DOI:
10.1109/TMC.2019.2904061

Document Version
Peer reviewed version

A Cost-Driven Approach to Caching-as-a-Service in Cloud-Based 5G Mobile Networks

Seyed Ehsan Ghoreishi, Dmytro Karamshuk, *Member, IEEE,* Vasilis Friderikos, *Member, IEEE,* Nishanth Sastry, *Member, IEEE,* Mischa Dohler, *Fellow, IEEE,* and A. Hamid Aghvami, *Fellow, IEEE*

Abstract—The exploding volumes of mobile video traffic call for deploying content caches inside mobile operator networks. With in-network caching, users' requests for popular content can be served from a content cache deployed at mobile gateways in vicinity to the end user. This inherently reduces the load on the content servers and the backbone of operator's network. In light of the increasing trend in virtualization of network functions, we propose a cost-effective caching as a service (CaaS) framework for virtual video caching in 5G mobile networks. In order to evaluate the pros and cons of our CaaS approach, we formulate two virtual caching problems, namely maximum return on investment (MRI) and maximum offloaded traffic (MOT). MRI aims at maximizing return on caching investment by finding the best trade-off between the cost of cache storage and bandwidth savings from caching video contents in the mobile network operator (MNO)'s cloud. Likewise, MOT aims to maximize the traffic offloaded from the MNO's core and backhaul within given budget constraints. More specifically, taking the popularity and size of video contents into account, MRI and MOT aim to find the optimal caching tables which maximize the ratio of transmission bandwidth cost to storage cost and the offloaded traffic for a given budget, respectively. We reduce the complexity of the proposed problem formulated as a binary-integer programming (BIP) by using canonical duality theory (CDT). Experimental results obtained using the invasive weed optimization (IWO) have shown significant performance enhancement of the proposed system in terms of return on investment, quality, offloaded traffic and storage efficiency.

Index Terms—Caching-as-a-Service (CaaS), 5G virtual caching, mobile video delivery, canonical duality, invasive weed optimization.

1 Introduction

The extensive growth in adoption of smartphones and tablets has led to a continuous increase in mobile video traffic. According to the recent reports [1], mobile video will represent 78 percent of global mobile data traffic by 2021, a 9- fold increase from 2016. This new phenomenon has urged mobile operators to redesign their networks and search for cost-effective solutions to bring content closer to the end user [2], [3].

One approach to this problem lies in installing geographically distributed content delivery networks (CDNs), which can efficiently serve users within certain geographic areas. However, in order to reach an end user's device, CDN-served traffic must still traverse through the mobile operator's core network and radio access network (RAN). The significant strain on the operator's core network and RAN backhaul contributes to congestion, delays in streaming video content and a constraint on the network's capacity to serve a large number video requests currently. In contrast, with in-network caching, users can access popular content from caches of nearby MNO gateways, i.e. evolved packet core (EPC) and RAN [3], [4], [5], [6], therefore significantly reducing video streaming latency, congestion and increasing the capacity of the network to serve video content.

From the MNO's perspective, in-network caching also helps to reduce inter- and intra-MNO traffic and optimize operating costs for leasing expensive fiber lines that connect eNodeBs to EPC [5], [6]. The reduction in the outbound traffic from the content provider's users associated with the MNO decreases the traffic load directed to public CDN. This, in turn, inherently results in the content provider to pay less for CDN services.

Recently, the new trend of virtualizing mobile network functions into software-based cloud servers has been envisioned, which yields several advantages such as optimization of resource utilization, reduction in both capital and operating expenditures, and increase in scalability and flexibility [7], [8]. The emergence of network function virtualization (NFV) [9] has stimulated research on the concepts of RAN as a service (RANaaS) and EPC as a service (EPCaaS). RANaaS virtualizes the traditional radio access processing functions into the cloud [10], with remote antennas [remote radio heads (RRHs)] connected with the servers running the virtualized baseband units (BBUs) in the MNO's cloud center by high-speed fronthaul fiber networks. Likewise, with EPCaaS, some EPC network functions are instantiated on virtual machines on top of a virtualized platform, running in a MNOs' cloud centers [7].

The increasing drive towards the virtualization of mobile networks and services has motivated recent research aimed at proposing CaaS inside MNOs' cloud centers [8]. With CaaS, rather than running traditional CDN services virtually, which are still statical storage of files and management units in virtual machines of the cloud, CaaS instances in the mobile cloud centers can be adaptively created, immigrated, scaled (up or down), shared and released depending on the

The authors are with the Department of Informatics, Centre for Telecommunications Research, King's College London, London, Bush House, 30 Aldwych, WC2B 4BG , London, UK (e-mail: seyed_ehsan.ghoreishi@kcl.ac.uk).

user demands and requirements from third-party service providers and content providers. The MNO may charge content and service providers for caching their contents [11], whereas the operator guarantees a level of service for the cached contents.

In this paper, we propose a virtual caching policy in a cloud-based mobile operator network which maximizes the return on caching investment and offloaded traffic. To the best of our knowledge, this problem has not been investigated before in proactive off-line scenarios. Offloaded traffic signifies the traffic load that would be directed to public CDNs in the absence of caching in the operator's network. By proactive caching, we mean that the caching decisions are made before the appearance of any request for any content. By off-line caching we mean that the caching scheme knows the popularities of the contents (i.e. the number of requests made for each content) [11].

1.1 Related Work

Many studies have proposed CDNs for Internet content [12], as well as CDN services running in the cloud [13]. However, as explained earlier, caching at Internet CDNs do not address the problems of latency and capacity for video delivery in wireless networks.

Reference [14] investigates the effectiveness of caching least frequently used (LFU) as published by Hulu, caching using least recently used (LRU) policy, and a combination of the two using traces collected from a university campus. Like the Internet caching techniques, the above Internet video caching techniques do not address the problem of video capacity or delay in cellular networks.

Some studies have developed caching techniques for ad hoc networks [15], [16]. However, these techniques are not applicable to the problem of video caching and delivery in cellular networks.

Several approaches have been proposed to analyze intelligent caching strategies for mobile content caching inside MNO's network [5]. An extensive overview of the techniques for in-network content caching in 5G mobile networks has been introduced in [6], whereas multiple edge caching approaches at the base station level have been discussed in [2], [4], [17]. [18] further assumes cooperative caching between macro and small base stations. These works however, do not address the problem of caching in a cloud-based mobile network.

Reference [8] represents the first attempt to develop a virtualized caching system inside MNOs' cloud center. The differences between our work and the work of [8] are fourfold: 1) the work in [8] only minimizes inter- and intra-MNO traffic load and does not take cost-efficiency and caching costs into account; 2) reference [8] does not take the scalable video coding (SVC) video requirements into consideration; 3) the constraints on the capacity of the fronthaul is not taken into consideration in [8]; 4) the virtual caching problem proposed in [8] is solved using a simplistic algorithm, which runs relatively fast, however, rarely achieves an optimal allocation [19].

The work in [20] focused on optimizing caching in heterogeneous networks with the aim is to allow collaboration between the different networks so that to entail the optimal offloading between different networks under the assumption of different content requests. assume a fully distributed caching environment at the level of the base stations The works in [21] and [22] are closely related. [21] provides architectural views on caching in emerging cloudified 5G networks whilst outlining a number of techniques related to virtualization and caching. [22] assumes a fully distributed caching environment at the level of base stations. Furthermore, the work in [23] links Information-centric-networking (ICN) techniques with caching especially tailored in the case where user mobility might impact the performance and content can be retrieved by other ICN routers hosting the content closer to the point of attachment of the user.

1.2 Contributions and Outline

Our contributions with respect to this paper are stated as follows:

- To the best of our knowledge, we present the first attempt to formalize a virtual caching framework to maximize the return on caching investment. Our budget-constrained approach maximizes the offloaded traffic while meeting the maximum budget threshold.
- By introducing a quality priority factor in our optimization problem, we assign a higher weight to contents with larger bit-rate, hence prioritizing the high bit-rate contents over the low bit-rate ones. This, in turn, results in great improvements in the end user's quality of experience (QoE)[1], as the streaming application observes higher throughput, less latency and smaller start up and buffering times, which are the key QoE differentiators [24], [25].
- We focus our analysis on SVC-based dynamic adaptive streaming over HTTP (DASH) format, which encodes a single video into different quality layers. Therefore, it is more resource-efficient than traditional H.264/AVC-based DASH in which a separate AVC video file is encoded for each video quality format [26].
- We solve the virtual caching problem using canonical duality theory (CDT) [27]. More specifically, we transform our binary-integer programming (BIP) problem into a canonical dual problem in continuous space, which is a concave problem. Additionally, we provide the conditions under which the solutions of the canonical dual problem and primal problem are identical.
- The canonical dual problem results in complex nonlinear equations which are efficiently solved by applying invasive weed optimization (IWO) algorithm [28].
- Our results provide insight into the gains achieved from the perspective of the end user, content provider and MNOs.

