Joint Functional Splitting and Content Placement for Green Hybrid CRAN

AJAY SRIRAM
# Contents

1 Introduction .................................................. 7  
1.1 Thesis Structure ........................................ 8

2 Literature Survey ............................................ 9

3 Network Architecture ...................................... 11  
3.1 Proposed Architecture .................................. 11  
3.2 Reference Architecture ................................ 12  
3.3 Functional Split Model .................................. 13  
3.4 Content Placement ....................................... 16

4 Problem Statement .......................................... 17  
4.1 Objective ................................................ 17  
4.2 Input Parameters ......................................... 18  
4.3 Optimization Variables ................................ 20  
4.4 Constraints ............................................... 21  
4.5 Methodology ............................................. 23

5 Simulation ..................................................... 25  
5.1 System Model ............................................. 25  
5.1.1 Assumptions .......................................... 26  
5.1.2 Delay Model .......................................... 27  
5.1.3 Power Model ......................................... 28  
5.2 Simulation Parameters .................................. 29  
5.3 Simulation Results ...................................... 29
## CONTENTS

6 Discussions 36
  6.1 Conclusion ......................................................... 36
  6.2 Future Work ......................................................... 37

Bibliography 38
## List of Figures

3.1 Hybrid virtualized RAN architecture ........................................... 12
3.2 Functional split model ................................................................. 13
3.3 Baseband processing chain adapted from [9] .................................... 14
3.4 Functional Split architecture in H-CRAN adapted from [14] ................. 15

5.1 Delay model of the proposed architecture ....................................... 27
5.2 Power model of the proposed architecture ..................................... 28
5.3 Plot between total network power consumption and percentage of active users ... 30
5.4 Hitrate against number of active users .......................................... 31
5.5 Average delay experienced by the users against number of active users .... 32
5.6 Midhaul bandwidth consumption against percentage of active users .......... 33
5.7 Impact of delay threshold on total power consumption ........................ 34
5.8 Impact of delay threshold on total required edge cloud cache capacity .... 35
# Abbreviation

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRAN</td>
<td>Cloud Radio Access Network</td>
</tr>
<tr>
<td>H-CRAN</td>
<td>Hybrid - Cloud Radio Access Network</td>
</tr>
<tr>
<td>RU</td>
<td>Radio Unit</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>CP</td>
<td>Cell Processing</td>
</tr>
<tr>
<td>UP</td>
<td>User Processing</td>
</tr>
<tr>
<td>DU</td>
<td>Digital Unit</td>
</tr>
<tr>
<td>CC</td>
<td>Central Cloud</td>
</tr>
<tr>
<td>EC</td>
<td>Edge Cloud</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>ILP</td>
<td>Integer Linear Program</td>
</tr>
<tr>
<td>MVNO</td>
<td>Mobile Virtual Network Operator</td>
</tr>
<tr>
<td>PON</td>
<td>Passive Optical Network</td>
</tr>
<tr>
<td>TWDM-PON</td>
<td>Time-wavelength Division Multiplexing PON</td>
</tr>
<tr>
<td>ONU</td>
<td>Optical Network Unit</td>
</tr>
<tr>
<td>LC</td>
<td>Line Card</td>
</tr>
<tr>
<td>MINLP</td>
<td>Mixed Integer Non Linear Program</td>
</tr>
<tr>
<td>MILP</td>
<td>Mixed Integer Linear Program</td>
</tr>
</tbody>
</table>
Acknowledgement

This thesis project was carried out at Radio Systems Laboratory, wireless@KTH under the department of Communication Systems, KTH, Sweden. This thesis work was supported by EU Celtic-Plus Project SooGREEN: Service Oriented Optimization of Green Mobile Networks.

I would like to thank my supervisors Abdulrahman Alabbasi, Meysam Masoudi and Mohammad Galal Khafagy for providing me complete support and motivation throughout the project. I would also like to thank Alberto Conte of Nokia Bell Labs for supporting me remotely and giving me valuable comments over the work. I would like to thank my supervisor and examiner Cicek Cavdar for giving me the opportunity to carry out this interesting thesis and also for steering this research in the right direction. I would like to thank all my colleagues at Wireless@KTH for an amazing work environment to carry out my thesis work.

At the end I would like to express my biggest gratitude to my loved ones, my family and friends, who have supported me throughout the entire process. I will be grateful forever for your love.

AJAY SRIRAM
Stockholm, Sweden
Abstract

A hybrid cloud radio access network (H-CRAN) architecture has been proposed to alleviate the midhaul capacity limitation in C-RAN. In this architecture, functional splitting is utilized to distribute the processing functions between a central cloud and edge clouds. The flexibility of selecting specific split point enables the H-CRAN designer to reduce midhaul bandwidth, or reduce latency, or save energy, or distribute the computation task depending on equipment availability. Meanwhile, techniques for caching are proposed to reduce the content delivery latency and the required bandwidth. However, caching imposes new constraints on the functional splitting. In this study, considering H-CRAN, a constraint programming problem is formulated to minimize the overall power consumption by selecting the optimal functional split point and content placement, taking into account the content access delay constraint. We also investigate the trade-off between the overall power consumption and occupied midhaul bandwidth in the network. Our results demonstrate that functional splitting together with enabling caching at edge clouds reduces not only content access delays but also fronthaul bandwidth consumption but at the expense of higher power consumption.
Abstrakt

Chapter 1

Introduction

Since mobile traffic has been growing exponentially and also users expecting the requested services with much less delay, 5G networks have an aim of accommodating more traffic with much lower latency. However, in achieving that, the cost and energy consumption should be sustainable [1]. In recent years, cloud radio access network (CRAN) architecture has been proposed as a solution to reduce the power consumption and cost [2, 3]. In hybrid CRAN, digital units (DU) are placed at central clouds (CC) and edge cloud (EC). The placement of DUs at CC and EC enables the sharing of computational resources. Previous research studies have shown the energy saving merit of CRAN architecture as in [4]. The communication baseband processing can be modeled as a chain of cell processing and its corresponding user processing. These baseband processing functions could either be processed completely at the central cloud, edge cloud or distributed among CC or EC. If all the baseband processing functions are performed at the CC, fronthaul bandwidth and transmission delay requirements become difficult to satisfy as all the functions will be pushed to CC.

