d \tag{6.34}$$

$$\forall d \in P$$

$$\sum_{s \in I} \Omega p^{sd} \leq M\, \Omega p^d \tag{6.35}$$

$$\forall d \in P$$

Constraints (4.34) and (4.35) are used to ensure that, the binary variable $\Omega p^d = 1$ if primary processing server node $d \in P$ is activated, otherwise $\Omega p^d = 0$.

$$\sum_{s \in I} \Omega b^{sd} \geq \Omega b^d \tag{6.36}$$

$$\forall d \in P$$

$$\sum_{\mathrm{s}\in\mathrm{I}} \Omega\mathrm{b}^{\mathrm{sd}} \leq \mathrm{M}\,\Omega\mathrm{b}^{\mathrm{d}} \qquad (6.37)$$

$$\forall \mathrm{d} \in \mathrm{P}$$

Constraints (4.36) and (6.37) ensure that, the binary variable $\Omega b^d = 1$ if backup server processing node $d \in P$ is activated, otherwise $\Omega b^d = 0$.

$$\gamma b_d \leq \Omega b^d \qquad (6.38)$$

$$\forall d \in P$$

$$\gamma b_d \leq (1 - \Omega p^d) \qquad (6.39)$$

$$\forall d \in P$$

$$\gamma b_d \geq \left(\left(\Omega b^d + (1 - \Omega p^d)\right) - 1\right) \qquad (6.40)$$

$$\forall d \in P$$

Constraints (6.38) to (6.40) are used in the linearization of the product of binary variables $\Omega\mathrm{b}^{\mathrm{d}}$ and $(1 - \Omega\mathrm{p}^{\mathrm{d}})$, where $d \in P$. The term $(1 - \Omega p^d)$ ensures that, $\gamma b_d = 0$, if primary server processing device $d \in P$ is already activated, otherwise $\gamma b_d = 0$.

$$\lambda_m^{(pb)} = \sum_{\substack{s\in S:\\ m=s}} \sum_{d\in P} \sum_{n\in N_m} (\lambda p_{mn}^{sd} + \lambda b_{mn}^{sd}) \qquad (6.41)$$

$$\forall m \in S$$

$$\lambda_m^{(pb)} = \sum_{\substack{s \in S: \\ m \neq s}} \sum_{\substack{d \in P: \\ s \neq d}} \sum_{n \in N_m} (\lambda p_{nm}^{sd} + \lambda b_{nm}^{sd})$$

$$\forall m \in (I \cup OLT \cup M^{(Sw)} \cup M^{(R)} \cup DC) \tag{6.42}$$

$$\lambda_m^{(pb)} = \sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} \sum_{\substack{n \in N_m: \\ n \in (N_m \cap C)}} (\lambda p_{mn}^{sd} + \lambda b_{mn}^{sd}) \tag{6.43}$$

$$\forall m \in C$$

Constraint (6.41) ensures that the total aggregate traffic on node $m \in S$ is accounted for only when the source node is transmitting. Whilst constraint (6.42) ensures that, the aggregate traffic on node $m \in N,$ where $m \notin C,$ is only accounted for if the transmitting node $m \neq s$ is not the source of the traffic. Finally constraint (6.43) determines the aggregate traffic in the core network, given that the transmitting node $m \in C$ is not equal to the source of the traffic node $s \in S$.

$$\theta_d^{(pb)} \leq M \Omega p^d \tag{6.44}$$

$$\forall d \in P$$

$$\theta_d^{(pb)} \leq \lambda_d^{(pb)} \forall d \in P \tag{6.45}$$

$$\theta_d^{(pb)} \geq \lambda_d^{(pb)} - (1 - \Omega p^d) M \tag{6.46}$$

$$\forall d \in P$$

Constraints (6.44) to (6.46) are used to linearise the product of binary variable Ωp^d and continuous non-negative variable $\lambda_d^{(pb)}$, where $d \in P$. This ensures that traffic (of primary and backup servers) on a processing node $d \in P$ is only accounted for if it is destined to that node for processing.

$$\lambda_m^{(pb)} \geq \mathcal{B}_m^{(pb)} \quad \forall m \in N \tag{6.47}$$

$$\lambda_m^{(pb)} \leq M\mathcal{B}_m^{(pb)} \quad \forall m \in N \tag{6.48}$$

Constraints (6.47) and (6.48) convert the continuous variable $\lambda_m^{(pb)}$, where $m \in N$ into its binary equivalent.

$$\lambda p^{sd} = D_s^{(BW)} \Omega p^{sd} \quad \forall s \in S, d \in P \tag{6.49}$$

Constraint (6.49) ensures that the total traffic demand of the primary server for each source node is met. The binary variable Ωp^{sd}, where $s \in S$ and $d \in P$, ensures that traffic is only directed to the destination node that is hosting a processing task.

$$\sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} (\lambda p_{mn}^{sd} + \lambda b_{mn}^{sd}) \leq C_{mn} \quad \forall m \in \left(I \cup ONU \cup OLT \cup M^{(Sw)} \cup M^{(R)} \cup DC\right): n \in N_m \tag{6.50}$$

Constraint (6.50) ensures that the total traffic (primary and backup) carried on link m,n, in the metro, access, DC and IoT layers do not exceed its capacity in Mbps.

$$Ag_m^{(pb)} \geq \frac{\lambda_m^{(pb)}}{B} \tag{6.51}$$

$$\forall m \in C$$

Constraint (*6.51*) gives the number of aggregation router ports for primary and backup server traffic at each IP/WDM node.

$$\sum_{s \in S} \sum_{\substack{d \in P: \\ s \neq d}} (\lambda p_{mn}^{sd} + \lambda b_{mn}^{sd}) \leq W_{mn}^{(pb)} B \tag{6.52}$$

$$\forall m \in C: n \in (C \cap N_m)$$

$$W_{mn}^{(pb)} \leq W^{(pb)} F_{mn} \tag{6.53}$$

$$\forall m \in C: n \in (C \cap N_m)$$

Constraints (6.52) and (6.53) represent the physical link capacity of the IP/WDM optical links. Constraint (6.52) ensures that the total traffic of primary and backup server on a link does not exceed the capacity of a single wavelength while constraint (6.53) ensures the total number of wavelength channels does not exceed the capacity of a single fibre link.

6.3 Power Consumption Evaluation Using MILP

The power consumption of the resilient distributed processing approach is evaluated using the modified MILP. To carry out the evaluations, the same scenarios considered previously in Chapter 4 and Chapter 5 of active IoTs is

considered: scenario #1, scenario #2, scenario #3 and scenario #4. The distributed processing architecture remains unchanged except for a minimal change in the core network layer in which an additional data centre (DC2) has been considered and is still a single hop from the aggregation core router port, as depicted in Figure 6.1.

The input parameters for the original MILP model remain the same however several additional assumptions must hold, which are defined as follows:

- Idle power consumption of networking and processing devices must only be consumed once. e.g. if a device is already activated by a primary server, then the same device must not be accounted for by a backup server, and vice versa.
- The infrastructure is said to be protected, only if, for every demand, the primary and backup servers are geographically apart, e.g. node disjoint. However, backup servers belonging to different demands can be placed in the same node.
- The primary and backup servers per demand must remain intact, i.e. service splitting is not taken into account.
- The performance of the evaluated scenarios are compared to their respective baselines which consider no protection, however, the service placement is optimised by MILP.
- The design problem remains unchanged similar to the un-capacitated problem in Chapter 4.
- The cloud data centres comprise of general purpose servers only.