In summary, our results suggest an improvement of more than 32% in return on investment, 21% in quality, 32% in offloaded traffic and 17% in storage efficiency in comparison to a naive LFU approach.

The rest of the paper is structured as follows. Section 2 describes the system model which represent a cloud based caching framework for mobile networks. The proposed virtual caching framework is formulated using mathematical

1. ITU-T FG IPTV, Liaison Statement 50, Definition of Quality of Experience, 2007

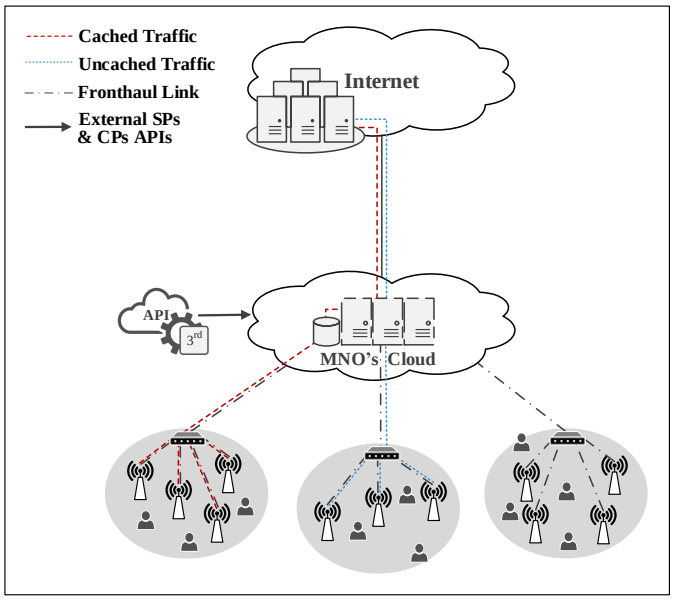

Fig. 1. Cloud-based virtual caching architecture.

programming techniques in 3. Section 4 presents the canonical dual framework to allow us solving the optimization problem. Section 5 conducts a simulation analysis of the model. The conclusion is presented in Section 7. Below is a summary of the abbreviations used throughout this paper.

2 System Model

We consider a virtual caching system inside the MNO's infrastructure as shown in Fig. 1. If a content is not available in the MNO's virtual cache, it needs to traverse the MNO's core and virtual BBU pool to get to the RRHs in a cluster, from which it is transmitted to the end users. Likewise, in order to cache a content in the operator's network, it needs to travel through the MNO core to be cached in the BBU pool, from which it is sent to the RRHs to be transmitted to the end users. The requests for the content are then served from the BBU pool in the MNO's infrastructure. Each 3rd-party service provider and content provider can program with the virtual caching by CaaSs application programming interfaces (APIs) . A service level agreement (SLA) is defined between the MNO and content providers, which determines the MNO's liabilities in providing the required resources to guarantee a level of service for the videos that have been cached. The MNO can dynamically charge for the resource utilization of the service and content providers.

The system consists of I video streams, which are indexed by the set $\mathcal{I} \triangleq \{1, \ldots, i, \ldots, I\}$. We index different quality layers of a video stream by the set $\mathcal{J} \triangleq \{1, \ldots, j, \ldots, J\}$. By q_{ij}, we denote the j^{th} quality layer of video object i, which has a size, source bit-rate and popularity (hit rate) of f_{ij}, b_{ij} and p_{ij}, respectively. We index different clusters by $\mathcal{N} \triangleq \{1, \ldots, n, \ldots, N\}$. One example of a cloud-based caching system architecture can be found in [8]. We use the following notations and variables:

Cache Assignment Binary Decision Variable (x_{nij}) represents an entry in the caching table $\mathbf{x}$. $x_{nij} = 1$ indicates that content q_{ij} is cached to serve users in cell n while meeting the SLA on users' experience of the content. If $x_{nij} = 0$ but content q_{ij} is available in the cache ($\sum_{n=1}^{N} x_{nij} \geq 1$) to serve users in a cell n' under SLA guarantees, requests for content q_{ij} from users in cell n can be directed to the cache without any SLA liabilities. If $\sum_{n=1}^{N} x_{nij} = 0$, requests for content q_{ij} are routed to the root.

Offloaded Traffic (l_{ij}) is the traffic load that would be directed to public CDNs in the absence of virtual caching in the MNO's network. In other words, it is the reduction in the transmission bandwidth as a result of caching q_{ij}, where $l_{ij} = f_{ij} \cdot p_{ij}$. We denote by $\mathtt{L}_n$ the cached traffic for each cluster n, which is given by $\mathtt{L}_n = \sum_{i=1}^{I} \sum_{j=1}^{J} l_{ij} \cdot x_{nij} \; \forall n \in \mathcal{N}$.

Storage Size ($\mathtt{S}_n$) is the storage capacity allocated to cluster n for cloud-based caching. For pricing purposes, we calculate the required storage under the assumption that cached files are not shared between different clusters. The total storage required for an individual cluster n is $\mathtt{S}_n = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij} \cdot x_{nij} \; \forall n \in \mathcal{N}$. We use the binary decision variable y_{ij} to find the total physical storage required $\mathbb{S} = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij} \cdot y_{ij}$, where $y_{ij} \; \forall i, j$ is given by

$$y_{ij} = \begin{cases} 1 & \text{if } \sum_{i=n}^{N} x_{nij} \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Fronthaul Capacity ($B_n^{\max}$) is the bandwidth capacity of the link between the operator's cloud center and cluster n. It should be noted that in order to meet the SLA with content providers, the MNO needs to provision for the peak rather than average bandwidth.

Quality Priority Factor ($\mathcal{Q}$) prioritizes the video contents with higher bit-rates over low bit-rate videos. *The offered throughput under TCP is inversely proportional to connections round trip time [29]. As shown in [4], in comparison with fetching data from public CDNs, caching contents inside the MNO's infrastructure results in a considerable decrease in round trip time. Therefore, in order to allocate higher bandwidth to video contents with high bit-rate requirements, we cache high bit-rate contents closer to the end users, which increases their TCP throughput and consequently reduces latency. The quality priority factor estimates the summation of the bit-rates of cached contents normalized over sum of bit-rates of all video contents as follows:

$$\mathcal{Q} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} \sum_{j=1}^{J} b_{ij} \cdot x_{nij}}{N \cdot \sum_{i=1}^{I} \sum_{j=1}^{j} b_{ij}}. \tag{2}$$

Return Function ($\mathcal{R}$) is the benefit gained from our virtualized caching system, which lays in the fact that caching video contents in the MNO's infrastructure would minimize the traffic load that would be directed to public CDNs. As customers of these CDNs, content providers are charged on the basis of the amount of traffic that is served from the CDN. We assume that the benefit of transmission bandwidth saving follows a predefined function $\Gamma : \mathbb{R} \rightarrow \mathbb{R}$. Thus, we estimate the benefit derived from the reduction in transmission bandwidth (hereinafter offloaded traffic) when

TABLE 1
Commonly Used Notation

Notation	Description
i	Video object index
j	Quality layer index of a video object
n	Cluster index
I	Total number of video objects
J	Total number of quality layers of a video object
N	Total number of clusters in the network
q_{ij}	The j^{the} quality layer of video content i
x_{nij}	A binary decision variable indicating whether video content q_{ij} is cached for cluster n
f_{ij}	Size of the j^{th} quality layer of video object i
p_{ij}	Popularity of the j^{th} quality layer of video object i
b_{ij}	Source bit-rate of the j^{th} quality layer of video i
l_{ij}	Offloaded traffic of cluster n
$\mathcal{Q}$	Quality priority factor
$\mathtt{L}_n$	Sum offloaded traffic of cluster n
$\mathtt{S}_n$	Size of cache storage of cluster n
$\mathcal{R}$	Offloaded traffic return function
$\mathcal{C}$	Cache storage cost function
$C^{\max}$	Total caching budget
$B_n^{\max}$	Link capacity of fiber line to RRH n

videos are cached for cluster n of the virtual caching system as

$$\mathcal{R}\left(\mathtt{L}_n\right) = \Gamma\left(\sum_{i=1}^{I}\sum_{j=1}^{J} l_{ij}\cdot x_{nij}\right) \qquad \forall n \in \mathcal{N}. \tag{3}$$

Cost Function ($\mathcal{C}$) is the cost incurred, which is represented by the amount of storage that is required for caching video contents. In general, public CDNs charge their customers based on the amount of bandwidth served by them. However, since the traffic load would traverse the MNO's infrastructure whether or not the contents are cached, the main factor incurring cost would be the cost of storage. We assume that the cache storage cost follows a predefined function $\Lambda : \mathbb{R} \to \mathbb{R}$. Hence, the cost associated with provisioned storage size $\mathtt{S}_n$ is

$$\mathcal{C}\left(\mathtt{S}_n\right) = \Lambda\left(\sum_{i=1}^{I}\sum_{j=1}^{J} f_{ij}\cdot x_{nij}\right) \qquad \forall n \in \mathcal{N}. \tag{4}$$

Both benefit and cost functions can be any appropriate function defined by the operator. A summary of commonly used notation is provided in TABLE 1.