To relax the fronthaul constraints in CRAN, the functional split technique has been proposed to split the baseband processing chain at several conceivable points. This technique divides the cloud in two hierarchical layers: CC and EC [6]-[11], thus, a dual-site processing is established. Now, since the DU functions are split into two sites, the part towards the radio units are placed at edge cloud for partial baseband processing, and the remaining processing is taken place at the central cloud. The edge and central clouds are connected via an optical fibre link which is termed herein as “midhaul” link [12]. The main motivation of dual-site processing are:

- By partial baseband processing at EC, bandwidth requirement can be significantly relaxed for the midhaul link.
- By equipping the EC with processors, to support the computational and content caching capabilities, traffic load can be terminated at EC relieving traffic load in midhaul.

Intuitively, if more baseband processing functions are centralized at the CC, higher power saving
can be achieved, whereas, the midhaul bandwidth requirement and the incurred user latency will increase. On the contrary, placing more processing functions at the EC may lead to higher power consumption but lower midhaul bandwidth consumption. With regards to reducing the end-to-end delay of user, significant research work has been done on content placement. Delay-sensitive content placement has been proposed in the literatures for improving the quality of service (QoS) of user in terms of the experienced delay. The user’s requested contents could be potentially placed either at EC or CC. Based upon the user’s delay threshold, the user could fetch the content from respective EC and CC. Several research works have been carried out in solving the cache placement problem separately. Fetching the content from EC and CC has dependency over the flexible functional split which is not addressed earlier. With regards to this, the interplay between power consumption, occupied midhaul bandwidth, and traffic latency needs to be investigated when content is placed at the EC, CC, or with functional splitting adopted to distribute content among them.

In this thesis work, network power consumption is minimized, by formulating a joint problem of content placement and flexible functional split in H-CRAN networks as constraint programming [5]. In the optimization problem, each content should be delivered to the user within a delay threshold. Also, other constraints corresponding to the functional split implementation have been taken into account. Moreover, we model the delay and power consumption network components from cloud to the user. Finally the trade off between the total system power consumption and bandwidth has been investigated.

1.1 Thesis Structure

Going into the thesis, chapter 2 gives the details about the research work that’s been done in CRAN, flexible functional split and content placement. Following that, complete network architecture is described in chapter 3. Then, we define our problem statement along with objective, constraints and methodology in chapter 4. Chapter 5, briefly describes the system model along with delay and power model used in the thesis. Later the Simulation are explained with appropriate plots. Finally, the overall conclusion of the thesis and the future scope of this thesis has been discussed.
Chapter 2

Literature Survey

The large demand on higher capacity encourages the researchers to explore on ultra dense radio networks requiring a centralized processing of the communication functions processing which can be realized by Cloud Radio Access Networks (CRAN) architecture.

In [13], an end-to-end delay model is been proposed for different functional split options. For the model considered, the different functional split options is influenced by different delay requirements thus affecting the system’s power consumption. The proposed delay model aims to reduce the system’s power consumption and bandwidth consumption. The results show that for every required functional split, to achieve minimum delay, is surely influenced by the processing power ratio of the processing units at edge clouds and central cloud. Furthermore in [14], a joint optimization problem is proposed and solved the impact of different functional split options thus solving the problem of interplay between the power and bandwidth consumption in Hybrid-Cloud Radio Access Network (H-CRAN). Weighted sum of the power and bandwidth is modeled and the main objective was to minimize the weighted sum. The main trade off between the total power consumption and midhaul bandwidth for the optimal function placement is solved.

In [15], all possible constraints and outline application of the flexible RAN centralization are discussed. In their author’s vision, flexible functional split is a main enabler for the future generation of cellular networks [16]. Various functional split options and their benefits are discussed more elaborately along with its challenges. For wide applications, it is shown that the requirement of C-RAN on backhaul capabilities must be relaxed. In [17], a dynamic virtual network embedded problem (VNE) is been formalized for supporting different functional split for 5G networks. The problem is then formulated as an integer linear programming (ILP), where mobile virtual network operators (MVNOs) send virtual request that are dynamically embedded by infrastructure providers. The main objective described is to select the optimal functional split options that jointly minimizes the fronthaul bandwidth and inter-cells interference. Later, a comparison is made between the performance and efficiency of formulated ILP and heuristic algorithms. Further, its been stated that
flexible functional split in RAN provides opportunity in exploration of complex algorithms designed to reduce the inter-cell interference. Based upon certain level of inter-cell interference, specific functional split would be more reliable and efficient. The processing requirements of the edge clouds, bandwidth requirements change for every possible functional split, stating that by employing correct functional split for each small cells, significant benefits are possible. This supports our aim saying that an optimal functional split could certainly benefit in terms of lowering the energy consumption.

Caching is also another approach that may not only relieve the fronthaul congestion but also can reduce the content delivery latency. In caching, popular contents are cached into a place closer to the users, e.g., in the EC, allowing user content demands to be accommodated more easily and quickly. Since content access delay is an important factor in caching problems, various algorithms and techniques have been proposed to incur lower latencies. The challenges, paradigm, and potential solutions for caching are discussed in [19]. In [18], cooperative hierarchical caching has been proposed to minimize the content access delay and boost the quality-of-experience (QoE) for end users. Cloud cache is been introduced as new layer in RAN hierarchy, building bridge between edge-based and core-based caching schemes. A low complex, heuristic cache management algorithm is been proposed comprising of proactive cache placement/distribution and reactive cache replacement. The results show that when the contents are distributed among the edge and central cloud, the total avg experienced delay out performs edge only and center only options. In [21], the authors proposed caching algorithms to optimize the content caching locations and hence reduce the delivery delay. In [22] the authors presented a caching structure and proposed a cooperative multicast-aware caching strategy to reduce the average latency of delivering content. In this study, the focus is more on the users’ QoE and the content access latency.
Chapter 3

Network Architecture

3.1 Proposed Architecture

The proposed hybrid CRAN architecture employs dual-site processing in virtualized CRAN, where the DUs are deployed at both CC and EC. By employing DU at both EC and CC, the baseband processing can be flexibly provisioned by a chain of virtualized function for RU or even for a user equipment, while the traffic is transported through the network. The architecture is as shown in Figure 3.1.

