This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2025.3622056

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

Online Resource-Aware Video Content Recommendation in Edge-Caches for Mobile Users

ANA CLAUDIA BASTOS LOUREIRO MONÇÃO[1], KARLLA B. CHAVES RODRIGUES[1], SAND LUZ CORRÊA[1] AND KLEBER VIEIRA CARDOSO[1]
[1]Federal University of Goiás (UFG), Goiânia, 74690-900, Brazil

Corresponding author: Karlla B. Chaves Rodrigues (e-mail: karllarodrigues@inf.ufg.br).

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), by the MCTIC/CGI.br/São Paulo Research Foundation (FAPESP) through the Smart 5G Core and MUltiRAn Integration (SAMURAI) Project under Grant 2020/05127-2, by the Slicing Future Internet Infrastructures (SFI2) Project under Grant 2018/23097-3, and by the RNP/MCTIC through the Brasil 6G Project under Grant 01245.020548/2021-07.

ABSTRACT The coupling of content caching at the wireless network edge and video streaming recommendation systems has been thoroughly investigated to enhance the cache hit and improve the user quality of experience (QoE). However, the existing literature lacks studies addressing the joint problem of QoE and cache hit ratio maximization while considering device characteristics and dynamic network resources of mobile users. This study introduces On-RAViR, an online framework comprising a Channel Quality Indicator (CQI) prediction module and a heuristic algorithm. This framework aims to maximize both cache hit ratio and user QoE under two constraints: the quality of the user equipment (UE) wireless link and the computing capabilities of the UE. We evaluate our framework employing a real-world video content dataset and a third-party 5G trace dataset. The results demonstrate that our framework produces rapid and high-quality solutions, increasing user QoE by 20% on average when compared to a state-of-the-art caching and recommendation heuristic unaware of computing and network resources.

INDEX TERMS Edge computing, quality of experience, recommendation systems, user mobility, video caching.

I. INTRODUCTION

According to projections, by the end of 2024, video will account for 74% of the total mobile network traffic [1]. To meet this growing demand without increasing backbone traffic, network and video content service providers are pushing caching capabilities to the edge of the wireless networks, a concept known as edge caching [2], [3].

In edge caching, frequently requested video contents are cached close to the end users on edge servers. These servers can be co-located with base stations, integrated with devices in the radio access network (RAN), or located in the core network [4]. Indeed, considering on-demand video streaming, which is the focus of this work, edge caches are accessible at a maximum of six IP hops from the user [5]. Therefore, properly caching video content on edge servers can significantly reduce the number of repeated requests to video streaming servers on the backbone (i.e., in the cloud) and reduce transmission service delay, thus improving the quality of the user experience (QoE). However, due to the limited storage capacity of edge servers compared to cloud servers, determining which video content to cache is crucial to the edge cache performance.

To address this issue, researchers have begun investigating the impact of video streaming recommendation systems on caching. Recommendation systems significantly influence users' content choices [6]. Consequently, they can be used in a controllable way (i.e., within acceptable tolerance limits) to induce access patterns that benefit different performance metrics, such as the cache hit ratio [7]–[9], the user QoE [10], the offloading probability [11], and the total content delivery cost [12], [13]. In the following subsection, we discuss relevant works that jointly optimize edge cache and video content recommendation decisions to enhance the cache hit ratio and the user QoE in the context of video streaming services.

A. RELATED WORK

The first work to take caching decisions based on personalized recommendation systems for edge caching was Sermpezis et al. [7]. The authors proposed the concept of "soft cache hits" wherein the idea is to recommend similar cached content to mobile users requesting videos not cached at the edge cache. The work presented optimization problems related to soft hits and cache placements in wireless networks and algorithms to solve them. Chatzieleftheriou et al. [8] evolved the initial idea introduced in [7] and proposed a model to tackle the Joint Caching and Recommendation Problem (JCRP) in small cell networks. In JCRP, the recommendation system is used as a demand-shaping tool to enhance the cache hit ratio. To this end, the JCRP model issues video content recommendations that may not necessarily rank top in the inferred user preferences but still score high. By limiting the distortion that such recommendations introduce to the original user content preferences, the JCRP model controllably guides user demand toward content that attracts the preference of multiple users. The work showed that JCRP brings relevant gains in the cache hit ratio without significantly degrading the user preferences. Different from [8], where the authors assumed that users fully accept the recommendation lists, in [14], the authors jointly optimized content caching and recommendation at wireless networks by setting a ratio to reject the recommendations for each user. The relationship between quality of service (QoS), quality of recommendation (QoR), and user QoE was studied by Nogueira et al. [15] to understand better how to make joint cache and video recommendation decisions. Using experiments with real users, the work confirmed that content preferences bound QoS-aware recommendations, i.e., users do not tolerate large content discrepancies. The authors also showed that considering the scenarios where recommendations can account for QoS, most content has at least one other content that is similar to them, and it is cached.