A salient assumption on the optimization problems defined in the sequel is that the consideration of a batch content processing. To this end, batch content pre-processing and the decision making of when to process and optimize popular content is implementation depended and as such can be deemed as beyond the scope of the paper.

3 Problem Formulation

In this section, we formulate two virtual proactive caching problems based on the system model introduced in Section 2. The first optimization problem is formulated to achieve the optimal trade-off between the cost of caching video content (investment) and the benefit gained from content caching (return)[2]. The second optimization problem aims to maximize the offloaded traffic under the constraint of the total caching budget.

3.1 Return on Investment Maximized Caching

We formulate the caching problem aimed at maximizing the return on investment [hereinafter referred to as maximum return on investment (MRI)] as follows:

$$\max_{\mathbf{x}} \ \mathcal{Q}\cdot\frac{\sum_{n=1}^{N}\mathcal{R}\left(\mathtt{L}_n\right)}{\sum_{n=1}^{N}\mathcal{C}\left(\mathtt{S}_n\right)} \tag{5}$$

subject to:

$$\sum_{i=1}^{I}\sum_{j=1}^{J} b_{ij}\cdot p_{ij}\cdot x_{nij} \le B_n^{\max} \qquad \forall n\in\mathcal{N} \tag{5a}$$

$$x_{nij-1} \ge x_{nij} \qquad \forall n\in\mathcal{N}, \forall i\in\mathcal{I}, \forall j\in\mathcal{J}_{-\{1\}} \tag{5b}$$

$$x_{nij} \in \{0,1\} \qquad \forall n\in\mathcal{N}, \forall i\in\mathcal{I}, \forall j\in\mathcal{J}. \tag{5c}$$

The objective of optimization problem (5) is to find the optimal caching table $\mathbf{x}$ which determines what content should be cached for which cluster such that the ratio of overall return (3) to overall cost (4) is maximized. Constraint (5a) ensures that the sum of bit-rates of the video objects cached in the caching system for cluster n is upper-bounded by the maximum fronthaul capacity threshold, $B_n^{\max}$. This ensures adequate provision for the peak bandwidth. Constraint (5b) ensures that if a video quality layer is cached, all the lower quality layers are cached too (SVC requirement). We use binary variables $x_{nij} \in \{0,1\}$ explained in Section .

3.2 Budget-Constrained Caching

The budget-constrained caching problem, namely maximum offloaded traffic (MOT) is formulated as follows:

$$\max_{\mathbf{x}} \sum_{n=1}^{N}\sum_{i=1}^{I}\sum_{j=1}^{J} l_{ij}\cdot x_{nij} \tag{6}$$

subject to:

$$\sum_{n=1}^{N} \mathcal{C}(\mathtt{S}_n) \le C^{\max} \tag{6a}$$

$$\sum_{i=1}^{I}\sum_{j=1}^{J} b_{ij}\cdot p_{ij}\cdot x_{nij} \le B_n^{\max} \qquad \forall n\in\mathcal{N} \tag{6b}$$

$$x_{nij-1} \ge x_{nij} \qquad \forall n\in\mathcal{N}, \forall i\in\mathcal{I}, \forall j\in\mathcal{J}_{-\{1\}} \tag{6c}$$

$$x_{nij} \in \{0,1\} \qquad \forall n\in\mathcal{N}, \forall i\in\mathcal{I}, \forall j\in\mathcal{J}. \tag{6d}$$

The objective of optimization problem (6) is to find the optimal caching table $\mathbf{x}$ which maximizes the amount of cached traffic. (6a) represent the budget constraint. Constraints (6b)-(6d) are identical to the constraints in (5).

$$\mathbf{Y}^n = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{7}$$

2. It is worth noting that the price of memory is decreasing by around 40% each year, whereas fronthaul/backhaul capacity leasing does not follow the same trend.

We define a cache indicator vector $\mathbf{x} \triangleq [\mathbf{x}_n]_{N\times 1}$, where $\mathbf{x}_n = [x_{na}]_{A\times 1}$. Each entry $x_{na} \in \{0,1\}$ indicates whether the cache allocation pattern a is allocated for cluster n or not.

Note that all the clusters in the virtual caching system have the same cache allocation patterns matrix. We rewrite (5) as a BIP problem as follows:

$$\min_{\mathbf{x}} \left\{ \mathcal{P}(\mathbf{x}) = -\mathcal{Q} \cdot \frac{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{R}(\mathtt{L}_{na}) \cdot x_{na}}{\sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{C}(\mathtt{S}_{na}) \cdot x_{na}} \right\} \tag{8}$$

subject to:

$$\sum_{a=1}^{A} b_{na} \cdot x_{na} \leq B_n^{\max} \qquad \forall n \in \mathcal{N} \quad (8a)$$

$$x_{na} \cdot (x_{na} - 1) = 0 \qquad \forall n \in \mathcal{N}, \forall a \quad (8b)$$

$$\sum_{a=1}^{A} x_{na} = 1 \qquad \forall n \in \mathcal{N} \quad (8c)$$

where $\mathcal{Q} = \sum_{n=1}^{N}\sum_{a=1}^{A}(b_{na} \cdot x_{na}/B)$, b_{na} and c_{na} are the transmission bandwidth benefit and storage cost of allocating pattern a to cluster n. For cluster n, (8a) puts an upper-bound of $B_n^{\max}$ on the fronthaul bandwidth capacity, which is equivalent to (5a). Constraint (8b) is a pure binary constraint that ensures $x_{na} \in \{0,1\}$. (8c) ensures that at most one allocation pattern is chosen for each caching layer.

Although the optimization problem (8) is simpler and more tractable than (5), the solution is still exponentially complex.

4 Canonical Dual Framework

4.1 Dual Problem Formulation

We convert our BIP problem (8) into a continuous space canonical dual problem using CDT [27], [30], which is solved in continuous space. We then identify the conditions under which the solution of the canonical dual problem is identical to that of the primal. A generic framework for solving 0-1 quadratic problems using CDT can be found in [31]. However, due to additional constraints, our problem is more complex. A framework for solving resource allocation BIP problems using CDT is given in [32], which will be extended to solve (8).

We define the feasible space for the primal problem (8) by $\mathcal{Z}_p = \left\{\mathbf{x} \in \{0,1\}^{NA}\right\}$. We temporarily relax the equality constraints (8b) and (8c) to inequalities and transform the primal problem with these inequality constraints into continuous domain canonical dual problem. We then solve the problem in continuous space and provide the conditions under which the solutions of the canonical dual problem and primal problem are identical.

As a key step towards canonical dual formulation, we define the geometrical operator for the primal problem as $\wedge(\mathbf{y}) = (\boldsymbol{\delta}, \boldsymbol{\beta}, \boldsymbol{\tau}, \boldsymbol{\sigma}) \in \mathcal{Y}_g$, which is a vector valued mapping where $\mathcal{Y}_g$ is the feasible space for $\mathbf{y}$, and

$$\begin{cases} \boldsymbol{\lambda} = [\sum_{a=1}^{A} b_{na}x_{na} - B_n^{\max}]_{N\times 1} \\ \boldsymbol{\mu} = [x_{na} \cdot (x_{na} - 1)]_{NA\times 1} \\ \boldsymbol{\nu} = [\sum_{a=1}^{A} x_{na} - 1]_{N\times 1} \end{cases} \tag{9}$$

Therefore, the feasible space for $\mathbf{y}$ is defined by $\mathcal{Y}_g = \mathbb{R}^N \times \mathbb{R} \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\lambda} \leq 0, \boldsymbol{\mu} \leq 0, \boldsymbol{\nu} \leq 0$. $\boldsymbol{\mu} = \sum_{n=1}^{N}\sum_{a=1}^{A} \mathcal{C}(\mathtt{S}_{na}) \cdot x_{na} - C^{\max}$.