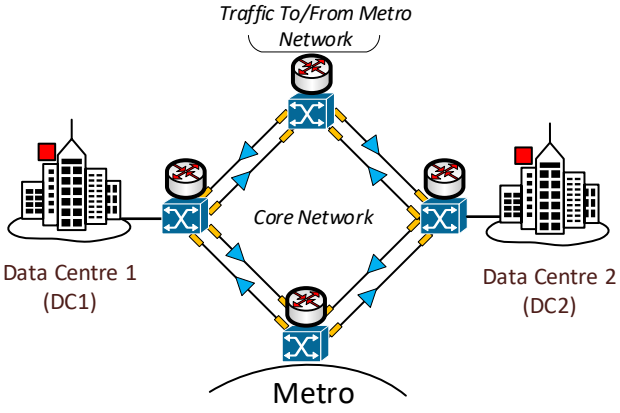


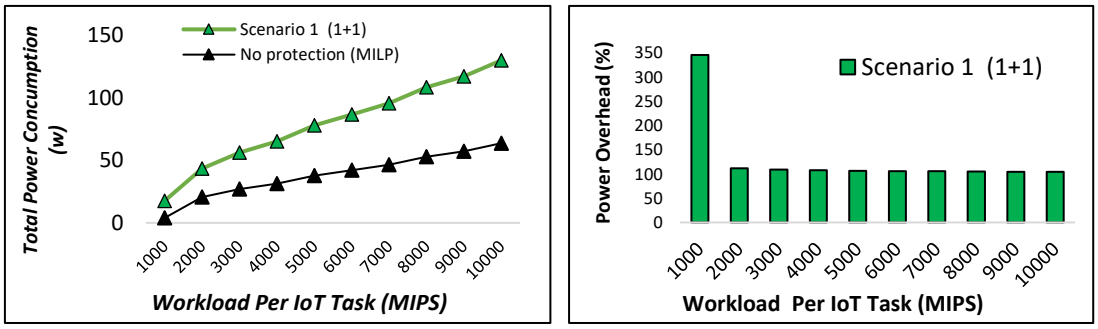
Figure 6.1 Additional DC Added to the Original Architecture.

6.3.1 **Scenario #1:** A single Active IoT

Figure 6.2(a) shows the total power consumption of the 1+1 server protection in Scenario #1 in which only a single IoT device is active out the total 20 IoT devices. It can be seen that the largest power consumption overhead (in percentage) is incurred in the case where the demand is 1000 MIPS. This is understandable since, at that particular workload, the IoT source node processes its own demand locally in the baseline approach (the baseline approach has no protection) which results in very low power consumption. However, when protection is considered, it would mean that an ONU device is activated which has a relatively higher idle power compared to the IoT node, hence 350% overhead in the total power consumption for having server protection at this level. Perhaps if a more efficient communication link such as device to device (D2D) communication based on WiFi Direct was available in the IoT layer, this overhead could be greatly reduced, as the communication ONU device will not occur [129].

Similar to the results in Chapter 4, un-capacitated, scenario #1, the services are kept in the IoT and CPE fog (CF) layers due to the low volume of

workload, as shown in Figure 6.3. However, the difference, in this case, is that two types of traffic exist, primary and backup. If we take 1000 MIPS as an example, the model allocates the primary server onto the IoT source node whilst the backup server is offloaded to another IoT located in the same group via the ONU devices. Likewise, similar cases can be observed for workloads at 2000 MIPS and beyond but in the CPE fog layer instead. Due to the low volume of workload, activating multiple ONUs in different parts of the network is more efficient than activating the higher layer servers due to their high idle power consumption. Hence, power consumption overhead for such cases remains almost constant as the baseline approach provided the same solution.



(a) (b)

Figure 6.2 (a) Total power consumption of 1+1 sever protection in Scenario #1 compared to the baseline, (b) power overhead in percentage for 1+1 protection compared to baseline, for the same scenario.

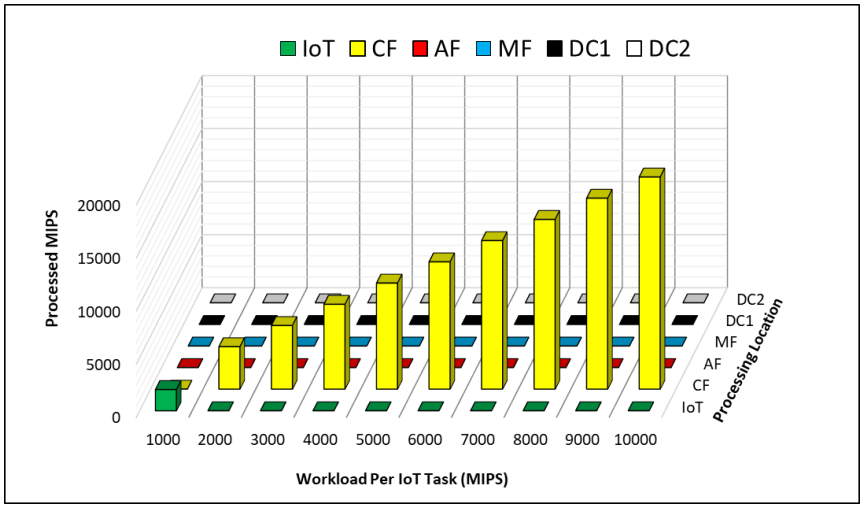
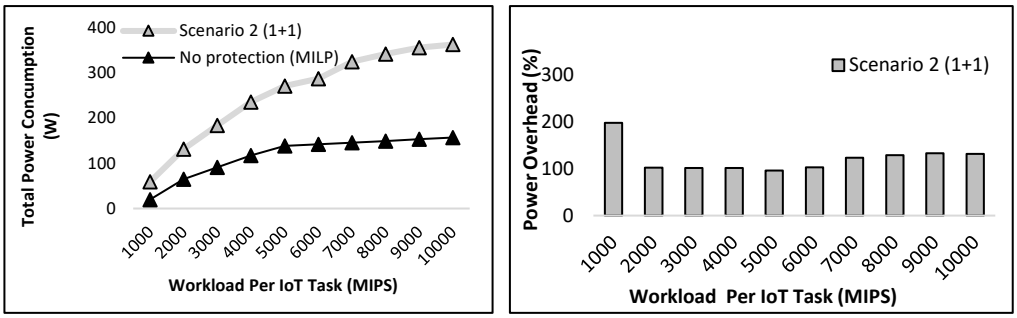


Figure 6.3 Primary and backup servers distribution in Scenario #1

6.3.2 Scenario #2: Five active IoTs in the same group

In this scenario, consistent with previous observations, Figure 6.4(b) shows that the greatest overhead occurs when protecting low workload volumes that are within the capacity of IoT devices since compared to the very low power consumption of the baseline, overheads are expected to be significant. Unlike scenario #1, as can be seen in Figure 6.4(b), overheads have only remained constant for workloads at 2000 MIPS to 4000 MIPS, at about 100%. This is because the baseline solution had chosen the same solution as the one in this scenario, which is to process both primary and backup traffic within the CPE fog layer. Compared with the next workload, at 6000 MIPS, the overheads drop slightly down to about 95%. This is attributed to the fact that in the baseline, for this workload, the metro fog server was wholly used whilst for the protection scenario, CPE fog servers were used in combination to host the primary servers. The same observation as above is made for workloads of 8000 MIPS to 1000 MIPS, except that, the overhead is slightly greater than the scenario at 6000 MIPS due to the activation of the cloud data centre (DC1) in combination with metro fog (MF), again proving the cloud's relevance for high workloads due to its efficiency, as shown in Figure 6.5



(a) (b)

Figure 6.4 (a) Total power consumption of 1+1 sever protection in Scenario #2 compared to the baseline, (b) power overhead in % for 1+1 protection compared to baseline, for the same scenario.