H-CRAN is a three-layer architecture, consisting of a cell layer (the coverage region of RU is referred to as a “cell”), EC layer and CC layer. The cell layer consists of cells, each serving several UEs. A group of cells are connected to the edge cloud as an aggregation point. The fronthaul between the cell and edge cloud can be implemented using a short fiber or wireless links, e.g. mm-Wave links or free-space optical links. Currently, for this project the link between the RS and edge cloud is considered to be the mm-Wave links. The edge clouds can be connected to the central cloud via midhaul using various technologies, from expensive dark fibre or TON solutions, to cost-efficient PON families or other Ethernet based technologies. The midhaul technology considered in this study is time-wavelength division multiplexing PON (TWDM-PON), and each midhaul link is a wavelength channel, needing an optical network unit (ONU) at edge cloud and a line card (LC) at central cloud as transceivers. EC and CC are equipped with DUs which are responsible for processing the virtualized functions, because their computational resources can be virtualized and shared by any connected RUs. For example, the traffic from the cell can be partially processed at edge cloud so that the midhaul bandwidth requirement can be relaxed, then the remaining processing could be done at the central cloud. However, the processing at edge cloud is less efficient as the number of DUs accommodated at the edge cloud is less than that of in central cloud.
Along with it, the storage capacity of edge cloud is also less compared to the storage capacity of central cloud. The flexibility of the functional processing at edge and central cloud also depends on the placement of the requested file. Hence, sharing infrastructure equipment based on optimal placement of requested file creates trade off between the power consumed, midhaul bandwidth and delay experienced. Then our quest becomes, whether to centralize our baseband processing at edge cloud, central cloud, or be flexible.

### 3.2 Reference Architecture

The following two, fully-centralized and fully distributed reference architecture with no functional splitting are used as a baseline for performance evaluation purposes.

1. When all the requested files are placed in EC, then all the baseband processing must be placed at the EC, and the connection from EC and CC is provided by backhaul. Since all the processing takes place at the EC, more power is consumed but on the other hand there is very less requirement of the bandwidth.

2. When all the requested files are placed at the CC, it gives a flexibility in optimal functional split. If all the functions are centralized at the CC, then less power is consumed but eats away more midhaul bandwidth.
3.3 Functional Split Model

We model the functional split of baseband processing chain for a cell and its corresponding UEs as shown in Figure 3.2. The baseband processing for a cell and its corresponding UEs is modelled as chain of functions consisting of $m$ cell processing (CP) and $n$ user processing (UP) functions. CPs are set of functions associated with the physical layer and dedicated for the processing of signals from the cells, when signals of UEs are multiplexed. For example, CPs include iFFT, cyclic prefix, and resource mapping etc. The cell processing will be terminated at CPm, and then the signals will be demultiplex as multiple signal streams, each belonging to a UE. Then, UPs is a sequence of functions that will continue to process the signal streams per UE basis which includes antenna mapping, forward error correction etc and other layer2 and layer3 functions. The simplicity model of cell processing and user processing is shown in Figure 3.3.

![Figure 3.2: Functional split model](image)

The functional split can occur before CP1, after UPn, or between any two functions. Specifically when the split happens at CP split 1 (CPS1), then all the remaining functions are accordingly placed at the CC, resulting in a fully centralized baseband functional processing. When the split occurs at UPSn+1, then all the functions are processed at the EC resulting in DRAN baseline. Figure 3.4, gives further detailed explanation of the functional split architecture in H-CRAN. First, as mentioned earlier, the baseband processing chain of CPs and UPs could be split at most once, i.e. either at cell processing or at user processing. Once the split happens, the lower part of the split will be processed at the EC and the upper part of the split will be processed at the CC. Each flexible functional split entails a certain bandwidth requirement, which is calculated based on formulas stated in [23, 24]. Once a specific functional split is chosen, then number of CP and UP hosted by the DUs at the edge cloud and central cloud can be decided. Each DU at EC and CC has specified maximum capabilities of hosting the CP and UP. For our simulation model, the maximum capabilities of of hosting CP and UP are mentioned in simulation parameters.
For example, let’s consider the cases depicted in Figure 3.3.

- For cell $\alpha$, the functional split occurs at CP. The functions below the split point are placed at EC and the functions above are pushed to the central cloud. Since more functions are pushed above, more midhaul bandwidth is required to transmit the partially processed signals. All the CPs and UPs hosted in the CC must be processed by the same DUs. Similarly, the CPs placed at the EC must be processed by the same DU.

- For cell $\beta$, the functional split occurs at UP. The functions after the split point are processed at the EC and the functions before the split point are processed at CC. As most of the functions are EC, less midhaul bandwidth will be incurred. As stated in the previous point, all the UPs and CPs at the edge cloud must be processed by the same DU, and same applies in CC.

Introducing the caching in this architecture alters the decision of the functional split for each UE. The placement of the functional split is also dependent over the placement of the requested content from the UE. The relationship between the functional split and Cache placement is explained elaborately later as constraint.
The users request is initiated by requesting a content. The requested content could either be placed at edge cloud or central cloud. Each edge cloud has certain specified maximum capacity for storing the contents. The content storage capacity of central cloud is big enough to hold all the contents. Based upon the user’s delay requirements, the file from the pool of central cloud cache is placed at edge cloud. When a user is requesting the content, if the content is found at edge cloud, then all the functions are centralized at the edge cloud and the user is served with the content with less delay. If the the file is not found at the edge cloud, then it could be fetched from the central cloud allowing for flexible functional split options. The dependency over the content placement and functional split is discussed later in this thesis.
3.4 Content Placement

The aim of this section is to highlight the dependency of the content placement over flexible functional split. In this study we assume that all files are initially stored at the CC. After solving the optimization problem, the best place to store the content to minimize the power consumption is found. Depending on the users’ request and the network status, e.g., x-haul load, the decision on the optimal place of the content can be made. The place of content has impact on the functional split. In fact, if the content is already at the EC, then all the processing will be done at the EC, since for the processing we need the content. Otherwise, if the content is at the CC, then we can benefit from distributing the functional processing. In this study, we assume that the network knows the content requests beforehand, and thus it can solve the optimization problem based on the network status and users’ request. This assumption illustrate the potential power saving considering the optimal cache placement and functional split. To depict the more realistic scenario, the actual data of user request could be collected and analyzed in order to place the most popular contents at the EC cache. This will be done in real time scenario, which could be the potential future work with regards to the content placement.
Chapter 4

Problem Statement

Previous studies have been done on the flexible baseband processing function placement in CRAN. Similarly, various research work have been carried out in optimal placement of content placement. Both problems have been solved individually but never solved together. In this thesis work, both problems have been considered together. In order to bridge the gap, a constraint has been modeled as stated later under constraint. Considering both, it is important to analyze the optimal placement of content and functional processing and study their mutual dependencies.