Next, we define the indicator function [31] as

$$V(\mathbf{y}) = \begin{cases} 0 & \text{if } \mathbf{y} \leq 0 \\ +\infty & \text{otherwise.} \end{cases} \tag{10}$$

We rewrite the primal problem (8) in the canonical form using indicator function (10) as follows:

$$\min \left\{ V(\wedge(\mathbf{y})) + \mathcal{P}(\mathbf{x}) \right\}. \tag{11}$$

We now define $\mathbf{y}^* = (\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ as the vector of dual variables associated with the corresponding restrictions $\mathbf{y} \leq 0$. The feasible space for $\mathbf{y}^*$ is defined by $\mathcal{Y}_d = \mathbb{R}^N \times \mathbb{R}^{NA} \times R^N | \boldsymbol{\lambda}^* \geq 0, \boldsymbol{\mu}^* \geq 0, \boldsymbol{\nu}^* \geq 0$. Based on the Fechnel transformation, the canonical sup-conjugate function of $V(\mathbf{y})$ is defined as

$$\begin{aligned} V^*(\mathbf{y}^*) &= \sup \left\{ \langle \mathbf{y}, \mathbf{y}^* \rangle - V(\mathbf{y}) | \mathbf{y} \in \mathcal{Y}_g, \mathbf{y}^* \in \mathcal{Y}_d \right\} \\ &= \sup_{\mathbf{y}^*} \left\{ \langle \boldsymbol{\lambda}^T\boldsymbol{\lambda}^* + \boldsymbol{\mu}^T\boldsymbol{\mu}^* + \boldsymbol{\nu}^T\boldsymbol{\nu}^* - V(\mathbf{y}) \rangle \right\} \\ &= \begin{cases} 0 & \text{if } \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^* \\ +\infty & \text{otherwise.} \end{cases} \end{aligned} \tag{12}$$

Using the definition of sub-differential, it can be easily verified that if $\mathbf{y}^* > 0$, then the condition $\mathbf{y}^T\mathbf{y}^* = 0$ leads to $\mathbf{y} = 0$, and consequently $\mathbf{x} \in \mathcal{X}_p$. Hence, the dual feasible space for the primal problem in (8) is an open positive cone defined by $\mathcal{X}_p^\sharp = \{\mathbf{y}^* \in \mathcal{Y}_d | \mathbf{y}^* > 0\}$.

We define the total complementarity function [27] as

$$\Xi(\mathbf{x}, \mathbf{y}^*) = \wedge(\mathbf{x})^T\mathbf{y}^* - V^*(\mathbf{y}^*) + \mathcal{P}(\mathbf{x}), \tag{13}$$

which is obtained by replacing $V(\mathbf{y}) = \wedge(\mathbf{x})^T\mathbf{y}^* - V^*(\mathbf{y}^*)$ (Fechnel-Young equality) in (11). We use the definitions of $\wedge(\mathbf{x}), V^*(\mathbf{y}^*)$ and $\mathcal{P}(\mathbf{x})$ to express $\Xi(\mathbf{x}, \mathbf{y}^*) = \Xi(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ as given by (14), shown in the next page.

Next, we define the canonical dual function [27], [31] using the canonical dual variables as

$$\Upsilon(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*) = \text{sta} \left\{ \Xi(\mathbf{x}, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*) \right\}, \tag{15}$$

where sta$(\cdot)$ denotes finding the stationary point of the function. We are primarily interested in the cache allocation vector $\mathbf{x}$ for a node n. The stationary point of $\Xi(\mathbf{x}, \mathbf{y}^*)$ occurs at

$$x_{na}(\mathbf{y}^*) = \frac{\vartheta + \zeta}{2\mu_{na}^*} \qquad \forall n, a, \tag{16}$$

where $\vartheta = [b_{na} \cdot \mathcal{R}(\mathtt{L}_{na})] / [B \cdot \mathcal{C}(\mathtt{S}_{na})]$ and $\zeta = \mu_{na}^* - \lambda_n^* b_{na} - \nu_n^*$. The stationary point is obtained through $\nabla_{\mathbf{x}}\Xi(\mathbf{x}, \mathbf{y}^*) = 0$. Using (15) and (16), we obtain the dual function, which is given by (17), shown at the next page.

The dual function is a concave function on $\mathcal{X}_p^\sharp$. The canonical dual problem associated with (8) can be formulated as

$$\min \left\{ \mathcal{P}(\mathbf{x}) | \mathcal{X}_p \right\} = \max \left\{ \Upsilon(\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*) | \mathcal{X}_p^\sharp \right\}. \tag{18}$$

Theorem 1. If $\mathcal{P}(\tilde{\mathbf{x}}) = \Upsilon(\tilde{\mathbf{y}}^*)$ where $\tilde{\mathbf{x}}$ denotes the KKT point of the primal problem and $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\lambda}}^*, \tilde{\boldsymbol{\mu}}^*, \tilde{\boldsymbol{\nu}}^*) \in \mathcal{X}_p^\sharp$ denotes the KKT point of the dual function, there exists a perfect

$$\Xi(\mathbf{x},\mathbf{y}^*)=\sum_{n=1}^{N}\sum_{a=1}^{A}\left[x_{na}\left(\lambda_n^* b_{na}+\nu_n^*-\mu_{na}^*\right)\right]-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A} b_{na}x_{na}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\mathcal{R}(\mathtt{L}_{na})x_{na}}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\mathcal{C}(\mathtt{S}_{na})x_{na}}$$
$$-\sum_{n=1}^{N}\lambda_n^* B_n^{\max}-\sum_{n=1}^{N}\nu_n^*+\sum_{n=1}^{N}\sum_{a=1}^{A}\mu_{na}^* x_{na}^2. \tag{14}$$

$$\Upsilon(\boldsymbol{\lambda}^*,\boldsymbol{\mu}^*,\boldsymbol{\nu}^*)=-\sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{\vartheta+\zeta}{2\mu_{na}^*}\left(3\vartheta+\zeta\right)\right]-\frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{\left[B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\right]^2}$$
$$-\sum_{n=1}^{N}\lambda_n^* B_n^{\max}-\sum_{n=1}^{N}\nu_n^*. \tag{17}$$

duality relationship between the primal problem in (8) and its canonical dual problem.

Proof. The proof directly extends from [30]. ■

Theorem 1 shows that the BIP in (8) is converted into a continuous space canonical dual problem which is perfectly dual to it. Moreover, the KKT point of the dual problem provides the KKT point of the primal problem.

Theorem 2. (global optimality conditions): If $\tilde{\mathbf{y}}^* = (\tilde{\boldsymbol{\lambda}}^*,\tilde{\boldsymbol{\mu}}^*,\tilde{\boldsymbol{\nu}}^*) \in \mathcal{X}_p^\sharp$, then $\tilde{\mathbf{x}}$ is a global minimizer of $\mathcal{P}(\mathbf{x})$ over $\mathcal{X}_p$ and $\tilde{\mathbf{y}}^*$ is a global maximizer of $\Upsilon(\tilde{\boldsymbol{\lambda}}^*,\tilde{\boldsymbol{\mu}}^*,\tilde{\boldsymbol{\nu}}^*)$ over $\mathcal{X}_p^\sharp$. Hence, $\mathcal{P}(\tilde{\mathbf{x}}) = \min\{\mathcal{P}(\mathbf{x})|\mathcal{X}_p\} = \max\left\{\Upsilon(\boldsymbol{\lambda}^*,\boldsymbol{\mu}^*,\boldsymbol{\nu}^*)|\mathcal{X}_p^\sharp\right\} = \Upsilon(\tilde{\boldsymbol{\lambda}}^*,\tilde{\boldsymbol{\mu}}^*,\tilde{\boldsymbol{\nu}}^*)$.

Proof. The proof directly extends from [30]. ■

According to Theorem 2, if the given global optimality conditions are met, the solution of the canonical dual problem provides an optimal solution to the primal problem. Solving the KKT conditions associated with (17) is necessary and sufficient for global optimality as the dual problem is a concave maximization problem over $\mathcal{X}_p^\sharp$.

The KKT conditions of the dual function in (17) are given by $(\partial\Upsilon/\partial\lambda_n^*)=0, (\partial\Upsilon/\partial\mu_{na}^*)=0$ and $(\partial\Upsilon/\partial\nu_n^*)=0$. (19), (20) and (21), shown in the next page, give the respective partial derivatives.

4.2 Invasive Weed Optimization Algorithm

Traditional gradient-based algorithms exist in literature for solving the non-linear equations resulting from the KKT conditions associated with the dual function. However, they show many defects such as oscillatory behavior, sensitivity to choice of initial values and complexity associated with the differentiation of KKT conditions and calculation of step size.

We deploy an IWO [28] algorithm for solving the complex non-linear equations associated with the KKT conditions [33]. Inspired by the invasive and robust nature of weeds, IWO is an evolutionary optimization algorithm, which has been shown to perform better than traditional approaches in terms of convergence. It also has the desirable properties of dealing with non-differentiable and complex objective functions and does not show the aforementioned defects. In summary, the key steps of IWO are as follows:

- *Initialization*, where seeds are randomly dispersed over the search space;
- *Reproduction*, where every seed grows to a flowering plant and produces seeds;
- *Spatial Dispersion*, where produced seeds are distributed based on a normal distribution with a mean of zero and standard deviation reducing from an initial value σ_{initial} to a final value σ_{finala} according to equation $\sigma_{\text{iter}} = [(iter_{\max}-iter)/iter_{\max}]^g(\sigma_{\text{initial}}-\sigma_{\text{final}})+\sigma_{\text{final}}$, where g is the modulation index;
- *Competitive Exclusion*, where a competitive mechanism is implemented for eliminating undesirable plants. A detailed discussion on IWO is out of scope of this paper. Interested reader is referred to [28], [34].