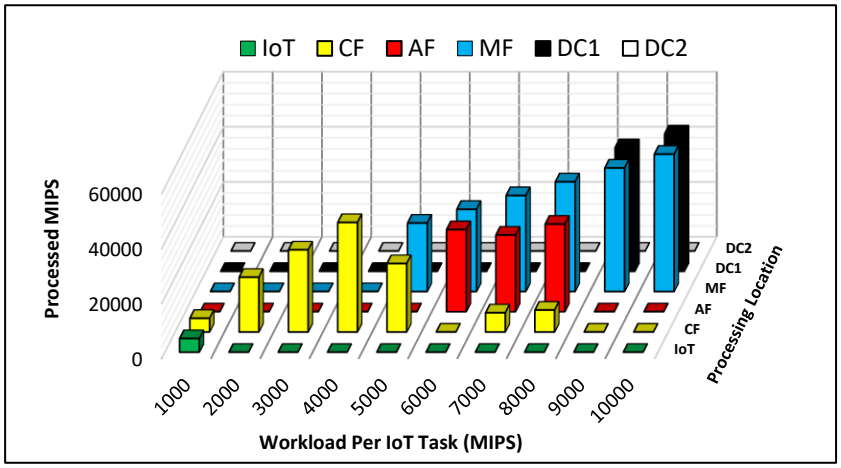
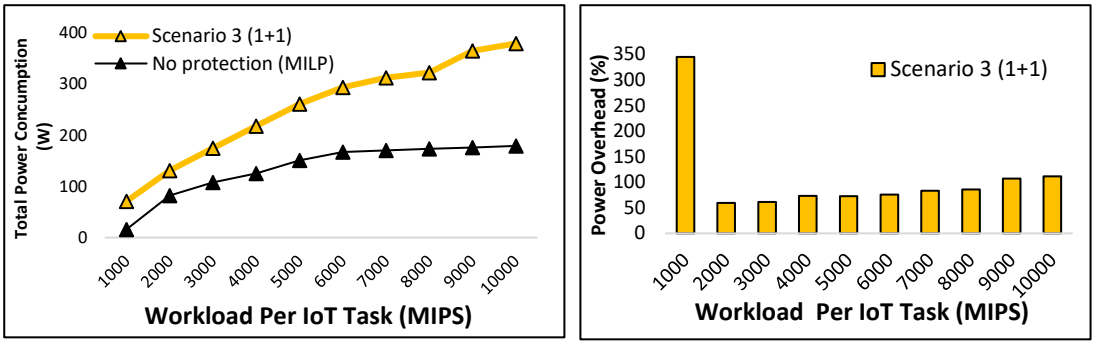


Figure 6.5 Primary and backup servers distribution in Scenario #2.

6.3.3 Scenario #3: *Three active IoTs, one per group*

In this scenario, due to the distribution of the IoT source nodes, the trends vary from scenario #2. As shown in Figure 6.7, for low demands at 1000 MIPS, the IoT layer is utilised to host both primary and secondary servers, since there is only a single active IoT per group, there is enough capacity to host both primary and secondary servers. Unlike scenario #2, when 5 active IoTs resided in the same group, the CPE fog was utilised to host the backup servers as this avoided going through the access network to access another IoT device in another group. Also for workloads of 9000 MIPS to 10,000 MIPS. As can be seen in Figure 6.7, the model makes use of the CPE fog servers in

addition to the access and metro fogs. The reason for this is that, since the IoT source nodes are no longer in the same group, e.g. 1 source node per ONU, it would make more sense to pack a single CPE fog server for each source node as the ONUs are already activated to gain access to the higher processing layers such as the access fog and metro fog.



(a) (b)

Figure 6.6 (a) Total power consumption of 1+1 sever protection in Scenario #3 compared to the baseline, (b) power overhead in percentage for 1+1 protection compared to baseline, for the same scenario.

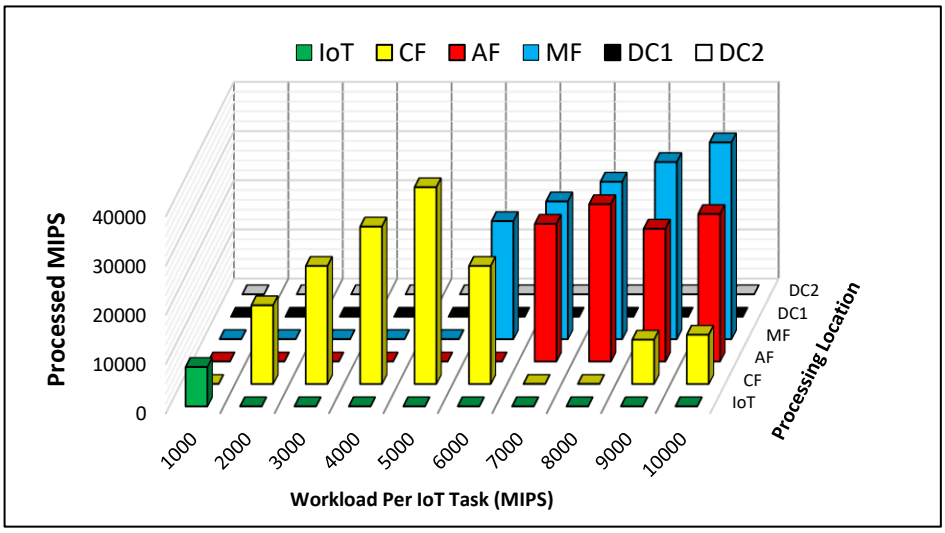
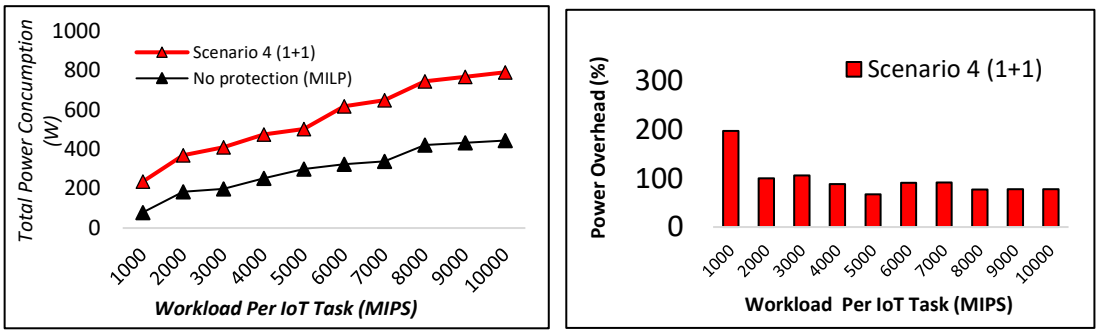