4.1 Objective

Our objective is to minimize the system’s power consumption. The objective function is expressed as

\[
\min(P_t)
\]  

(4.1)

Where \(P_t\) denotes the total power consumption in the system. The total power consumption is expressed as,

\[
P_t = (g \times P_{LC}) + \left[ \left( P_{CC}^{cool} + I P_{CC}^{DU} \right) I(l > 0) + P_{CC}^{cache} \right] + \sum_{r \in R} \left[ \left( \sum_{c \in C_r} (P_{Tx} + P_{FH}) I(|I_c| > 0) \right) + \left( I \left( \sum_{c \in C_r} |I_c| > 0 \right) P_{ONU} + P_{EC}^{cool} I(l_r > 0) + I_r P_{EC}^{DU} + I \left( \sum_{c \in C_r} \sum_{i \in I_c} \delta_i > 0 \right) P_{EC}^{cache} \right) \right]
\]  

(4.2)

where,

- \(P_{CC}^{DU}, P_{EC}^{DU}\) : Power consumption of DU at CC and EC.
- \(P_{LC}\) : Power consumption of LC.
- \(P_{ONU}\) : Power consumption of ONU.
CHAPTER 4. PROBLEM STATEMENT

- $P_{Tx}, P_{FH}$: Power consumption of radio transmission and fronthaul.
- $P_{cool}^{CC}, P_{cool}^{EC}$: Power consumption of cooling at CC and EC.
- $P_{cache}^{CC}, P_{cache}^{EC}$: Power consumption of caching at CC and EC.

4.2 Input Parameters

System Parameters

- **Topology**: One CC is connected to multiple EC. Each EC is connected to an exclusive set of cells, and each cell exclusively covers a set of UEs.
- $I_x$: Set of UEs. When $x = 0$, it refers to all UEs in H-CRAN, otherwise, it refers to set of UEs in cell $x = c$.
- $C_x$: Set of cells. When $x = 0$, it refers to all cells in the entire H-CRAN, $x = r$ refers to the set of cells belonging to EC $r$.
- $D_x$: Set of DUs. When $x = 0$, it refers to all DUs in H-CRAN, $x = -1$ refers to set of DUs in the CC, $x = r$ refers to the set of DUs in EC $r$.
- $\mathbb{R}$: Set of ECs.
- $\mathbb{W}$: Set of wavelengths.
- $F$: Set of files to be cached.
- $F_x$: Set of functional split options, where $x = \{UP, CP\}$.
- $d_i$: Delay threshold of a UE $i$.
- $H^y_x(.)$: Pre-calculated mapping from a split option $x = \{UP, CP\}$ to the number of (UP and CP) functions at site $y = \{CC, EC\}$.
- $J_i(.)$: Pre-calculated mapping from UP split of UE $i$ to the required midhaul bandwidth, which is proportional to the number of resource blocks (RB) allocated to UE $i$.
- $G_c(.)$: Pre-calculated mapping from CP split of cell $c$ to the required midhaul bandwidth, which is proportional to the number of antennas and carrier bandwidth.
- $K$: Bandwidth capacity of a wavelength. Note that this is different from bandwidth induced and consumed by users and cells processing split, described with $J_i(.)$ and $G_c(.)$. 
• $L_y$: The capacity of a DU located at the “$y$” site, $y = \{CC, EC\}$, in terms of the number of $x$ functions that can be accommodated by this DU ($x$ represents CP or UP, and $y$ represents CC or EC). **Note:** $L_{CC}^{EC} < L_{CP}^{CC}$

• $S_f$: Size of the content.

• $[f_i, d_i]$: User demand pair which shows user $i$’s request of content $f$ within delay threshold $d_i$.

• $C_x$: Maximum storage capacity of the cache at $x = \{EC, CC\}$.

**Power calculation parameters**

• $P_{DU}^{CC}, P_{DU}^{EC}$: Power consumption of DU at CC and EC

• $P_{LC}$: Power consumption of LC.

• $P_{ONU}$: Power consumption of ONU

• $P_{Tx}, P_{FH}$: Power consumption of radio transmission and fronthaul

• $P_{cool}^{CC}, P_{cool}^{EC}$: Power consumption of cooling at CC and EC.

• $P_{cache}^{CC}, P_{cache}^{EC}$: Power consumption of caching at CC and EC.

**Delay parameters**

• $D_{prc}(p_i, q_c)$: Delay induced by functions processing given a specific split decision and is calculated as,

$$D_{prc}(p_i, q_c) = \sum_{i \in [p_i, F_{UP}]} d_{i,prc}^{CC} + \sum_{i \in [0, p_i]} d_{i,prc}^{EC} + \sum_{i \in [q_c, F_{UP}]} d_{i,prc}^{CC} + \sum_{i \in [0, q_c]} d_{i,prc}^{EC} \quad (4.3)$$

where the first two terms denote the UP processing delay at CC and EC and the last two terms show that of CP at CC and EC. Each delay component is function of the equipment processing speed and required processing [13].

• $D_{rsf}$: Delay induced by the number of radio sub-frames and is calculated as

$$D_{rsf} = N_{rsf} T_{rsf} \quad (4.4)$$

where $T_{rsf}$ is the time required to transmit one radio subframe. $N_{rsf}$ is the number of radio subframes and is given by,

$$N_{rsf} = \left[ \frac{S_f}{u_{prb} u_{MI} N_s} \right] \quad (4.5)$$

where $N_s$ is the number of symbols per physical resource block (PRB), $u_{prb}$ is the number PRB, and $u_{MI}$ is the modulation index in bits/prb.
• $D_{N_{of}}$: Delay induced by the number of optical frames.

$$D_{N_{of}} = N_{of}(p_i, q_c)T_{of} \quad (4.6)$$

where, $T_{of}$ is the optical frame time and $N_{of}(p_i, q_c)$ denotes the required number of optical frames to transmit UP/CP data which is given by,

$$N_{of}(p_i, q_c) = \left\lceil \frac{V_{cc}(p_i, q_c)N_{rsf}}{S_{of}/|C|} \right\rceil \quad (4.7)$$

where $V_{cc}(p_i, q_c)$ is data resulted from CP/UP [23]. $S_{of}$ is the optical frame size and the denominator of (4.7), shows the amount of bytes that can be used by cell $c$ in an optical frame in the midhaul. Note that we assume that the function processing delay is calculated individually for each radio subframe, then accumulated for all radio subframes.