5 Simulation Results

We assume a cached-enabled cloud-based operator network consisting of four clusters. We evaluate the performance of our caching schemes in terms of return on investment, offloaded traffic, quality metric and cache size, which represent the gain achieved from the viewpoint of the content provider, MNO, end-user and MNO, respectively.

As in [35], [36], we assume that the video popularity is Zipf-like with a parameter of 0.65 and the video file sizes follow a Pareto (0.25) distribution with a minimum size of 60 megabytes. Without loss of generality, we suppose caching is performed at the level of entire video objects as in [37]. We can simply adjust index i to represent the i^{th} chunk rather than video object to enable caching at the chunk level.

We compare our proposed approach with the hit rate optimal caching algorithm LFU, which caches the most popular video contents [37], [38]. In contrast with the other widely used caching algorithm, LRU, LFU focuses on historical popularity over a long period of time. As a caching technique, our approach also considers a long term content popularity. Therefore, it is pertinent to compare our proposed scheme with LFU. Additionally, *the results in [37] confirm the relative loss in hit rate of LRU compared with LFU observed for homogeneous content.

$$\frac{\partial \Upsilon}{\partial \lambda_n^*} = \sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{b_{na}}{4\mu_{na}^*}(3\vartheta+\zeta)+3b_{na}\frac{\vartheta+\zeta}{4\mu_{na}^*}\right] - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{-b_{na}^2}{2\mu_{na}^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)+\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{R}(\mathtt{L}_{na})b_{na}}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{B\cdot b_{na}\cdot\mathcal{C}(\mathtt{S}_{na})}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{\left[B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\right]^2} - \sum_{n=1}^{N}B_n^{\max}, \tag{19}$$

$$\frac{\partial \Upsilon}{\partial \mu_{na}^*} = -\sum_{n=1}^{N}\sum_{a=1}^{A}\left[\left(\frac{\mu_{na}^*-\vartheta-\zeta}{4{\mu_{na}^*}^2}\right)(3\vartheta+\zeta)-\frac{3\vartheta+3\zeta}{4\mu_{na}^*}\right] - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\mu_{na}^*-\vartheta-\zeta}{2{\mu_{na}^*}^2}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(B\cdot\mathcal{C}(\mathtt{S}_{na})\frac{\mu_{na}^*-\vartheta-\zeta}{2{\mu_{na}^*}^2}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{\left[B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\right]^2} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\mu_{na}^*-\vartheta-\zeta}{2{\mu_{na}^*}^2}\right)\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}, \tag{20}$$

$$\frac{\partial \Upsilon}{\partial \nu_n^*} = \sum_{n=1}^{N}\sum_{a=1}^{A}\left[\frac{1}{4\mu_{na}^*}(3\vartheta+\zeta)+\frac{3\vartheta+3\zeta}{4\mu_{na}^*}\right] - \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\frac{-b_{na}}{2\mu_{na}^*}\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)+\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{-\mathcal{R}(\mathtt{L}_{na})}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)} + \frac{\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\frac{B\cdot\mathcal{C}(\mathtt{S}_{na})}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(b_{na}\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{R}(\mathtt{L}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)}{\left[B\cdot\sum_{n=1}^{N}\sum_{a=1}^{A}\left(\mathcal{C}(\mathtt{S}_{na})\frac{\vartheta+\zeta}{2\mu_{na}^*}\right)\right]^2} - N. \tag{21}$$

We consider three scenarios and measure the aforementioned metrics in each scenario. In *Scenario 1*, the total number of popular contents varies in the range [500,10000] (with 4 quality layers) whereas the sum fronthaul capacity is set to 25 Gbps. The default number of content is set as 4000 in scenarios 2 and 3. We vary the overall fronthaul capacity in the commercially available range of 15 to 40 Gbps (increments of 5 Gbps) in the former and the total cost from 0 to 1 in the latter. We relax the caching budget constraint in scenarios 1 and 2. A comparison of the performance of different caching techniques under the three scenarios is illustrated in TABLE 2

In each of the above mentioned simulation scenarios, we solve the KKT conditions for each dual variable associated with the dual problem using the IWO algorithm (implemented in MATLAB) and compute the allocation vector $\mathbf{x}_n$ using (16). A pseudo code for the resource allocation algorithm is given as Algorithm 1. TABLE 3 provides a summary of the simulation parameters for IWO.

Fig. 2 shows the convergence of IWO algorithm for one of the KKT conditions ($\partial\Upsilon/\partial\lambda_n^*$). The x-axis shows the number of iterations whereas the y-axis shows the value of fitness function, which is $\partial\Upsilon/\partial\lambda_n^*$. Over 500 iterations, the value of fitness function is 2.64×10^{-06}.

5.1 Scenario 1 - Variable Content Population

Fig. 3 demonstrates the performance of the caching algorithms under *Scenarios 1*. As shown in Fig. 3(a), for MRI, a growth in the size of the database initially decreases the return on investment. However, once the content population reaches a certain size ($\geq$ 6000), it enters a steady state and remains unchanged. Before reaching a steady state, both overall cache size and offloaded traffic have an increasing behavior as the number of contents rises [see Fig. 3(b) and Fig. 3(c)]. However, the growth in the cache size incurs higher cost than the benefit gained from the increase in the offloaded traffic load, which leads to a gradual decrease in the return on investment. At the point that the return on investment reaches a steady state, the same occurs to cache size and offloaded traffic. This can be justified by the direct relationship between return on investment and the ratio of return function (related to offloaded traffic) and cost function (related to cache size).

Algorithm 1: C-RAN caching based on IWO (adapted from [33])

initialize $\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*, \boldsymbol{\tau}^*, \forall n \in \mathcal{N}, iter = 0$;
$\forall\, \partial\Upsilon/\partial\boldsymbol{\nu}^*$, where $\boldsymbol{\nu}^* \in (\delta^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*, \boldsymbol{\tau}^*)$
create randomly dispersed initial population of Q individuals (weeds):
$\mathcal{W} = \{W_1, \cdots, W_Q\}$;
while $|\boldsymbol{\nu}^*| > \varrho$ *or* $iter = iter_{max}$ **do**
  evaluate the fitness of each individual i.e., calculate $f(W_n), \forall n \in \mathcal{W}$ and the colony's best (f_{best}) and worst (f_{worst}) fitness;
  sort $\mathcal{W}$ in ascending order according to $f(W_n)$;
  select the first Q_p individuals of $\mathcal{W}$ to create the set $\mathcal{W}_p$;
  Reproduction:
  $\forall\, W_j, j = 1, \cdots, Q_p$
  generate
  $S_j = \frac{f(W_j) - f_{\text{worst}}}{f_{\text{best}} - f_{\text{worst}}} \times (S_{\max} - S_{\min}) + S_{\max}$ seeds;
  create the population of the generated seeds, $\mathcal{W}_s = \{W_s\}$;
  Spatial Dispersion:
  for $i = 1 : |\mathcal{W}_s|$ **do**
    $W_s^i \leftarrow W_s^i + \phi^i$, where $\phi^i \sim L(0, \sigma_{\text{iter}})$;
  end
  Competitive Exclusion:
  create parents and seeds, $\mathcal{W}^* = \mathcal{W} \cup \mathcal{W}_s$;
  sort $\mathcal{W}^*$ in ascending order according to fitness;
  select the first $Q_{\max}$ individuals of $\mathcal{W}^*$ and create $\mathcal{W}$;
end
select the best fitted individuals $\boldsymbol{\lambda}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*$ and $\boldsymbol{\tau}^*$;
calculate $\mathbf{x}_n$ using (16);

TABLE 2
Performance Comparison of Caching Techniques

Metric[1]	Scenario	Figure	BPA[2]	MRI (%)	MOT (%)	LFU (%)
ROI	1	Fig. 3(a)	MRI	-	+37.1	+33.7
	2	Fig. 4(a)	MRI	-	+38.2	+32.13
	3	Fig. 5(a)	MRI	-	+16.44	+30.87
CS	1	Fig. 3(b)	MRI	-	−42.27	−21.82
	2	Fig. 4(b)	MRI	-	−34.41	−17.06
	3	Fig. 5(b)	MRI	-	−16.22	−13.51
OT	1	Fig. 3(c)	MOT	+28.45	-	+34.72
	2	Fig. 4(c)	MOT	+25.7	-	+34.5
	3	Fig. 5(c)	MOT	+16.2	-	+28.9
QM	1	Fig. 3(d)	MRI	-	+13.01	+21.82
	2	Fig. 4(d)	MRI	-	+11.74	+21.61
	3	Fig. 5(d)	MRI	-	+12.67	+20.87

[1]ROI: return on investment; CS: cache size; OT: offloaded traffic; QM: quality metric.
[2]BPA: best performing algorithm.