Figure 6.7 Primary and backup servers distribution in Scenario #3

6.3.4 Scenario #4: *Twenty active IoTs*

In accordance with the observations made previously, in the case of very high workload volumes, the cloud DCs seem to be predominantly favourable compared to the fog layer processing nodes, largely due to their processing efficiency and enormous computational resources they can provide. This raises the question of whether the savings introduced by distributed processing are still worthwhile with the current general purpose server specifications. Perhaps the only way the cloud's intervention lessens is by having better and more powerful processing servers located in the edge of the network.



(a) (b)

Figure 6.8 (a) Total power consumption of 1+1 sever protection in Scenario #4 compared to the baseline, (b) power overhead in percentage for 1+1 protection compared to baseline, for the same scenario.

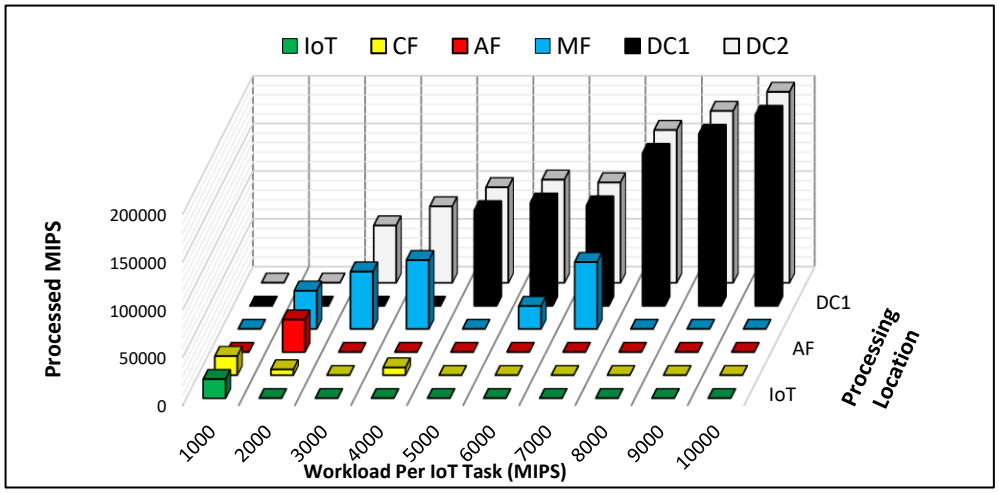


Figure 6.9 Primary and backup servers distribution in Scenario #4.

6.4 Summary

This chapter has provided an overview of resilience in the context of IoT and the different levels of protection approaches that are typically offered in terms of server protection. The original MILP model was considered for evaluation and the additional modification to the variables and equations were redefined to account for the variations introduced by resilience. The power consumption overhead due to resilience was found to be significant in cases where the workload could be hosted in the IoT layer, due to the low power consumption of local computation and avoiding the activation of ONUs. In this regard, it was suggested that device to device communication could potentially be beneficial as communication can be achieved directly between the IoT devices instead of going through the ONU.

Moreover, the observations made confirmed the relevance of the cloud datacentres during very high workload volumes. The results showed that, from an energy efficiency point of view, the fog layers had limited role to play in this case in hosting the primary and backup servers, since the processing efficiency offered by the cloud DC's compensated for the network overhead incurred to get to them. Future work can introduce additional optimisation metrics such as latency alongside energy efficiency and this can modify the optimum server placement decisions.

Chapter 7 Conclusions and Future Research Directions

This chapter outlines the main contributions that have been presented in this thesis. It also suggests possible research avenues and future directions in the area of IoT based distributed processing and energy efficiency.

8.1 Conclusions

This thesis addressed energy efficiency of distributed processing approaches for resource intensive visual-based processing services in the context of IoT. An enormous number of small-sized, intelligent and relatively powerful devices exist in the edge of the network, hence, collectively, these devices offer a massive pool of resources that can potentially ease the pressure on cloud data centres by processing sensed data close to the end-devices. In this direction, a distributed architecture has been used which is based on Passive Optical Networks (PONs) due to their suitability for high bit rate application services and their scalability which particularly fits well with massive IoT deployments. Several layers of processing were considered in the IoT-cloud continuum, along with emerging energy efficient special purpose data centres (SP-DCs) that are highly optimised to perform a specific task, hence, they can be much more efficient than their general purpose counterparts.

Chapter 5 considers two design phases, a un-capacitated one which gives insights about long term network deployment and a capacitated one that is aimed at plans for the short term network deployment. Power consumption evaluations were undertaken using realistic input data from manufactures for the devices and equipment where possible. The application resource

characteristics such as processing requirement were based on representative data extracted from related work in the literature and an online tool was used as a guide to estimate traffic demands since processing and traffic were assumed to be proportional to each other. In the capacitated case, given, non-splittable service tasks and for low workload volumes, significant energy savings of up to 90% were made compared to the conventional cloud due to local computation. However, for relatively higher workloads, the savings dropped down to 30% due to activation of the metro fog layer that utilises more powerful servers. As for the un-capacitated scenario, it was found that building too many small CPE fog servers was not a good option in the long run for high workload volumes and hence the cloud DC was used due to its server processing efficiency.

Chapter 5 extended on the work in Chapter 4 by investigating the impact of service splitting on the reduction of power consumption. It was found that, in the short term, for low workloads such as scenario #1, a single IoT with 3000 MIPS, the savings increased from 46% with no service splitting to 88% with service splitting. However as the workload increased, it was found that service splitting was only beneficial in limited circumstances such as processing parts of the given services locally in the IoT layer in order to prevent activation of additional servers in the upper layers such as the metro fog and the cloud. Hence, CPE fog and access fog layers had limited or no role to play due to their high processing inefficiency and PUE value, respectively. Chapter 5 also investigates the impact of inter-service traffic processing overhead between the subtasks of a service request. A range of synchronisation processing overhead will be covered which includes 1%, 5%, 10% and 20%.

In Chapter 6, we considered the impact of resilience on server protection and hence the original MILP model was modified to produce the results. It was found that the power consumption overhead due to resilience was significant in cases where the workload could be hosted in the IoT layer. Local computation in the baseline approach with no protection consumed negligible power compared to activating an ONU to locate the backup server to another processing location. Moreover, the observations made confirmed the relevance of the cloud datacentres during very high workload volumes. The results showed that the fog layers had limited role to play in hosting the primary and backup servers since the processing efficiency offered by the cloud DC's compensated for the network overhead incurred to get to them.