The interplay between recommendation and cache decision was also investigated in device-to-device (D2D) communications with the goal of maximizing the cache hit ratio [16], [17]. The authors in [17] showed that this problem is NP-hard and proposed a polynomial time complexity optimization algorithm with convergence guarantee. In [16], an algorithm based on long short-term memory (LSTM) was proposed to solve the problem.

In the works discussed so far, recommendations are defined only by content and cache hit. Other important QoS indicators are not considered, such as video resolution and the computing and network resources available on the user side to play the video. As a result, such works may recommend items whose representations are unsuitable for the available user resources, degrading user satisfaction [18]. In addition, video resolution severely affects the cache occupancy and, thus, the cache hit. To fill this gap, in [10], we formulated and optimally solved an optimization problem called Resource-Aware Video Recommendation (RAViR) for jointly caching and recommending videos to mobile users that maximizes the cache hit ratio and the user QoE under two constraints: the user equipment (UE) computing capabilities and the quality of the UE wireless link at the time of the recommendation. Results showed that RAViR increases user QoE and cache hit ratio by at least 68% and 14%, respectively, compared with JCRP. A similar problem was later investigated in [19] but with a bundle approach.

Regarding QoE in video streaming services, many papers have investigated how to measure the user QoE when they consume on-demand video streaming services. Some works used implicit metrics related to user engagement [20] or the cache hit ratio [8]. Many works assess the user QoE correlating explicit metrics (usually QoS metrics) with implicit metrics such as MOS and VMAF, both widely used in the literature. Mean opinion score (MOS) [21] is a video quality assessment metric where the user evaluates her experience from a minimum to a maximum value (usually from 1 to 5) when watching a video. Video multimethod assessment fusion (VMAF) [22] is a metric developed by Netflix to predict the perceived visual quality of a video by comparing it to the original reference video. In [21], the authors use a congestion index (CI) to assess the ability of the network to successfully deliver video streams based on the maximum available bandwidth on the path from the server to the user. The QoE is then obtained using a regression technique correlating the CIs with MOS. In [15], the authors used the VMAF score to directly gauge the impact of video impairments on QoE and to determine the influence of network conditions on QoE indirectly.

Finally, it is important to mention that the interplay between QoS, QoR, and QoE has also motivated recent works on similarity caching and cost-aware caching (e.g., [23]) where both the similarity between content and the cost to serve them is determined to make cache decisions. However, although such topics bring novel opportunities and challenges in edge caching, they are out of the scope of this article.

B. OUR CONTRIBUTION

Although resource-aware joint caching and recommendation can achieve better results than conventional joint caching and recommendation solutions, the existing works [10], [19] assumed that user request patterns for content change slowly over time, typically on the order of hours. However, in scenarios involving user mobility, the quality of the wireless link may change at a much smaller scale, influencing video requests, recommendations, and, consequently, the cache allocation. An optimal solution in such scenarios is impractical since the resource-aware joint caching and recommendation problem is NP-complete. Thus, as the number of users and videos in the catalog increases, the time needed to solve the problem optimally increases rapidly. Therefore, an online solution is desired, which motivates our work. In this article, we propose On-RAViR, an online framework consisting of a channel quality indicator (CQI) prediction module and a heuristic algorithm to solve the resource-aware joint caching and recommendation problem. Before recommending a video for the user, the prediction module in our framework forecasts the quality of the user's wireless connection based on the user's CQI history, i.e., the CQI values observed for the user connection before the recommendation or during the last watched video. A heuristic algorithm then uses the estimated CQI value, the UE computing capabilities, the user content preferences, and the cached videos to recommend a video for the user. We also propose a new metric that combines QoE and QoR, the latter influenced by QoS parameters (availability of computing and network resources), contributing to the discussion on a unified metric that correlates QoS, QoE, and QoR. We evaluate On-RAViR using real-world video content and third-party 5G trace datasets. Results demonstrate that On-RAViR produces fast and high-quality solutions compared with RAViR. Results also show that On-RAViR increases user QoE by 20% (on average) compared to JCRP-HEU. In summary, the main contributions of this article are:

- We revisit the formulation of the resource-aware joint caching and recommendation problem.
- We design an efficient online solution (On-RAViR) for the problem that can be used in scenarios involving user mobility and frequent recommendation requests.
- We design a new metric to assess the QoE of mobile users during the recommended video playback period. The QoE metric is based on MOS and the Weber-Fechner law [24].
- We evaluate On-RAViR using real-world, third-party 5G data traces.
- We thoroughly evaluate On-RAViR, comparing it with the optimal algorithm (RAViR) and a state-of-the-art solution modified to operate online (JCRP-HEU) in different scenarios.

This article revisits and extends our previous work [10]. In particular, the online framework (On-RAViR), the proposed QoE metric, and the framework's performance evaluation are completely new contributions. This article is organized as follows. Section II describes the system model. The problem formulation and the design of On-RAViR are presented in Section III. Section IV presents the evaluation and discusses the results. Section V concludes the article and presents future work.

II. SYSTEM MODEL

We consider an on-demand video streaming provider (e.g., Netflix) offering services within a mobile network infrastructure controlled by a mobile network provider. The video streaming service uses a catalog of video items, where each item represents a video content (e.g., The Godfather, Pulp Fiction, etc.) encoded in a specific representation (e.g., medium, large, hd720, etc.). The service also includes a recommendation system for its users. The mobile network infrastructure comprises a multi-access edge computing (MEC) system. In this infrastructure, several base stations (BSs) offer connectivity to mobile users who consume the video streaming service via smartphones (UEs) connected to the mobile network. Edge servers deployed at the core network host a cache service. Due to limited storage capacity, the cache stores only a subset of the catalog at any given time, while servers in remote data centers retain copies of the catalog. Figure 1 illustrates the reference scenario considered in this article.

A. CATALOG

We denote by $\mathcal{V}$ the set of video content and by $\mathcal{R}$ the set of all possible video representations. Each video representation $r \in \mathcal{R}$ is characterized by a bitrate $btr(r)$ and a resolution $res(r)$. We denote by $\mathcal{I}$ the catalog of video items, where each video item $i_{v,r} \in \mathcal{I}$ represents a video content $v \in \mathcal{V}$ encoded into a representation $r \in \mathcal{R}$, with an encoding bitrate $btr(r)$, a corresponding video resolution $res(r)$, and a size $size(i_{v,r})$. We denote by C the total storage capacity of the edge cache, measured in normalized file size units.

B. COMPUTING AND NETWORK RESOURCE

We denote the set of mobile users by $\mathcal{U}$. Each user $u \in \mathcal{U}$ consumes the video streaming service through a UE connected to a BS. The UE of user u has a certain computing capability (e.g., processor type, graphic card, memory) and, consequently, can support a maximum resolution for video playback, denoted by ψ_u. The quality of the user's UE wireless link may vary over time due to several factors, including signal impairments. As illustrated in Figure 1, we consider that the quality of the wireless link can be periodically obtained by the video streaming service via the MEC radio network information service (RNIS) [25]. This quality is reported as the CQI, one of the key parameters in the channel state information (CSI) [26]. We denote the set of all possible CQI values by $\mathcal{Q}$. Each $q \in \mathcal{Q}$ is associated with a pair of modulation and coding scheme (MCS) and transport block size (TBS) by means of which it is possible to determine an average available bandwidth (bitrate) [27], denoted by $\phi(q)$.