TABLE 3
IWO Numerical Parameter Values

Parameter	Value
Size of initial population (Q)	20
Minimum fitness threshold (ϱ)	10^{-6}
Maximum number of iterations ($iter_{\max}$)	500
Maximum number of plants ($Q_{\max}$)	15
Minimum number of seeds ($S_{\min}$)	0
Maximum number of seeds ($S_{\max}$)	7
Initial standard deviation (σ_{initial})	10
Final standard deviation (σ_{final})	0.01

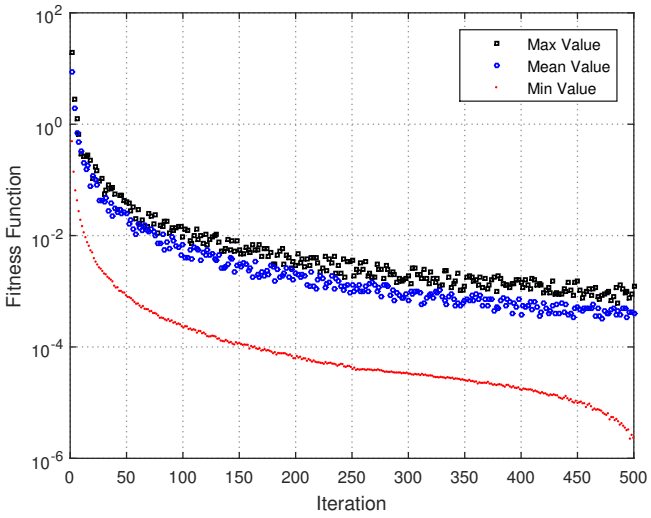

Fig. 2. Convergence of IWO for the KKT condition $\partial\Upsilon/\partial\lambda_n^* = 0$.

As can be seen from Fig. 3(d), there is a slight positive correlation between return on investment and quality metric. As the content population increases, MRI demonstrates a tendency to cache more video objects in order to prevent the quality metric to be reduced significantly. This in turn decreases the return on investment due to the noticeable rise in cache storage cost.

Likewise, Fig. 3(a) indicates that in case of MOT and LFU, return on investment decreases alongside the increase in the size of the content database. However, due to the cost unawareness nature of the aforementioned schemes, they demonstrate a considerably higher decrease in return on investment in comparison with MRI. As shown in Fig. 3(c) and Fig. 3(b), by considering both the size and popularity of video contents and not taking storage into account, MOT induces the highest increase in offloaded traffic, and consequently cache storage requirements.

With SVC, in oder to decode a higher video quality representation, all of the lower quality layers are needed. Therefore, low video quality layers which are smaller in size and bit-rate have greater popularity than high quality layers. Since LFU only takes popularity into consideration, it caches highly popular videos, which are normally smaller in size and bit-rate in comparison with higher quality representations. Hence, as shown in Fig. 3(b), it leads to lower storage requirements compared with MOT in addition to lower offloaded traffic [Fig. 3(c)] and quality [Fig. 3(d)].

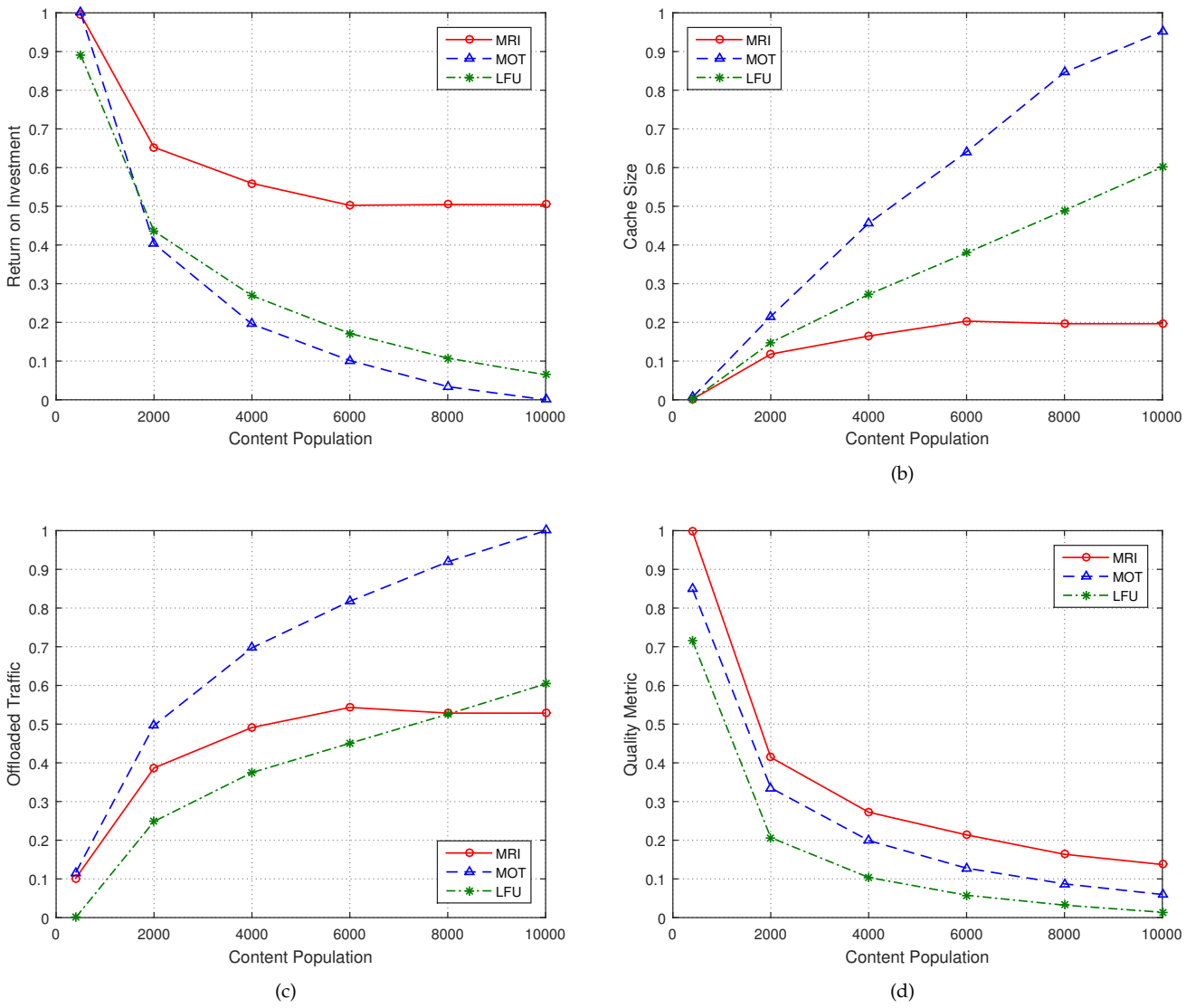

Fig. 3. Scenario 1 - varying number of contents: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

5.2 Scenario 2 - Variable Fronthaul Capacity

Fig. 4 evaluates the performance of the caching schemes under *Scenarios 2*. Similar to *Scenario 1*, MRI outperforms both MOT and LFU with regard to return on investment, storage efficiency and quality. Likewise, the best performance in terms of increasing the offloaded traffic is achieved by MOT. As the fronthaul capacity increases, MRI also takes higher bit-rate video objects into account, which increases the quality metric significantly as shown in Fig. 4(d). Therefore, with the increase of the fronthaul capacity, at the cost of a slight reduction in the return on investment [Fig. 4(a)] and storage efficiency [Fig. 4(b)], we achieve a considerable increase in the quality metric [Fig. 4(d)] and a satisfactory rise in offloaded traffic load [Fig. 4(c)].

For MRI and LFU, increasing the fronthaul capacity relaxes the fronthaul capacity constraint, and hence enables caching more video contents. However, giving priority to video objects that are large in size and popularity, MOT leads to a higher increase in offloaded traffic compared with both MRI, which takes cost into consideration by maximizing the return on investment and LFU, which only considers popularity.

5.3 Scenario 3 - Variable Cost

Fig. 4 analyzes the performance of the caching schemes under *Scenarios 3*. Unlike the first and second scenarios where no caching budget constraint is set, here we consider maximum budget as the varying factor. Since MRI does not have a budget constraint [see (5)], varying the cache budget causes no change to its performance, and hence it demonstrates a static behavior. Similar to *Scenarios 1* and *2*, in this scenario, MRI has a better performance in terms of return on investment, storage efficiency and quality. Likewise, MOT results in a higher increase in offloaded traffic load.