8.2 Future Research Directions

The work in this thesis has tackled the challenging task of resource management in a highly heterogeneous distributed processing architecture such as the fog/edge processing layer. The findings of this study are key to more energy efficient network design architectures for resource intensive applications and lead to the following future research directions:

1. **IoT network scale and workload characteristics**: The work in this thesis has considered a limited number of IoT devices due to the complexity of runtime which leads to large memory requirements when executing the MILP optimisation using AMPL. Moreover, results were based on a deterministic set of workloads which can be at times inaccurate measure of processing service patterns and traffic patterns in the real world. Future related work can consider larger scale IoT

networks along with the inclusion of the dynamic nature of workloads in practical circumstances.

2. **Network topology and link resilience:** In the considered work, only the tree topology of PON was considered. It would be of interest to investigate other topologies such as the bus and ring topologies in particular as they allow interconnection of ONUs at the CPE layer. Moreover, different link protection schemes in the PON access layer can be considered. Link protection is particularly important in PON-based mission-critical services, due to the point-to-multipoint feature of the PON. A single failure to OLT or a cut in feeder fibre can result in the disconnection of a large number of users.
3. **Delay in mission-critical applications:** The main goal of fog computing/edge processing is to reduce the latency of mission-critical services in order to make better decisions closer to the source. Joint delay and power consumption MILP cost functions would provide further insights into the design and development of future IoT networks if optimised jointly.
4. **Renewable energy sources at fog sites:** Incorporation of green energy at the edge of the network to focus on the reduction of CO_2 emissions is of particular interest. A MILP model for hybrid-power IoT processing applications can be utilised to maximise the use of green energy sources whilst minimising the non-green ones in the distributed processing approach.
5. **Practical implementation through development of heuristics:** The solutions obtained from the MILP models can be approximated through the use of efficient heuristic algorithms that can mimic the

behaviour of the optimal cases, hence, this can sever two purposes; a) as a validation tool for the correctness of the MILP computation and b) as a practical tool that can be implemented into hardware.

References

[1] O. Vermesan and P. Friess, *Internet of Things Applications - From Research and Innovation to Market Deployment*. 2014.

[2] F. Liang *et al.*, “A Survey on the Edge Computing for the Internet of Things,” *IEEE Access*, vol. 6, pp. 6900–6919, 2017.

[3] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[4] M. H. Asghar, “Principle Application and Vision in Internet of Things (IoT),” 2002.

[5] M. Chiang and T. Zhang, “Fog and IoT: An Overview of Research Opportunities,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, 2016.

[6] a Zanella, N. Bui, a Castellani, L. Vangelista, and M. Zorzi, “Internet of Things for Smart Cities,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, 2014.

[7] S. Zeadally, S. U. Khan, and N. Chilamkurti, “Energy-efficient networking: past, present, and future,” *J. Supercomput.*, vol. 62, no. 3, pp. 1093–1118, 2012.

[8] O. Skarlat, S. Schulte, M. Borkowski, and P. Leitner, “Resource provisioning for IoT services in the fog,” *Proc. - 2016 IEEE 9th Int. Conf. Serv. Comput. Appl. SOCA 2016*, pp. 32–39, 2016.

[9] “Which IoT applications work best with fog computing? | Network World.” [Online]. Available: https://www.networkworld.com/article/3147085/which-iot-applications-

work-best-with-fog-computing.html. [Accessed: 25-Oct-2019].

[10] R. Deng, R. Lu, C. Lai, and T. H. Luan, “Towards power consumption-delay tradeoff by workload allocation in cloud-fog computing,” *IEEE Int. Conf. Commun.*, vol. 2015–Septe, pp. 3909–3914, 2015.

[11] P. Hu, S. Dhelim, H. Ning, and T. Qiu, “Survey on fog computing: architecture, key technologies, applications and open issues,” *J. Netw. Comput. Appl.*, vol. 98, no. September, pp. 27–42, 2017.

[12] B. J. Baliga, R. W. A. Ayre, K. Hinton, R. S. Tucker, and F. Ieee, “Green Cloud Computing : Balancing Energy in Processing , Storage , and Transport,” 2011.

[13] A. P. Bianzino, C. Chaudet, D. Rossi, and J. L. Rougier, “A survey of green networking research,” *IEEE Commun. Surv. Tutorials*, vol. 14, no. 1, pp. 3–20, 2012.

[14] M. Barcelo, A. Correa, J. Llorca, A. M. Tulino, J. L. Vicario, and A. Morell, “IoT-Cloud Service Optimization in Next Generation Smart Environments,” vol. 34, no. 12, pp. 4077–4090, 2016.

[15] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[16] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, *Resource Provisioning for IoT application services in Smart Cities*. 2017.

[17] N. Gershenfeld, R. Krikorian, and D. Cohen, *Converging_Technologies_for_Smart_Environments_and_Integrated_Ecosystems*, vol. 291, no. 4. 2004.

[18] S. Sarkar and S. Misra, “Theoretical modelling of fog computing : a green computing paradigm to support IoT applications,” vol. 5, pp. 23–29, 2016.

[19] “Gartner: Internet of Things Installed Base Will Grow to 26 Billion Units By 2020 | Ministry of Communications and Information Technology.” [Online]. Available: https://www.mcit.gov.sa/en/media-center/news/91424. [Accessed: 24-Oct-2019].

[20] U. Nations, D. of Economic, S. Affairs, and P. Division, “World Population Prospects 2019 Highlights.”

[21] ITU, “ITU Internet Reports - Executive Summary,” 2005.

[22] European Comission, *Internet of Things in 2020: A roadmap for the future*. 2008.

[23] D. E. Zheng and W. A. Carter, “Leveraging the Internet of Things for a More Efficient and Effective Military A Report of the CSIS Strategic Technologies Program,” 2015.

[24] L. Tian, H. Wang, Y. Zhou, and C. Peng, “Video big data in smart city: Background construction and optimization for surveillance video processing,” *Futur. Gener. Comput. Syst.*, vol. 86, pp. 1371–1382, 2018.

[25] S. Sarkar, S. Chatterjee, and S. Misra, “Assessment of the Suitability of Fog Computing in the Context of Internet of Things,” *IEEE Trans. Cloud Comput.*, vol. PP, no. 99, pp. 1–1, 2015.

[26] L. Tan, “Future internet: The Internet of Things,” *2010 3rd Int. Conf. Adv. Comput. Theory Eng.*, pp. V5-376-V5-380, 2010.

[27] M. Wu, T. J. Lu, F. Y. Ling, J. Sun, and H. Y. Du, “Research on the architecture of Internet of Things,” in *ICACTE 2010 - 2010 3rd International Conference on Advanced Computer Theory and Engineering, Proceedings*, 2010, vol. 5.

[28] M. Aazam, I. Khan, A. A. Alsaffar, and E. N. Huh, “Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved,” *Proc. 2014 11th Int. Bhurban Conf. Appl. Sci. Technol. IBCAST 2014*, pp. 414–419, 2014.

[29] N. Kaur and S. K. Sood, “An Energy-Efficient Architecture for the Internet of Things (IoT),” *IEEE Syst. J.*, no. 99, pp. 1–10, 2015.