C. CONTENT PREFERENCE

We denote by $\mathcal{T}$ the set of all possible thematic categories. Each content $v \in \mathcal{V}$ is associated with a feature vector $\mathbf{f}^v$ whose j-th element $\mathbf{f}^v(j)$, $j \in \mathcal{T}$, represents the adherence level of category j to content v. This adherence level is normalized and assumes values in the range [0,1]. Each user $u \in \mathcal{U}$ is also associated with a feature vector $\mathbf{f}^u$, where each element $\mathbf{f}^u(j)$, $j \in \mathcal{T}$, represents the interest of user u in thematic category j. This interest is estimated based on the video content watched and evaluated by the user, denoted by $\mathcal{A}_u \subseteq \mathcal{V}$. Each element $\mathbf{f}^u(j)$ of the user feature vector $\mathbf{f}^u$ is estimated as $\mathbf{f}^u(j) = \sum_{v \in \mathcal{A}_u} \mathbf{f}^v(j) / \sum_{j \in \mathcal{T}} \sum_{v \in \mathcal{A}_u} \mathbf{f}^v(j)$. This value is also normalized in the range [0,1]. Aligned with state-of-the-art recommendation systems (e.g. [8]), we compute the interest of user u in the video content v as:

$$Sim(u,v) = \begin{cases} 0, & \text{if } u \text{ watched } v, \\ \frac{\sum_{j \in \mathcal{T}} \mathbf{f}^u(j) \cdot \mathbf{f}^v(j)}{\sqrt{\sum_{j \in \mathcal{T}} \mathbf{f}^u(j)^2} \cdot \sqrt{\sum_{j \in \mathcal{T}} \mathbf{f}^v(j)^2}} & \text{otherwise.} \end{cases} \tag{1}$$

Thus, for each user $u \in \mathcal{U}$ and each video content $v \in \mathcal{V}$, the probability of user u being interested in the video content v is

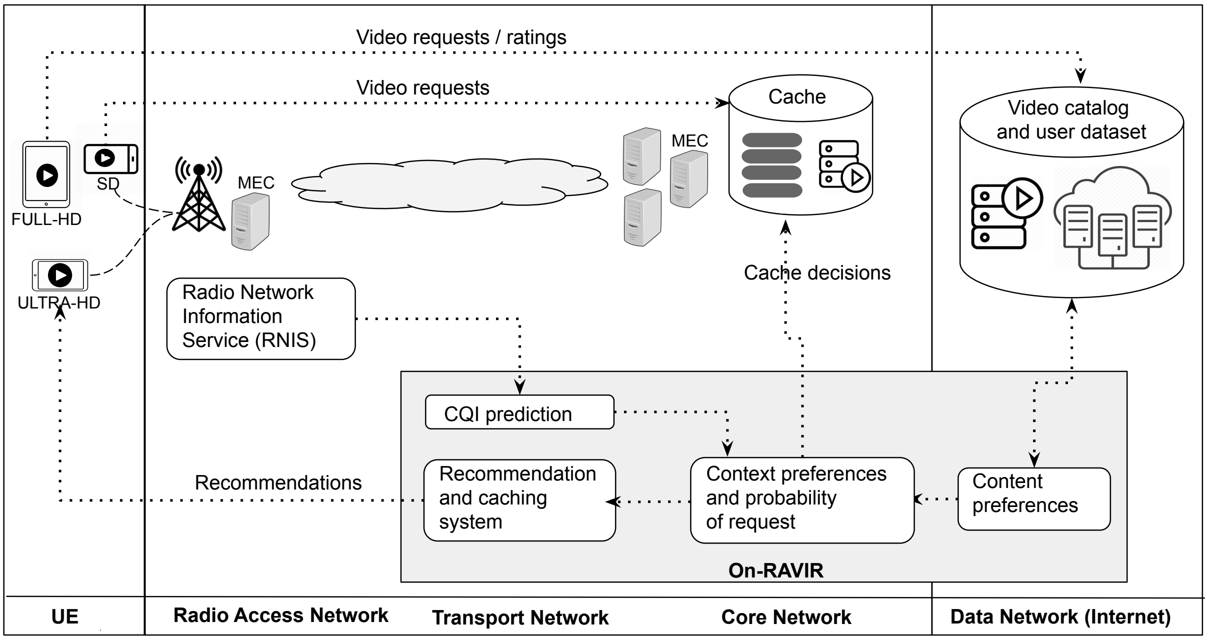


FIGURE 1: Video streaming recommendation scenario for mobile users using edge caches.

computed as:

$$P_u^{Cont}(v) = \frac{Sim(u,v)}{\sum_{z \in \mathcal{V}} Sim(u,z)}. \tag{2}$$

This distribution is normalized, i.e., $\sum_{v \in \mathcal{V}} P_u^{Cont}(v) = 1$.