For a fixed number of content items (4000), as the budget constraint increases, MOT continues to cache more contents. As shown in Fig. 4(c), giving priority to video objects which are large in size and popularity, MOT leads to a higher increase in offloaded traffic in comparison with MRI, which takes cost into consideration by maximizing the return on

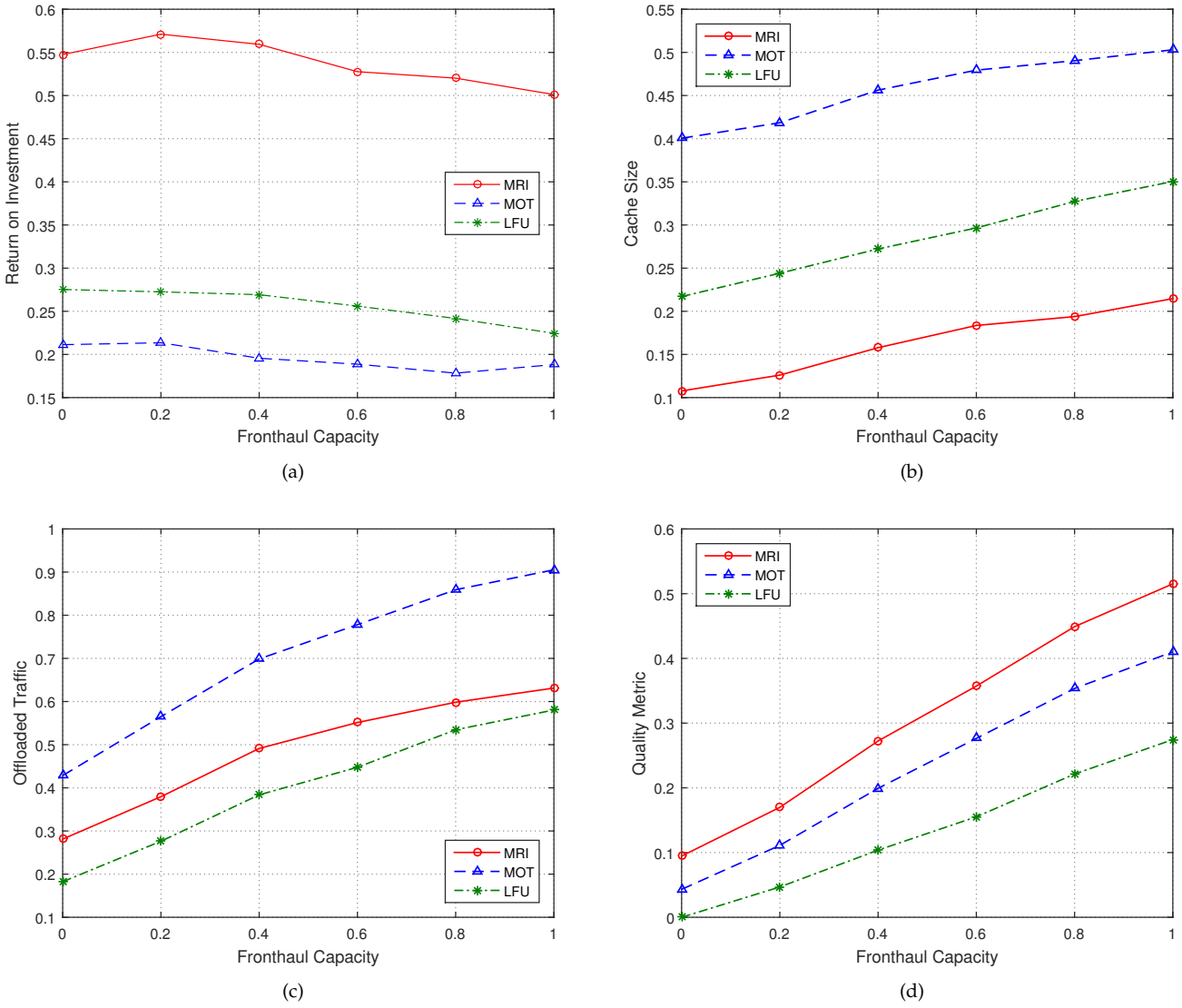

Fig. 4. Scenario 2 - varying fronthaul capacity: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

investment and LFU, which only considers popularity. This in turn causes MOT and MRI to have the lowest and highest storage efficiency, respectively [see Fig. 4(b)].

We note that after a certain increase in the budget, LFU reaches a steady state as it has already cached the contents with highest popularity. Caching more contents requires higher fronthaul capacity, which is set to 25 Gbps in this scenario. However, since MOT takes both popularity and size of the objects into consideration, further increase in the budget results in availability of more storage. This leads MOT to cache larger contents (consequently lower storage efficiency). However, it achieves a considerable gain in offloaded traffic. Having cached higher quality representations, MOT exhibits a better performance in terms of quality when compared to LFU.

5.4 Summary

TABLE 4 presents a comparison of the average performance of the three caching algorithms under all scenarios. In summary, MRI outperforms the other schemes in terms of return on investment, cache storage efficiency and quality. In comparison with MOT and LFU, MRI results in an average improvement of 30.58% and 32.23% in return on investment, 31.63% and 17.46% in storage efficiency and 12.47% and 21.43% in quality, respectively. On the other hand, MOT has the best performance with regard to the increase in overall offloaded traffic. It outperforms MRI by 23.45% and LFU by 32.7% .

6 Complexity and Optimality Analysis

Lastly, we discuss in this sub-section the complexity of the IWO algorithm and provide some further insights regarding the optimality or, in other words, the competitiness of the solutions. IWO is an iterative algorithm and is used for each dual variable associated with the dual function in (17). In each iteration for $\boldsymbol{\lambda}^* \geq 0, \boldsymbol{\mu}^* \geq 0, \boldsymbol{\nu}^* \geq 0$, we compute N, NA and N variables, respectively. Therefore, it has an overall worst case complexity of $\mathcal{O}(iter_{\max} \cdot \{2N + NA\})$ [33]. A comprehensive assessment of the performance of

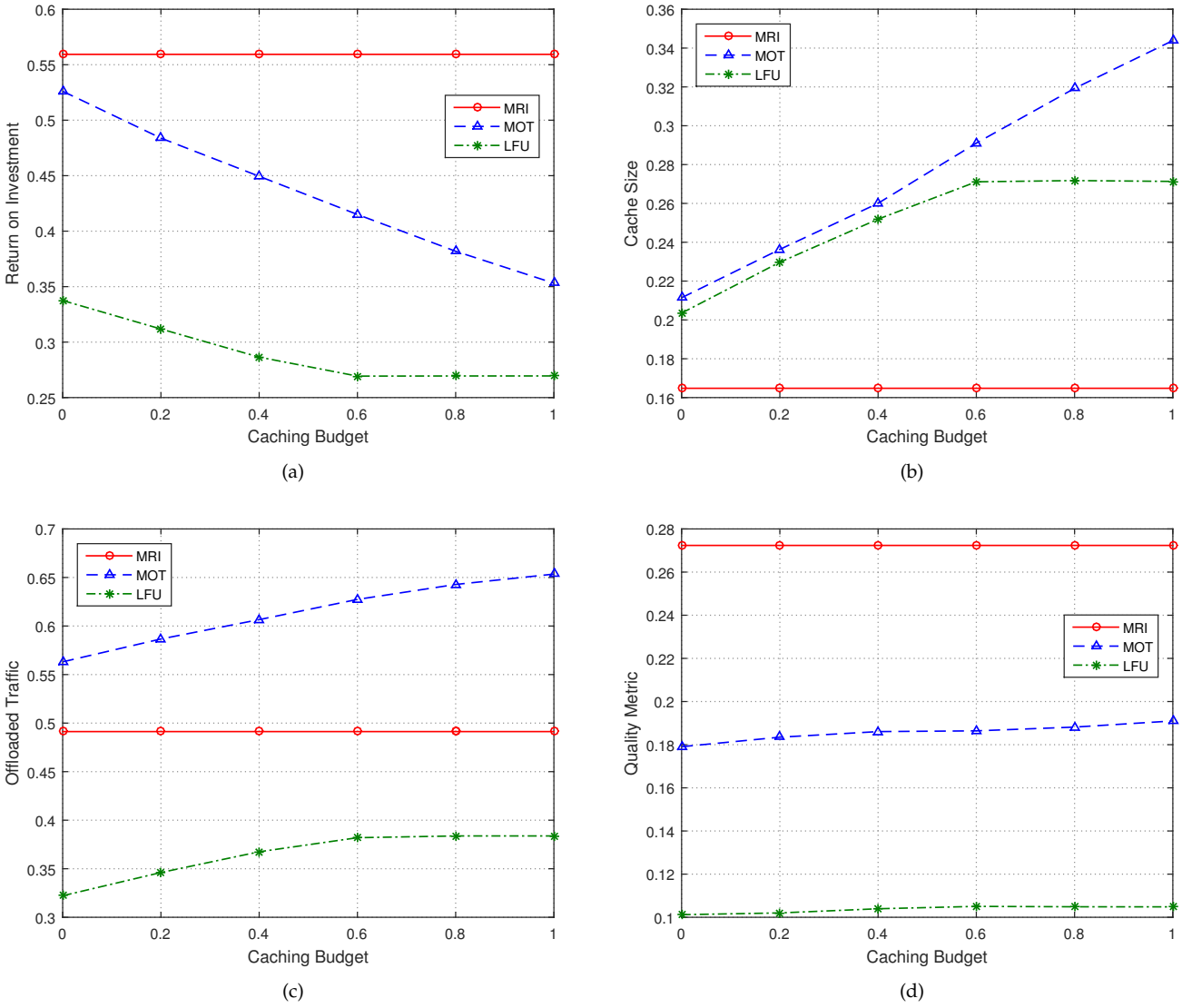

Fig. 5. Scenario 3 - varying caching budget: (a) return on investment; (b) cache size; (c) offloaded traffic; (d) quality metric.