• $D_{ONU}, D_{LC}$: Delay induced due to ONU and LC.

• $D_{opg}$: Delay incurred due to optical propagation.

• $D_{mWprg}, D_{mWcnv}$: Delay due to mm-Wave propagation and mm-Wave conversion.

• $D_{rpg}$: Delay due to radio propagation and is calculated by dividing the user’s distance to the RU by speed of light.

• $D_{sw}$: Delay due to switches.

• $D_{cache}^{CC}, D_{cache}^{EC}$: Delay incurred due to cache processing at CC and EC.

The calculation of each delay parameter is explained in [13].

### 4.3 Optimization Variables

• $p_i \in [0, F_{UP}]$: Integer variable denoting the UP functions split of UE $i$. Larger number of UP are at EC for higher value of $p_i$, hence, if $p_i = F_{UP}$ then all UP functions are distributed, otherwise, if $p_i = 0$ then all UP functions are centralized.

• $q_c \in [0, F_{CP}]$: Integer variable denoting the CP functions split of cell $c$. Larger number of CP are at EC for higher value of $q_c$. Hence, if $q_c = F_{CP}$ then all CP functions are distributed, otherwise, if $q_c = 0$ then all CP functions are centralized.

• $m_i \in D_r$: Integer variable indexing the DU hosting UPs of UE $i$ at EC $r$. Note that since the association between $i$ and $r$ is fixed, UE $i$ can choose a DU from a given set.

• $n_i \in D_{-1}$: Integer variable indexing the DU hosting UPs of UE $i$ at CC.
• $x_c \in D_r$: Integer variable indexing the DU hosting CPs of cell $c$ at EC $r$.
• $y_c \in D_{r-1}$: Integer variable indexing the DU hosting CPs of cell $c$ at CC.
• $w_r$: Integer variable indexing the wavelength used by EC $r$.
• $l_r$: Integer variable denoting number of active DUs at EC $r$.

$$l_r = \sum_{d \in D_r} (I(x_1 = d) \lor \ldots \lor I(x_{|C|} = d)),$$  \hspace{1cm} (4.8)

where $\lor$ shows OR function.
• $l$: Integer variable denoting number of active DUs at CC.

$$l = \sum_{d \in D_{r-1}} (I(y_1 = d) \lor \ldots \lor I(y_{|C|_0} = d)),$$ \hspace{1cm} (4.9)

where $\lor$ shows OR function.
• $g$: Integer variable denoting number of active wavelengths in the midhaul.

$$g = \sum_{w \in W} (I(w_1 = w) \lor \ldots \lor I(w_{|R|} = w)),$$ \hspace{1cm} (4.10)

where $\lor$ shows OR function.
• $b_{f,r,i}$: Binary variable denoting if file $f$ is placed at EC $r$ for user $i$.
• $\delta_i$: Binary variable denoting if user’s requested file is at EC and is calculated as

$$\delta_i = \sum_{r \in R} \sum_{f \in F} b_{f,r,i}.$$ \hspace{1cm} (4.11)

### 4.4 Constraints

1. The functional split can occur only once, either at CP or UP.

$$I(p_i < F_U, p) + I(q_c < F_C, p) = 1, \hspace{1cm} \forall i \in \mathbb{I}_c, \forall c \in \mathbb{C}_a.$$ \hspace{1cm} (4.12)

2. If UP of UE $i$ is split, then the UPs below the split point (that are to be placed at EC) must be placed in the same DU along with their CP at the EC.

$$I(p_i < F_U, p) \Rightarrow (m_i = x_c), \hspace{1cm} \forall i \in \mathbb{I}_c, \forall c \in \mathbb{C}_a.$$ \hspace{1cm} (4.13)
3. If CP of cell $c$ is split, then the CPs above the split point (that are to be placed at CC) must be processed by the same DU with all UPs (of all UEs in cell $c$) at CC.

$$I(q_c < F_{CP}) \Rightarrow (n_i = y_c), \quad \forall i \in I_c, \forall c \in C_o. \quad (4.14)$$

4. The total number of CPs that are accommodated by a DU $d$ at EC $r$ cannot exceed this EC-DUs CP capacity.

$$\sum_{c \in C_r} H_{CP}^{EC}(q_c).I(x_c = d) \leq L_{CP}^{EC}, \quad \forall r \in \mathbb{R}, \forall d \in \mathbb{D}_r. \quad (4.15)$$

5. The number of CPs that are accommodated by a DU $d$ in CS cannot exceed this CC-DUs CP capacity.

$$\sum_{c \in C_o} H_{CP}^{CC}(q_c).I(y_c = d) \leq L_{CP}^{CC}, \quad \forall d \in \mathbb{D}_{-1} \quad (4.16)$$

6. The number of UPs that are accommodated by a DU $d$ at EC $r$ cannot exceed this EC-DUs UP capacity.

$$\sum_{c \in C_r} \sum_{i \in I_c} H_{UP}^{EC}(p_i).I(m_i = d) \leq L_{UP}^{EC}, \quad \forall r \in \mathbb{R}, \forall d \in \mathbb{D}_r. \quad (4.17)$$

7. The number of UPs that are accommodated by a DU $d$ at CC cannot exceed this CC-DUs UP capacity.

$$\sum_{i \in I_o} H_{UP}^{CC}(p_i).I(n_i = d) \leq L_{UP}^{CC}, \quad \forall d \in \mathbb{D}_{-1}. \quad (4.18)$$

8. The total occupied midhaul bandwidth in a wavelength cannot exceed the wavelengths capacity, i.e., $K$.

$$\sum_{r \in \mathbb{R}} I(w_r = w). \sum_{c \in C_r} \left(G_c(q_c) + \sum_{i \in I_c} J_i(p_i)\right) \leq K, \quad \forall w \in \mathbb{W}. \quad (4.19)$$

9. If Function processing is at CC then Cache cannot be placed at local EC. If Cache is optimally placed at local EC, then Function processing must be at local EC.