[30] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, “Future internet: The internet of things architecture, possible applications and key challenges,” *Proc. - 10th Int. Conf. Front. Inf. Technol. FIT 2012*, pp. 257–260, 2012.

[31] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, “Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures,” *Commun. Surv. Tutorials, IEEE*, vol. 13, no. 2, pp. 223–244, 2011.

[32] F. Lin, Q. Liu, X. Zhou, Y. Chen, and D. Huang, “Cooperative differential game for model energy-bandwidth efficiency tradeoff in the Internet of Things,” *China Commun.*, vol. 11, no. 1, pp. 92–102, 2014.

[33] T. Langford, Q. Gu, A. Rivera-Longoria, and M. Guirguis, “Collaborative computing on-demand: Harnessing mobile devices in executing on-the-fly jobs,” *Proc. - IEEE 10th Int. Conf. Mob. Ad-Hoc*

Sens. Syst. MASS 2013, no. 4, pp. 342–350, 2013.

[34] E. Miluzzo, R. Cáceres, and Y. Chen, “Vision : mClouds – Computing on Clouds of Mobile Devices,” *MCS ’12 Proc. third ACM Work. Mob. cloud Comput. Serv.*, pp. 9–13, 2012.

[35] M. Silverio-Fernández, S. Renukappa, and S. Suresh, “What is a smart device? - a conceptualisation within the paradigm of the internet of things.”

[36] “iPhone 11 - Apple (UK).” [Online]. Available: https://www.apple.com/uk/iphone-11/. [Accessed: 24-Oct-2019].

[37] R. Want, B. N. Schilit, and S. Jenson, “Enabling the Internet of Things,” 2015.

[38] S. Hamid Mohamed, S. Member, T. E. El-Gorashi, J. M. Elmirghani, and S. Member, “A Survey of Big Data Machine Learning Applications Optimization in Cloud Data Centers and Networks.”

[39] N. Bizanis and F. A. Kuipers, “SDN and Virtualization Solutions for the Internet of Things: A Survey,” *IEEE Access*, vol. 4, pp. 5591–5606, 2016.

[40] A. R. Biswas and R. Giaffreda, “IoT and Cloud Convergence: Opportunities and Challenges,” *2014 IEEE World Forum Internet Things*, pp. 375–376, 2014.

[41] C. Gray, R. Ayre, K. Hinton, and R. S. Tucker, “Power consumption of IoT access network technologies,” *2015 IEEE Int. Conf. Commun. Work.*, pp. 2818–2823, 2015.

[42] T. G. Orphanoudakis, C. Matrakidis, and A. Stavdas, “Next generation

optical network architecture featuring distributed aggregation, network processing and information routing," *2014 Eur. Conf. Networks Commun.*, pp. 1–5, 2014.

[43] F. Neri and J. M. Finochietto, "Passive Optical Networks," *Tutorial*, vol. n/a.

[44] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols and Applications," *IEEE Commun. Surv. Tutorials*, vol. PP, no. 99, pp. 1–1, 2015.

[45] B. Nathali Silva, M. Khan, and K. Han, "Internet of Things: A Comprehensive Review of Enabling Technologies, Architecture, and Challenges," *IETE Tech. Rev.*, vol. 35, no. 2, pp. 205–220, 2018.

[46] M. Gigli and S. Koo, "Internet of Things: Services and Applications Categorization," *Adv. Internet Things*, vol. 01, no. 02, pp. 27–31, 2011.

[47] "IoT Gateway || ez enRoute Ltd., an IoT Company." [Online]. Available: https://www.ezenroute.com/iot-gateway.html. [Accessed: 20-Oct-2019].

[48] L. Wang, "Processing Distributed Internet of Things Data in Clouds," 2015.

[49] "Applications of IoT in Manufacturing Plants - The Manufacturer." [Online]. Available: https://www.themanufacturer.com/articles/applications-iot-manufacturing-plants/. [Accessed: 24-Oct-2019].

[50] "Safety on smart motorways - GOV.UK." [Online]. Available:

https://www.gov.uk/government/news/safety-on-smart-motorways. [Accessed: 24-Oct-2019].

[51] N. Callaghan, T. Avery, and M. Mulville, “Smart" Motorway Innovation for Achieving Greater Safety and Hard Shoulder Management,” 2017.

[52] “Q and A: Hiring the messenger - Highways England’s telecommunications - The Transport Network.” [Online]. Available: https://www.transport-network.co.uk/Q-and-A-Hiring-the-messenger---Highways-Englands-telecommunications/15424. [Accessed: 24-Oct-2019].

[53] “Motorway Midas Loop | Bridgepoint Road Markings.” [Online]. Available: http://www.bridgepointroadmarkings.com/services/inductive-loop-cutting/motorway-midas-loop/. [Accessed: 24-Oct-2019].

[54] S. Tomovic, K. Yoshigoe, I. Maljevic, and I. Radusinovic, “Software-Defined Fog Network Architecture for IoT,” *Wirel. Pers. Commun.*, vol. 92, no. 1, pp. 181–196, 2017.

[55] S. Y. Jang, Y. Lee, B. Shin, and D. Lee, “Application-aware IoT camera virtualization for video analytics edge computing,” *Proc. - 2018 3rd ACM/IEEE Symp. Edge Comput. SEC 2018*, pp. 132–144, 2018.

[56] G. Gan, Z. Lu, and J. Jiang, “Internet of things security analysis,” in *2011 International Conference on Internet Technology and Applications, iTAP 2011 - Proceedings*, 2011.

[57] B. Cheng, G. Solmaz, F. Cirillo, E. Kovacs, K. Terasawa, and A. Kitazawa, “FogFlow: Easy Programming of IoT Services Over Cloud and Edges for Smart Cities,” *IEEE Internet Things J.*, vol. 5, no. 2, pp.

696–707, 2018.

[58] X. Chen, “Decentralized computation offloading game for mobile cloud computing,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, 2015.

[59] S. F. Abedin, M. G. R. Alam, R. Haw, and C. S. Hong, “A system model for energy efficient green-IoT network,” *2015 Int. Conf. Inf. Netw.*, pp. 177–182, 2015.

[60] “Internet of Things Data To Top 1.6 Zettabytes by 2020 -- Campus Technology.” [Online]. Available: https://campustechnology.com/articles/2015/04/15/internet-of-things-data-to-top-1-6-zettabytes-by-2020.aspx. [Accessed: 12-Oct-2019].

[61] “Self-driving cars could create 1GB of data a second.”

[62] a. L. Brandão, “On the Energy Consumption of Relay Networks,” *Veh. Technol. Conf. Fall (VTC 2010-Fall), 2010 IEEE 72nd*, no. 1, pp. 9–11, 2010.

[63] S. B. Nath, H. Gupta, S. Chakraborty, and S. K. Ghosh, “A Survey of Fog Computing and Communication: Current Researches and Future Directions,” no. i, pp. 1–47, 2018.

[64] A. Yousefpour *et al.*, “All one needs to know about fog computing and related edge computing paradigms: A complete survey,” *J. Syst. Archit.*, no. December 2018, 2019.