TABLE 4
Average Performance Comparison of Caching Techniques

Metric[1]	BPA[2]	MRI (%)	MOT (%)	LFU (%)
ROI	MRI	-	+30.58	+32.23
CS	MRI	-	−31.63	−17.46
OT	MOT	+23.45	-	+32.7
QM	MRI	-	+12.47	+21.43

[1]ROI: return on investment; CS: cache size; OT: offloaded traffic; QM: quality metric.
[2]BPA: best performing algorithm.

IWO algorithm in terms of convergence and computational time can be found in [28] and [34].

With respect to optimality, the use of such metaheuristic framework provides us, unavoidably, solutions that might not be optimal. However, as numerical investigations reveal the solutions found lead to significant additional improvements in the overall system performance compared to simple greedy based algorithms such as LFU. The high quality of the proposed solutions compared to current well-used greedy algorithms and the computational efficiency of finding them, strongly supports the potential application of the proposed framework in a real-world settings.

7 Conclusion

In this paper, we have proposed a CaaS framework for virtual caching in the MNO's infrastructure. Our first proposed scheme caches video contents in the cloud-based mobile network with the aim of maximizing the return on caching investment. Our second approach aims at maximizing the offloaded traffic as a result of caching for a given caching budget. We use CDT to convert our BIP virtual caching problem into its canonical dual. We use the IWO algorithm to obtain the solution of the dual problem. Numerical and simulation results have shown that the proposed framework outperforms LFU algorithm by more than 32%, 21%, 32%

and 17% improvements in terms of return on investment, quality, offloaded traffic and storage efficiency, respectively.

A possible future avenue of research would be to consider the effect of cache sharing between different tenants in virtualized sliced network architectures where the cost as defined will change as well as the effect on fronthaul/backhaul sliced capacity per tenant. It is also interesting to understand the impact of temporal dynamics of access patterns and content popularity on the performance of the model and elaborate on the reactive mechanisms for on-the-fly adjustments of the caching strategies in accordance to dynamically changing conditions. Finally, the future iterations of the proposed model might feature transition towards distributed optimization where caching decisions are made in a decentralized fashion between a multitude of co-operating counterparts.

REFERENCES

[1] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper*, Cisco White Paper, February 2017.

[2] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, March 2012, pp. 1107–1115.

[3] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, August 2014.

[4] H. Ahlehagh and S. Dey, "Video-aware scheduling and caching in the radio access network," *IEEE/ACM Trans. Netw.*, vol. 22, no. 5, pp. 1444–1462, October 2014.

[5] S. Spagna, M. Liebsch, R. Baldessari, S. Niccolini, S. Schmid, R. Garroppo, K. Ozawa, and J. Awano, "Design principles of an operator-owned highly distributed content delivery network," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 132–140, 2013.

[6] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, 2014.

[7] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz, "EASE: EPC as a service to ease mobile core network deployment over cloud," *IEEE Netw.*, vol. 29, no. 2, pp. 78–88, March 2015.

[8] X. Li, X. Wang, C. Zhu, W. Cai, and V. Leung, "Caching-as-a-service: Virtual caching framework in the cloud-based mobile networks," in *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*, April 2015, pp. 372–377.

[9] Authored by network operators, "Network functions virtualisation: An introduction, benefits, enablers, challenges & call for action," in *SDN and OpenFlow World Congress*, 2012, pp. 22–24.

[10] K. Chen and R. Duan, "C-RAN–the road towards green RAN," White Paper, China Mobile Research Institute, 2011.

[11] A. Gharaibeh, A. Khreishah, B. Ji, and M. Ayyash, "A provably efficient online collaborative caching algorithm for multicell-coordinated systems," *IEEE Trans. Mobile Comput.*, vol. PP, no. 99, September 2015.

[12] Z. Li and G. Simon, "In a Telco-CDN, pushing content makes sense," *IEEE Trans. Netw. and Serv. Manag.*, vol. 10, no. 3, pp. 300–311, September 2013.

[13] M. Hu, J. Luo, Y. Wang, and B. Veeravalli, "Practical resource provisioning and caching with dynamic resilience for cloud–based content distribution networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2169–2179, August 2014.

[14] D. K. Krishnappa, S. Khemmarat, L. Gao, and M. Zink, "On the feasibility of prefetching and caching for online tv services: a measurement study on hulu," in *Passive and Active Measurement*. Springer, 2011, pp. 72–80.

[15] W. H. O. Lau, M. Kumar, and S. Venkatesh, "A cooperative cache architecture in support of caching multimedia objects in MANETs," in *Proceedings of the 5th ACM International Workshop on Wireless Mobile Multimedia*, ser. WOWMOM '02. New York, NY, USA: ACM, 2002, pp. 56–63.

[16] L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 1, pp. 77–89, January 2006.

[17] L. Wang, K.-K. Wong, S. Jin, G. Zheng, and R. Heath, "A new look at physical layer security, caching, and wireless energy harvesting for heterogeneous ultra-dense networks," *IEEE Commun. Mag.*, vol. 56, 01 2018.

[18] W. Jiang, G. Feng, and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[19] R. Buyya, J. Broberg, and A. M. Goscinski, *Cloud computing: principles and paradigms*. John Wiley & Sons, 2010, vol. 87.

[20] X. Li, X. Wang, K. Li, Z. Han, and V. C. Leung, "Collaborative multi-tier caching in heterogeneous networks: Modeling, analysis, and design," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6926–6939, 2017.

[21] X. Li, X. Wang, K. Li, and V. C. Leung, "Caas: Caching as a service for 5g networks," *IEEE Access*, vol. 5, pp. 5982–5993, 2017.

[22] S. Zhang, N. Zhang, P. Yang, and X. Shen, "Cost-effective cache deployment in mobile heterogeneous networks," *IEEE Trans. Vehicular Technol.*, vol. 66, no. 12, pp. 11 264–11 276, Dec 2017.

[23] Z. Zhang, C.-H. Lung, I. Lambadaris, and M. St-Hilaire, "When 5g meets icn: An icn-based caching approach for mobile video in 5g networks," *Comput. Commun.*, 2017.

[24] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 362–373, August 2011.

[25] S. Krishnan and R. Sitaraman, "Video stream quality impacts viewer behavior: Inferring causality using quasi-experimental designs," *IEEE/ACM Trans. Netw.*, vol. 21, no. 6, pp. 2001–2014, December 2013.

[26] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[27] D. Yang Gao, "Canonical dual transformation method and generalized triality theory in nonsmooth global optimization," *J. Global Optim.*, vol. 17, no. 1–4, pp. 127–160, 2000.

[28] A. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.

[29] V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 314–329, August 1988.

[30] D. Yang Gao, N. Ruan, and H. D. Sherali, "Canonical dual solutions for fixed cost quadratic programs," in *Optimization and optimal control*. Springer, 2010, pp. 139–156.

[31] D. Y. Gao, R. lin Sheu, S. yi Wu, and K. L. Teo, "Canonical dual approach for solving 0-1 quadratic programming problems," *J. Ind. Manag. Optim.*, vol. 4, pp. 125–142, 2007.

[32] A. Ahmad and M. Assaad, "Polynomial-complexity optimal resource allocation framework for uplink SC-FDMA systems," in *Proc. IEEE Global Telecoms. Conf. (GLOBECOM)*. IEEE, December 2011, pp. 1–5.

[33] A. Aijaz, M. Tshangini, M. Nakhai, X. Chu, and A.-H. Aghvami, "Energy-efficient uplink resource allocation in LTE networks with M2M/H2H co-existence under statistical QoS guarantees," *IEEE Trans. Commun.*, vol. 62, no. 7, pp. 2353–2365, July 2014.

[34] E. Pourjafari and H. Mojallali, "Solving nonlinear equations systems with a new approach based on invasive weed optimization algorithm and clustering," *Swarm and Evolutionary Computation*, vol. 4, pp. 33–43, 2012.

[35] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Youtube traffic characterization: A view from the edge," in *ACM SIGCOMM Conf. Internet Meas.* New York, NY, USA: ACM, October 2007, pp. 15–28.

[36] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "Analyzing the video popularity characteristics of large-scale user generated content systems," *IEEE/ACM Trans. Netw.*, vol. 17, no. 5, pp. 1357–1370, October 2009.

[37] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, "Impact of traffic mix on caching performance in a content-centric network," in *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*, March 2012, pp. 310–315.

[38] J. Wang, "A survey of web caching schemes for the internet," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 29, no. 5, pp. 36–46, October 1999.