$$p_i - F_{UP} \leq M(1 - b_{r,f_i}), \quad \forall i \in I_c, \forall r \in \mathbb{R}, \forall f \in F. \quad (4.20)$$

$$p_i - F_{UP} \geq -M(1 - b_{r,f_i}), \quad \forall i \in I_c, \forall r \in \mathbb{R}, \forall f \in F. \quad (4.21)$$

where $M$ denotes the big M method for optimization and $b_{r,f_i}$ denotes the placement of content $f$ at $r$ for every user $i$. 
10. The capacity of cache at EC $r$ cannot exceed the maximum capacity of the EC’s cache

$$X_r \leq C_{EC}, \quad \forall b \in B.$$ (4.22)

where,

$$X_r = \sum_{f \in F_r} I(b_{f,r}) S_f, \quad \forall r \in R.$$ (4.23)

11. The total delay of a user should be less than the delay threshold

$$D_{Total_i} \leq d_i$$ (4.24)

where $D_{Total_i}$ for a user is given by,

$$D_{Total_i} = D_{prc}(p_i, q_c) + D_{rsf} + D_{N_{of}} + D_{ONU} + D_{LC} + D_{opg} +$$

$$D_{mWprg} + D_{mWcnv} + Drpg + D_{sw} + \delta_i D_{EC}^{\text{cache}} + (1 - \delta_i) D_{CC}^{\text{cache}}$$ (4.25)

As mentioned earlier, the joint problem of flexible functional split and content placement has been considered in this thesis. Each user in the simulation has specific demands. The user demand is defined based on the delay requirements of the application type. The experienced delay of each user is modeled as function of communication function placement and requested content placement.

$$D = f(\text{function placement, content placement})$$ (4.26)

where $D$ signifies the user’s experienced delay. Multiple constraints are to be considered which bridges the two defined problems.

### 4.5 Methodology

The decision variables of the optimization problem are either integer variable or binary variable. The indicator functions are nonlinear in the decision variables for instance, $I$ in the constraint in Eq. (4.3). This nonlinear function together with the integer variables makes our mathematical model a mixed integer non linear problem (MINLP). MINLP is an optimization problem containing variables that are integers and binary and also it has non-linear functions in either the objective function and/or the constraints. The MINLP has the challenges of dealing non-linearities along with combinatorial explosion of integer variables.

The MINLP problems are non-deterministic polynomial-time hard (NP-hard), meaning that we cannot find the optimal solutions in a polynomial time. Therefore at cost of loosing the optimality
one can use heuristic algorithms to find a solution for these types of problems. However, there exist approaches such as branch and bound algorithm and exhaustive search algorithms that can solve the problem for very small problem dimensions. These techniques are not scalable with the size of network dimension e.g., size of optimization variables. As stated before, since our problem in this thesis work is MINLP, we have following alternatives to deal with it:

1. One approach is to linearize the nonlinear functions and then solve the problem with mixed integer linear programming (MILP) solutions.

2. The second approach is to use constraint programming, which suits well for non-linear constraints.

3. The third approach is to use heuristic algorithms. But since, our main aim is to look for optimal solution, we do not choose this approach. This approach has been considered for the future scope of this thesis.

It is worth mentioning that the MILP solutions are subset of the constraint programming. The constraint programming solutions are based on computer science fundamentals which includes logical programming and graph theory, in contrast to mathematical programming, that are based on numerical linear algebra.

The constraint programme reduces set of all possible values of the decision variables that satisfies all the defined constraints. In this thesis work, IBM Ilog Cplex constraint programming tool is used to solve the optimization problem. The IBM Ilog Cplex CP solver solves various discrete scheduling and combinatorial optimization problems. It is very efficient in solving the packing problems and also it enables the incorporation of constraints. One main advantage of constraint programming is the ability to derive the benefits of the structure of the problem to construct an adaptive search strategy to solve it. Constraint programming optimizing engine flexibly uses many unique concepts to find the optimal solutions in the efficient manner [20]. For example, the constraint programming solver uses the adaptive search method, which in turn consist of numerous heuristic search algorithms. The problem defined in this thesis contains bin packing and combinatorial optimization. The problem contains several constraints and more importantly it falls under the domain of the constraint programming tool. due to these facts we have chosen constraint programming as tool to solve our optimization problem.
Chapter 5

Simulation

5.1 System Model

For simulation, one CC is considered connected to multiple EC via a midhaul link. Optical fiber links are considered for the midhaul. Each EC serves a group of RUs and each RU serves a set of UEs. A total of 95 users are considered for 100 percent load. Only 95 users are considered for the considered model because increasing the number of users, fails to satisfy the delay constraint as more users are to be served. We assume that users are distributed uniformly in the cell coverage area of 250m and one resource block group is assigned to each user. In our system, each user requests randomly with a uniform distribution, a content of size $S_f$ from a set $F$, within a delay threshold $d_i$. All the simulation parameters are summarized in Table 5.1. The link between the EC and the RUs is assume to be mm-Wave link. The CCs and EC are equipped with DUs, where each DU has a specified maximum capacity of CP and UP processing. The DUs at CC is considered to have more computational capacity compared to DUs at EC. Presence of DUs at both EC and CC allows the possibility of having flexible placement of functional processing and content placement. Both EC and CC are provided with cache storage capacity. The cache storing capacity of the CC is considered to be large enough to store all the contents. The contents requested by each user could either be cached at the EC or CC. Each EC cache capacity has a specified maximum capacity in storing the contents. This scenario is simulated and the system’s behavior is analyzed in terms of a) total network power consumption, b) hitrate, where the hitrate is defined as the ratio of served users with satisfied demands, e.g., delay, to the total number of users, c) user’s average experienced delay, and d) midhaul bandwidth consumption.
5.1.1 Assumptions

In order to reduce the complexity of the work, certain assumptions are been taken into account for the simulation purpose.