D. REPRESENTATION PREFERENCES

We assume that previous CQI records describing the quality of the UE wireless connection are available through the RNIS service. Using the historical series of the observed CQI values, the system estimates the CQI. This CQI represents the average quality of the wireless connection the user will experience during the video playback. We denote this estimate by $q_u \in \mathcal{Q}$. Section III explains how we calculate this estimate in our framework. Given q_u, we calculate the highest video resolution that can be transmitted over the UE's wireless connection during the video playback as follows: $res(q_u) = \max\{res(r) \mid r \in \mathcal{R} \wedge btr(r) \leq \phi(q_u)\}$.

To capture the relevance of a given video representation $r \in \mathcal{R}$ to user u, we use a relevance factor denoted by $Sat(u,r) \in [0,1]$ given by:

$$Sat(u,r) = \begin{cases} 0, & res(r) > best(u) \\ 1, & res(r) = best(u) \\ 1 - \frac{(best(u) - res(r))}{best(u)}, & res(r) < best(u) \end{cases} \tag{3}$$

where $best(u) = \min\{\psi_u, res(q_u)\}$ is the best video resolution for user u according to the UE resolution and the estimated quality of the UE wireless link. Thus, for each user $u \in \mathcal{U}$ and each representation $r \in \mathcal{R}$, the probability that the user u is interested in the representation r is calculated as follows:

$$P_u^{Res}(r) = \frac{Sat(u,r)}{\sum_{s \in \mathcal{R}} Sat(u,s)}. \tag{4}$$

This distribution is normalized, so that $\sum_{r \in \mathcal{R}} P_u^{Res}(r) = 1$.

E. ULTIMATE USER PREFERENCES

We then define a user preference distribution, denoted by P_u^{Pref}, which captures the user's joint preferences u over a video content $v \in \mathcal{V}$ and a video representation $r \in \mathcal{R}$, for all video items in the catalog, i.e.,

$$P_u^{Pref}(i_{v,r}) = \frac{P_u^{Cont}(v) \cdot P_u^{Res}(r)}{\sum_{i_{z,s} \in \mathcal{I}} P_u^{Cont}(z) \cdot P_u^{Res}(s)}, \tag{5}$$

where $z \in \mathcal{V}$ and $s \in \mathcal{R}$ are, respectively, the content and the representation associated with an item $i_{z,s} \in \mathcal{I}$. This distribution is normalized so that $\sum_{i_{v,r} \in \mathcal{I}} P_u^{Pref}(i_{v,r}) = 1$. Equation (5) comprises the preferences of user u in terms of video content and representation. Thus, the higher the value of $P_u^{Pref}(i_{v,r})$, the better the user experience.

However, recommendation systems strongly influence users' content choices. To account for this impact, we define the probability distribution P_u^{Rec} over the set $\mathcal{V}$ of video content. Let $\mathcal{V}_u \subseteq \mathcal{V}$ be the set of content recommended to user u, where $\mathcal{V}_u$ consists of the top N ($N = |\mathcal{V}_u|$) content in the user content preference distribution P_u^{Cont}. As in other works on recommendation systems (e.g., [8]), we assume that recommendations provide all N content in $\mathcal{V}_u$ with an equal boost, i.e., $P_u^{Rec}(v) = 1/N, \forall v \in \mathcal{V}_u$. Let α_u be the weight expressing the importance that the user u attaches to recommendations. We define the ultimate video request distribution as:

$$P_u^{Req}(i_{v,r}) = \alpha_u \cdot P_u^{Rec}(v) + (1 - \alpha_u) \cdot P_u^{Pref}(i_{v,r}), \tag{6}$$

for the videos that are recommended to the user u, and

$$P_u^{Req\sim}(i_{v,r}) = (1 - \alpha_u) \cdot P_u^{Pref}(i_{v,r}), \tag{7}$$

for the videos that are not recommended to the user u.