[65] T. Taleb, K. Samdanis, B. Mada, H. F.-… S. & Tutorials, and U. 2017, “On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration,”

Ieeexplore.Ieee.Org, vol. 19, no. 3, pp. 1657–1681, 2017.

[66] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and A. Polakos, “A Comprehensive Survey on Fog Computing: State- of-the-art and Research Challenges.”

[67] M. Satyanarayanan, P. Bahl, R. Cáceres, and N. Davies, “The case for VM-based cloudlets in mobile computing,” *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, 2009.

[68] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in *MCC’12 - Proceedings of the 1st ACM Mobile Cloud Computing Workshop*, 2012, pp. 13–15.

[69] “Industrial Internet Consortium.” [Online]. Available: https://www.iiconsortium.org/index.htm. [Accessed: 24-Oct-2019].

[70] Z. Á. Mann, “Optimization Problems in Fog and Edge Computing,” in *Fog and Edge Computing*, John Wiley & Sons, Inc., 2019, pp. 103–121.

[71] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker, “Fog computing may help to save energy in cloud computing,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1728–1739, 2016.

[72] V. Angelakis, I. Avgouleas, N. Pappas, E. Fitzgerald, and D. Yuan, “Allocation of Heterogeneous Resources of an IoT Device to Flexible Services,” *IEEE Internet Things J.*, vol. 3, no. 5, pp. 691–700, Oct. 2016.

[73] C. C. Byers, “Architectural Imperatives for Fog Computing: Use Cases, Requirements, and Architectural Techniques for Fog-Enabled IoT

Networks,” *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 14–20, 2017.

[74] D. Perez Abreu, K. Velasquez, M. Curado, and E. Monteiro, “A resilient Internet of Things architecture for smart cities,” vol. 72, pp. 19–30, 2017.

[75] J. Pan and J. McElhannon, “Future Edge Cloud and Edge Computing for Internet of Things Applications,” *IEEE Internet Things J.*, vol. 5, no. 1, pp. 1–1, 2017.

[76] F. Idzikowski, L. Chiaraviglio, A. Cianfrani, J. Vizcaino, M. Polverini, and Y. Ye, “A Survey on Energy-Aware Design and Operation of Core Networks,” *IEEE Commun. Surv. Tutorials*, no. c, pp. 1–1, 2016.

[77] E. Sturzinger, M. Tornatore, and B. Mukherjee, “Application-Aware Resource Provisioning in a Heterogeneous Internet of Things.”

[78] M. Taneja and A. Davy, “Resource aware placement of IoT application modules in Fog-Cloud Computing Paradigm,” *Proc. IM 2017 - 2017 IFIP/IEEE Int. Symp. Integr. Netw. Serv. Manag.*, pp. 1222–1228, 2017.

[79] V. B. C. Souza, W. Ramirez, X. Masip-Bruin, E. Marin-Tordera, G. Ren, and G. Tashakor, “Handling service allocation in combined Fog-cloud scenarios,” *2016 IEEE Int. Conf. Commun. ICC 2016*, pp. 0–4, 2016.

[80] H. Guo, J. Liu, and H. Qin, “Collaborative Mobile Edge Computation Offloading for IoT over Fiber-Wireless Networks,” *IEEE Netw.*, vol. 32, no. 1, pp. 66–71, Jan. 2018.

[81] Y. Xiao, M. Noreikis, and A. Yla-Jaaiski, “QoS-oriented capacity

planning for edge computing," in *IEEE International Conference on Communications*, 2017.

[82] S. Mondal, G. Das, and E. Wong, "A Novel Cost Optimization Framework for Multi-Cloudlet Environment over Optical Access Networks," in *2017 IEEE Global Communications Conference, GLOBECOM 2017 - Proceedings*, 2017, vol. 2018–January, pp. 1–6.

[83] M. Taheri and N. Ansari, "A feasible solution to provide cloud computing over optical networks," *IEEE Netw.*, vol. 27, no. 6, pp. 31–35, 2013.

[84] S. H. S. Newaz, W. Susanty Binti Haji Suhaili, G. M. Lee, M. R. Uddin, A. F. Y. Mohammed, and J. K. Choi, "Towards realizing the importance of placing fog computing facilities at the central office of a PON," *Int. Conf. Adv. Commun. Technol. ICACT*, no. i, pp. 152–157, 2017.

[85] A. M. Al-Salim, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "Energy Efficient Big Data Networks: Impact of Volume and Variety," *IEEE Trans. Netw. Serv. Manag.*, vol. 15, no. 1, pp. 458–474, Mar. 2018.

[86] A. M. Al-Salim, T. E. H. El-Gorashi, A. Q. Lawey, and J. M. H. Elmirghani, "Greening big data networks: Velocity impact," *IET Optoelectron.*, vol. 12, no. 3, pp. 126–135, Jun. 2018.

[87] B. G. Bathula, M. Alresheedi, and J. M. H. Elmirghani, "Energy Efficient Architectures for Optical Networks."

[88] B. G. Bathula and J. M. H. Elmirghani, "Energy efficient Optical Burst

Switched (OBS) networks," in *2009 IEEE Globecom Workshops, Gc Workshops 2009*, 2009.

[89] T. E. H. El-Gorashi, X. Dong, and J. M. H. Elmirghani, "Green optical orthogonal frequency-division multiplexing networks," *IET Optoelectron.*, vol. 8, no. 3, pp. 137–148, 2014.

[90] H. M. M. Ali, T. E. H. El-Gorashi, A. Q. Lawey, and J. M. H. Elmirghani, "Future Energy Efficient Data Centers with Disaggregated Servers," *J. Light. Technol.*, vol. 35, no. 24, pp. 5361–5380, Dec. 2017.

[91] A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "BitTorrent content distribution in optical networks," *J. Light. Technol.*, vol. 32, no. 21, pp. 3607–3623, Nov. 2014.

[92] N. I. Osman, T. El-Gorashi, L. Krug, and J. M. H. Elmirghani, "Energy-efficient future high-definition TV," *J. Light. Technol.*, vol. 32, no. 13, pp. 2364–2381, Jul. 2014.

[93] M. Musa, T. Elgorashi, and J. Elmirghani, "Energy efficient survivable IP-Over-WDM networks with network coding," *J. Opt. Commun. Netw.*, vol. 9, no. 3, pp. 207–217, Mar. 2017.

[94] A. N. Al-Quzweeni, A. Q. Lawey, T. E. H. Elgorashi, and J. M. H. Elmirghani, "Optimized Energy Aware 5G Network Function Virtualization," *IEEE Access*, vol. 7, pp. 44939–44958, 2019.

[95] M. S. Hadi, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "Patient-Centric Cellular Networks Optimization Using Big Data Analytics," *IEEE Access*, vol. 7, pp. 49279–49296, 2019.

[96] Z. T. Al-Azez, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, “Energy Efficient IoT Virtualization Framework With Peer to Peer Networking and Processing,” *IEEE Access*, vol. 7, pp. 50697–50709, 2019.

[97] H. Q. Al-Shammari, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, “Service Embedding in IoT Networks,” *IEEE Access*, vol. 8, pp. 2948–2962, 2020.