- Each user is requesting a single content.
- One-way communication is considered for simulation, i.e., downloading content is considered.
- All the users are requesting for the content at the same time
- Same content could be placed in several caches.
5.1.2 Delay Model

The delay model used in this work is depicted in Figure 5.1. The total delay experienced by a user is an aggregated delay incurred at the CC, the EC, and the RU. The delay at the CC is the sum of the delay due to CP/UP at the CC (CC-Prc), number of optical frames (#OF), optical conversion (O-Cnvrt) and propagation (O-Prg), switches (SW) and caching ($D_{cache}^{CC}$). On the other hand, the delay incurred at the EC is due to the processing at the EC (EC-Prc), number of radio sub-frames (#RF), mm-Wave conversion (mW-Cnvrt) and propagation (mW-Prg), radio propagation (RF-Prg) and caching ($D_{cache}^{EC}$). As stated earlier in the constraints, this total delay experienced by a user must be less than its imposed delay threshold. The delay threshold average is same for all the users considered in the simulation.

Figure 5.1: Delay model of the proposed architecture
5.1.3 Power Model

The power model implemented in this work is depicted in Figure 5.2. The total power consumed in the network could be realized as the power consumed in the CC and EC. The power consumed in CC is the contribution of power consumed by active DUs, cooling system at CC, OLT, and caching at CC. Similarly, the power consumed at each EC is the sum of the power consumed by active DUs at the EC, cooling system at the EC, ONU, mm-Wave and radio transmission power. As mentioned earlier, the computational capacity of the DUs at the CC is higher compared to the DUs at EC. Thus by assumption, the power consumed by the DU at CC is less compared to power consumed by the DU at EC. As described in the problem statement, the main objective of this work is to minimize the overall power consumption of the network.

Figure 5.2: Power model of the proposed architecture
CHAPTER 5. SIMULATION

5.2 Simulation Parameters

The parameters and its corresponding values used in the development of the simulator are mentioned in Table 5.1.

Table 5.1: Simulation Parameters

<table>
<thead>
<tr>
<th>PARAMETERS</th>
<th>VALUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology</td>
<td>1 CC, 4 ECs, 5 RU per EC, each RU serve upto 5 users</td>
</tr>
<tr>
<td>Configuration of RU</td>
<td>20 MHz, 2*2 MIMO, 64 QAM</td>
</tr>
<tr>
<td>Capacity of DU at EC</td>
<td>3 CPs / 15 UPs</td>
</tr>
<tr>
<td>Capacity of DU at CC</td>
<td>37 CPs / 135 UPs</td>
</tr>
<tr>
<td>Radius of the Cell</td>
<td>250m</td>
</tr>
<tr>
<td>Size of the requested file</td>
<td>20 MB</td>
</tr>
<tr>
<td>Capacity limit of midhaul link</td>
<td>26000 Mbps</td>
</tr>
<tr>
<td>Number of CP/UP ($F_{UP}, F_{CP}$)</td>
<td>$F_{UP} = 3, F_{CP} = 3$</td>
</tr>
<tr>
<td>Power of DU at EC/CC</td>
<td>50 W / 100 W</td>
</tr>
<tr>
<td>LC power + ONU power</td>
<td>20 W, 5 W</td>
</tr>
<tr>
<td>Radio access +</td>
<td></td>
</tr>
<tr>
<td>Fronthaul link power consumption</td>
<td>20 W + 40 W</td>
</tr>
<tr>
<td>Power of cache activation at EC/CC</td>
<td>30 W / 20 W</td>
</tr>
<tr>
<td>Delay of Optical transmission</td>
<td>0.4 * 1e-3 sec</td>
</tr>
<tr>
<td>Ethernet switching delay</td>
<td>5.2 * 1e-6 sec</td>
</tr>
<tr>
<td>m-Wave conversion delay</td>
<td>30 * 1e-6 sec</td>
</tr>
<tr>
<td>Optical switching delay</td>
<td>2.5 * 1e-3 sec</td>
</tr>
</tbody>
</table>

5.3 Simulation Results

Figure 5.3, shows the total power consumption of the network with varying percentage of active users. Placing all the contents at EC restricts the centralization of the functions at EC according to constraint (9), thus it consumes more power due to low computational capacity of the DUs at EC. After a point, we can see a saturation in power consumption. This saturation is because of the capacity of EC which is fully utilized and has no room to accommodate users anymore. Less power is consumed when the requested content are placed at CC. The low consumption of power is mainly due to the high computational capacity of the DUs at CC and low hit rate.
CHAPTER 5. SIMULATION

Figure 5.3: Plot between total network power consumption and percentage of active users

The feasible and optimal solution curve is plotted for three different average delay threshold i.e. 45 msec, 60 msec and 70 msec. When the average delay threshold is stringent, the feasible solution curve tends towards the the curve where all files placed at EC. When the delay requirement is flexible, the feasible solution curve tend toward the curve where all files placed at CC. The reason is that delay tolerant contents are stored at CC, benefiting from the efficient power consumption and delay sensitive contents are stored at edge benefiting from low access delay feature of the architecture. In terms of requesting the contents, there is high possibility that multiple users request for the same files thus could reduce the processing which in turn reduces the overall power consumption. To illustrate this behaviour, the user request was modelled such that 80 percent of the users belonging to same EC request for the same 20 percent of the total files placed at the CC. this case was analyzed for different delay threshold and plotted along with the earlier curves (dotted lines) to analyze the potential saving of the total power consumption. For the average delay threshold of 45 msec and at maximum load, the total power consumption is 8.33% less than the initial uniform distribution request model, thus showing 8.33% potential saving of total power consumption. For the Average delay threshold of 60 msec and at maximum load, the total power consumption is 4.4% is less than the initial uniform distribution request model, thus showing 4.4% potential saving of total...
power consumption. For the Average delay threshold of 70 msec and at maximum load, the total power consumption is 1.5% is less than the initial uniform distribution request model, thus showing 1.5% potential saving of total power consumption. For the relaxed delay threshold of 70 msec, the difference in the total power consumption is less because, the users with relaxed delay requirement could access their content from the CC.

As overall, the total power consumption shows the increasing trend because, The total power consumed is depending on activation certain factors such as transponders (LC, ONU), DU’s, wavelengths in the midhaul. Thus when there are less number of users, less power is consumed as we don’t need to activate complete resources. As the number of users of increases, all the resources are utilised thus more is consumed. Thus, we observe increasing trend in the power consumption curve against number of active users.