F. RECOMMENDATION AS TRAFFIC ENGINEERING

Instead of issuing recommendations for the top N items on the user content preference distribution P_u^{Cont}, as is generally the case for a recommendation system, we select N items from a subset of a recommendation window $\mathcal{WC}_u \subset \mathcal{V}$, defined by the top N_u content of P_u^{Cont}, where $N_u > N$. The N_u value is defined as a function of a preference distortion tolerance $tol(u) \in [0, 1)$ that quantifies the extent to which recommendations can distort the user's initial preferences. Higher $tol(u)$ values allow more flexibility in selecting items recommended to users, thus promoting caching efficiency.

III. PROBLEM FORMULATION AND ON-RAVIR SOLUTION

Let $\mathcal{WI}_u$ be a subset of the catalog containing all the video items whose content is contained in the recommendation window $\mathcal{WC}_u$, i.e, $\mathcal{WI}_u \subset \mathcal{I} = \{i_{v,r} \mid v \in \mathcal{WC}_u\}$. Let $\{y_{i_{v,r}}\}$, $i_{v,r} \in \mathcal{I}$, be a set of binary decision variables, so that $y_{i_{v,r}} = 1$ if video item $i_{v,r}$ is in cache and $y_{i_{v,r}} = 0$, otherwise. Let $\{x_{u,i_{v,r}}\}$, $u \in \mathcal{U}, i_{v,r} \in \mathcal{I}$, be another set of binary decision variables, so that $x_{u,i_{v,r}} = 1$ if $i_{v,r}$ is recommended to user u and $x_{u,i_{v,r}} = 0$, otherwise. We formulate the resource-aware joint caching and recommendation (RAViR) problem as follows:

$$\max_{y,x} \sum_{u\in\mathcal{U}} \sum_{i_{v,r}\in\mathcal{WI}_u} y_{i_{v,r}}(x_{u,i_{v,r}} \cdot P_u^{Req}(i_{v,r}) + (1 - x_{u,i_{v,r}}) \cdot P_u^{Req\sim}(i_{v,r})) \tag{8}$$

subject to:

$$\sum_{i_{v,r}\in\mathcal{I}} y_{i_{v,r}} \cdot size(i_{v,r}) \leq C \tag{9}$$

$$\sum_{i_{v,r}\in\mathcal{WI}_u} x_{u,i_{v,r}} = N, \ \forall u \in \mathcal{U} \tag{10}$$

$$\sum_{i_{z,r}\in\mathcal{WI}_u} x_{u,i_{z,r}} \cdot \delta(z, v) \leq 1 \ \forall v \in \mathcal{WC}_u, \ \forall u \in \mathcal{U} \tag{11}$$

$$\mathcal{A}_u \neq \varnothing, \ \forall u \in \mathcal{U} \tag{12}$$

$$P_u^{Res}(r) > 0, \ \forall u \in \mathcal{U}, \ \forall i_{v,r} \in \mathcal{WI}_u. \tag{13}$$

Equation (8) maximizes the cache hit ratio and the user experience using each user list $\mathcal{WI}_u$. Equation (9) reflects the cache storage capacity constraint, while Equation (10) ensures that exactly N videos are recommended to each user. In Equation (11), $\delta(z, v)$ is an indication function that equals 1 if $z = v$, and 0 otherwise. This equation ensures that, for each user, only one representation will be selected for each recommended content. Finally, equation (13) ensures that the recommended video item does not exceed each user's computing and network resources.

The RAViR problem is NP-complete. Each item $i_{v,r}$ in catalog $\mathcal{I}$ has a size $size(i_{v,r})$ and a value given by $\sum_{u\in\mathcal{U}} x_{u,i_{v,r}} \cdot P_u^{Req}(i_{v,r}) + (1 - x_{u,i_{v,r}}) \cdot P_u^{Req\sim}(i_{v,r}))$. The items have to be packed in a cache space C, maximizing the total cache value while not exceeding the cache capacity. Thus, the RAViR problem corresponds to the 0-1 Knapsack Problem (KSP), a known NP-complete problem [28].