[98] A. Shehabi *et al.*, “United States Data Center Energy Usage Report,” *Lawrence Berkeley Natl. Lab. Berkeley, CA, Tech. Rep.*, no. June, pp. 1–66, 2016.

[99] “‘Skynet’, China’s massive video surveillance network | Abacus.” [Online]. Available: https://www.abacusnews.com/who-what/skynet-chinas-massive-video-surveillance-network/article/2166938. [Accessed: 25-Oct-2019].

[100] “In Your Face: China’s all-seeing state - BBC News.” [Online]. Available: https://www.bbc.co.uk/news/av/world-asia-china-42248056/in-your-face-china-s-all-seeing-state. [Accessed: 25-Oct-2019].

[101] D. Barrett, “One surveillance camera for every 11 people in Britain, says CCTV survey,” 2013. [Online]. Available: https://www.telegraph.co.uk/technology/10172298/One-surveillance-camera-for-every-11-people-in-Britain-says-CCTV-survey.html. [Accessed: 19-Mar-2019].

[102] Y. Luo, F. Effenberger, and M. Sui, “Cloud computing provisioning

over passive optical networks," in *2012 1st IEEE International Conference on Communications in China, ICCC 2012*, 2012, pp. 255–259.

[103] A. Yousefpour, G. Ishigaki, and J. P. Jue, "Fog Computing: Towards Minimizing Delay in the Internet of Things," *Proc. - 2017 IEEE 1st Int. Conf. Edge Comput. EDGE 2017*, pp. 17–24, 2017.

[104] W. Zhang, B. Lin, Q. Yin, and T. Zhao, "Infrastructure deployment and optimization of fog network based on MicroDC and LRPON integration."

[105] L. Kazovsky, S. W. Wong, T. Ayhan, K. M. Albeyoglu, M. R. N. Ribeiro, and A. Shastri, "Hybrid optical-wireless access networks," *Proc. IEEE*, vol. 100, no. 5, pp. 1197–1225, 2012.

[106] F. Jalali, S. Khodadustan, C. Gray, K. Hinton, and F. Suits, "Greening IoT with Fog: A Survey," in *Proceedings - 2017 IEEE 1st International Conference on Edge Computing, EDGE 2017*, 2017, pp. 25–31.

[107] P. Chołda and P. Jaglarz, "Optimization/simulation-based risk mitigation in resilient green communication networks," *J. Netw. Comput. Appl.*, vol. 59, pp. 134–157, Jan. 2016.

[108] Y. Zhang, M. Tornatore, P. Chowdhury, and B. Mukherjee, "Energy optimization in IP-over-WDM networks," *Opt. Switch. Netw.*, vol. 8, no. 3, pp. 171–180, 2011.

[109] C. Delgado, J. R. Gállego, M. Canales, J. Ortín, S. Bousnina, and M. Cesana, "On optimal resource allocation in virtual sensor networks," *Ad Hoc Networks*, vol. 50, pp. 23–40, Nov. 2016.

[110] D. Meisner, B. T. Gold, and T. F. Wenisch, *PowerNap: Eliminating Server Idle Power*. .

[111] “Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper - Cisco.” [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html. [Accessed: 26-Oct-2019].

[112] M. Y. Arslan, I. Singh, S. Singh, H. V. Madhyastha, K. Sundaresan, and S. V. Krishnamurthy, “Computing while charging: Building a Distributed Computing Infrastructure Using Smartphones,” *8th Int. Conf. Emerg. Netw. Exp. Technol.*, pp. 193–204, 2012.

[113] L. Nonde, T. E. H. El-gorashi, and J. M. H. Elmirghani, “Energy Efficient Virtual Network Embedding for Cloud Networks,” *Energy Effic. Virtual Netw. Embed. Cloud Networks*, vol. 33, no. 9, pp. 1828–1849, 2015.

[114] A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, “Distributed energy efficient clouds over core networks,” *J. Light. Technol.*, vol. 32, no. 7, pp. 1261–1281, 2014.

[115] J. M. H. Elmirghani *et al.*, “GreenTouch GreenMeter Core Network Energy Efficiency Improvement Measures and Optimization [Invited],” *IEEE/OSA J. Opt. Commun. Netw.*, vol. 10, no. 2, 2018.

[116] “Cisco Industrial Benchmark,” 2016. [Online]. Available: https://www.cisco.com/c/dam/global/da_dk/assets/docs/presentations/vBootcamp_Performance_Benchmark.pdf. [Accessed: 16-Mar-2018].

[117] “Tensor Cores in NVIDIA Volta Architecture | NVIDIA.” [Online]. Available: https://www.nvidia.com/en-gb/data-center/tensorcore/. [Accessed: 16-Oct-2019].

[118] “Intel® Xeon® Processor E5-2680 (20M Cache, 2.70 GHz, 8.00 GT/s Intel® QPI) Product Specifications.” [Online]. Available: https://ark.intel.com/content/www/us/en/ark/products/64583/intel-xeon-processor-e5-2680-20m-cache-2-70-ghz-8-00-gt-s-intel-qpi.html. [Accessed: 26-Oct-2019].

[119] “Intel® Xeon® Processor X5675 (12M Cache, 3.06 GHz, 6.40 GT/s Intel® QPI) Product Specifications.” [Online]. Available: https://ark.intel.com/content/www/us/en/ark/products/52577/intel-xeon-processor-x5675-12m-cache-3-06-ghz-6-40-gt-s-intel-qpi.html. [Accessed: 26-Oct-2019].

[120] “Intel® Xeon® Processor E5-2420 (15M Cache, 1.90 GHz, 7.20 GT/s Intel® QPI) Product Specifications.” [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. [Accessed: 26-Oct-2019].

[121] “FAQs - Raspberry Pi Documentation.” [Online]. Available: https://www.raspberrypi.org/documentation/faqs/. [Accessed: 26-Oct-2019].

[122] “Raspberry Pi 3: Specs, benchmarks & testing — The MagPi magazine.” [Online]. Available: https://magpi.raspberrypi.org/articles/raspberry-pi-3-specs-benchmarks. [Accessed: 26-Oct-2019].

[123] “Raspberry Pi Zero W (Wireless) | The Pi Hut.” [Online]. Available: https://thepihut.com/products/raspberry-pi-zero-w. [Accessed: 26-Oct-2019].

[124] D-Link, “8-Port 10/100 Switch,” 2010.

[125] “Cisco Network Convergence System 5500 Series Modular Chassis Data Sheet - Cisco.” [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/routers/network-convergence-system-5500-series/datasheet-c78-736270.html. [Accessed: 26-Oct-2019].

[126] “Cisco Nexus 9300-FX Series Switches Data Sheet - Cisco.” [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-9000-series-switches/datasheet-c78-742284.html. [Accessed: 26-Oct-2019].

[127] E. Bash, *Routing, Flow, and Capacity Design in Communication and Computer Networks*, vol. 1. 2015.

[128] K. Benson, “Enabling resilience in the Internet of Things,” in *2015 IEEE International Conference on Pervasive Computing and Communication Workshops, PerCom Workshops 2015*, 2015, pp. 230–232.

[129] B. Bangerter, S. Talwar, R. Arefi, and K. Stewart, “Networks and devices for the 5G era,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 90–96, 2014.