![Figure 5.4: Hitrate against number of active users](image)

Figure 5.4 shows the evaluation of hitrate against number of active users. It shows that our results attains full hitrate as the solution is optimal and feasible i.e: the placement of functional
processing and the placement of the contents are optimal and all the users requesting for the content are fulfilled with their requirements successfully. The EC caching only case attains full hitrate until certain point but there is a breakpoint beyond which there is significant decrease in the hitrate. This breakpoint corresponds to the saturation point in previous Figure.5.3. The reason for the breakpoint in hitrate is due to the capacity constraint beyond which the EC cannot serve anymore users. On the other hand, the CC only case shows very low hitrate, as delay stringent requirements could not be satisfied by placing the content at the CC.

Figure 5.5: Average delay experienced by the users against number of active users

Figure 5.5, shows the average experienced delay of the users against percentage of active users in the network. As described in delay model earlier, the total experienced delay is the contribution of delay incurred by number of optical sub-frames and radio sub-frames required to transmit the user requested content via midhaul, fronthaul and radio link. Thus, as the number of active users increases, the total average delay increases as more resources are required in order to accommodate all the users. Comparing three different architectures, the EC only curve has less average delay, as
all the contents are placed at the EC and the users requesting for the content could fetch it with very less delay. When all the contents are placed at CC, more delay is incurred in fetching the content for the users. In this process, certain users with delay stringent requirements cannot be satisfied thus reduces the overall hitrate, as seen in previous Figure.5.4. The optimal and feasible solution curve optimal places the functional processing and contents such that every users in the network are fulfilled with their requirements. Delay stringent Users are served from EC and delay flexible users are served from the CC. When more users belonging to one EC request for same content, could fetch it with less delay once the requested content is already been placed at the EC. Thus when the user request was modelled such a way that 80% of the users requesting for the 20% of the total placed at the CC, the total average experienced delay is even less compared to our feasible solution. Comparing the Figure.5.3 and Figure.5.5, the CC only case consumes very less power but takes away more delay. On the other hand, the EC only case consumes more power and but takes less delay in serving users. Hence a trade-off been observed between the total power consumption and average experienced delay.

Figure 5.6: Midhaul bandwidth consumption against percentage of active users
Another main parameter to be analysed is the midhaul bandwidth consumption. In our architecture the midhaul is considered to be optical fibre. When all the contents are placed at the EC, then all the functional processing are done at the EC. By processing all the function at the EC, no midhaul bandwidth is consumed. When all the contents are placed at CC, there is flexibility of placing the functional processing and thus consumes the midhaul bandwidth. The feasible and optimal solution curve aims in placing the contents and functional processing based upon the user requirement and thus utilizes moderate midhaul bandwidth. Thus once again comparing the Figure 5.3 and Figure 5.6, a trade-off been observed between the total power consumption and total midhaul bandwidth consumption.

![Figure 5.7: Impact of delay threshold on total power consumption](image)

Figure 5.7, the impact of delay threshold of the users over the total power consumption of the network is shown for different loads. When the delay threshold is very minimum, all the contents are forced to be placed at the EC such that the users could fetch it with less delay. When all the contents are placed at the EC, all the processing must be done at the EC as per constraint (9).
Due to low computation capacity of the DUs at EC, more power is consumed. When the users have enough delay threshold, the functional processing could be done at CC and content can fetched from the CC. Due to high computational capacity of the DUs at CC, less power is consumed. Thus we see a decreasing trend in the total power consumption of the network when delay threshold increases. This analysis support the the fact that there is a huge trade-off between the total power consumption and the average experienced delay.

Figure 5.8: Impact of delay threshold on total required edge cloud cache capacity

Figure 5.8, shows the impact of varying the delay threshold over total required edge cloud cache capacity. When the delay threshold of the user is stringent, the requested content must be placed at the EC and also all the function must be processed at EC, so that the user’s requirement is satisfied. Thus we see high requirement of the edge cloud cache capacity for less delay threshold. When the delay threshold is increased, the users have the flexibility of fetching the content from the CC, thus there is significant decrease in the required edge cloud cache capacity for increasing delay threshold.
Chapter 6

Discussions

6.1 Conclusion

Simulator was modeled based on a CRAN architecture, and an analysis was done over flexible functional split and content caching. Additional constraint were proposed to the original CRAN architecture to simultaneously consider a flexible functional split and content placement among the central/edge clouds, taking their mutual design relations into account. The performance of the proposed modified architecture was analyzed in terms of the overall power consumption, experienced delay, hitrate and midhaul bandwidth consumption. The main motivation for the content placement was to improve user’s QoS in terms of experienced delay.

From the obtained results, it is observed that the proposed solution consumes less power compared to all files at EC. All files at CC consumes even more less power but the users with stringent delay are not met with their requirements in terms of the delay as it takes more delay fetching the content from the CC. But fetching the content from the EC takes less delay. Thus, a trade-off is been observed between the the network overall power consumption and user experienced delay. This conclusion becomes more evident in the analysis between the user’s delay threshold and total power consumption. Increasing the delay threshold among the users allow the users to fetch the content from CC thus a significant decrease in the power consumption. When it comes to the total required caching capacity at the EC, increasing the delay threshold among the users, reduces the required capacity at EC as the content could be placed at the CC. Only the contents of the users having stringent delay requirements are placed at the EC, thus making the system optimal. With respect to the midhaul bandwidth consumption, when all the functional processing are centralized at the CC, then it consumes more midhaul bandwidth as to send the requested content information and functional split information. The feasible and optimal solution consumes less midhaul bandwidth as only delay flexible users requirement are pushed to CC. Thus the same trade-off is observed between the overall power consumption and midhaul bandwidth.
As a result, a complete analysis is done on the systems behavior for various functional split and content placement options. The proposed solution provides feasible and optimal solution for the functional processing placement and content placement.

6.2 Future Work

1. We used IBM Ilog Cplex solver as a tool to solve the constraint programme. This solver obtains the optimal value if the parameter optimality tolerance gap is set to 0. However, the nature of the problem is NP-hard meaning, the complexity of solving the problem grows exponentially with the problem size. Thus, heuristic algorithm or artificial intelligence technique can be proposed to solve the problem in efficient way.

2. Considering the content popularity for each content would provide more realistic approach towards the content placement problem. For that purpose, real time data of the user’s request must be analyzed to provide the content popularity.

3. In practical the computation capacities of the DUs at EC and CC are different. Different ratios of the computational capacities of DUs at EC and CC can be considered and their impact over the overall power consumption can be analyzed.
Bibliography


[20] Use constraint programming to compute optimized schedules and solve other hard optimization problems.