In [10], we could optimally solve the RAViR problem for instances of relevant size because we had enough time between recommendation requests. Using the MovieLens dataset [8], the optimal algorithm (RAViR) takes approximately 24 minutes to solve the problem in a computer with a quad-core Intel Core i7 @ 2.7 GHz and 16 GB RAM. In this work, we consider a more realistic scenario where mobile users demand recommendations at a much lower time scale, in the order of seconds. Thus, we developed an online framework called On-RAViR to obtain satisfactory solutions more quickly. On-RAViR proceeds in two steps, as described below.

In the first step, for each user u needing a recommendation at time t, the CQI predictor estimates the average wireless connection quality that the user will experience during the video content playback, i.e., q_u. This estimate is calculated by obtaining (from the RNIS) the CQI values observed for u in the last s_q time instants, i.e., $H(u, k)$, $k = t - 1, \ldots, t - s_q$. This time series is then used to estimate $CQI(u, t), CQI(u, t + 1), \ldots, CQI(u, t + s_u)$, where s_u is the average size (in seconds) of the videos watched by u. Each $CQI(u, l)$, $l = t, \ldots, t + s_u$, is computed as a moving average of size m, i.e.:

$$CQI(u, l) = \sum_{k=l-s_q}^{m} H(u, k)/m. \tag{14}$$

The estimate q_u is then calculated as the average of these estimates, i.e.:

$$q_u = \sum_{l=t}^{t+s_u} CQI(u, l)/(s_u + 1). \tag{15}$$

We must stress that we also evaluated some machine learning (ML) methods, such as LSTM, for prediction in addition to the moving average. Nonetheless, for this problem, where we want to estimate a single CQI value for the period in which the video will be played, the ML methods produced similar results to the moving average but at a higher computing cost. Thus, we used the moving average as the estimator because of its simplicity, efficacy, and efficiency. These characteristics are important since On-RAViR is intended to operate online.

Once q_u is estimated for each user u requiring a recommendation, in the second step, a heuristic algorithm solves the RAViR problem. This algorithm is based on the assumption that the video items that appear most often in the users' $\mathcal{WI}_u$ serve the largest number of users and, presumably, those that will be most requested by these users. Therefore, for each video item $i_{v,r}$ in the catalog $\mathcal{I}$, we calculate a utility for $i_{v,r}$ based on the number of times this video item appears in users' $\mathcal{WI}_u$, i.e.:

$$util(i_{v,r}) = \sum_{u\in\mathcal{U}} \sigma(i_{v,r}), \tag{16}$$

where $\sigma(i_{v,r}) = 1$ if $i_{v,r} \in \mathcal{WI}_u$, and $\sigma(i_{v,r}) = 0$ otherwise. We then fill the cache with the most useful video items and make recommendations based on the users' $\mathcal{WI}_u$, prioritizing the cached video items.

IV. EVALUATION

This section evaluates the performance of On-RAViR using a video catalog and a 5G trace obtained from real-world data. Section IV-A describes how the video catalog is derived. Section IV-B presents how the UE capability and the CQI history are obtained. The QoE metric used to evaluate On-RAViR is presented in Section IV-C. In Section IV-D, we compare On-RAViR with the optimal algorithm for the RAViR problem (RAViR) and also with a heuristic for the JCRP (JCRP-HEU), presenting the results.

A. VIDEO CATALOG AND CONTENT PREFERENCES

We use the MovieLens project dataset [29] to obtain video content and user content preferences. This dataset consists of approximately 100,000 user ratings (ranging from 0 to 5) applied to a content catalog of 9,742 movies ($|\mathcal{V}| = 9{,}742$), comprising 20 thematic categories ($|\mathcal{T}| = 20$), and a community of 610 users ($|\mathcal{U}| = 610$). Based on Youtube [27], we consider 6 video resolutions and their corresponding bitrates, illustrated in Table 1, to compose the set of video representations ($|\mathcal{R}| = 6$).

TABLE 1: Video representations considered in this work.

Type r	Label	Video Representation $(btr(r), res(r))$
1	medium	(1.5Mbps, 360px)
2	large	(4Mbps, 480px)
3	hd720	(7.5Mbps, 720px)
4	hd1080	(12Mbps, 1080px)
5	hd1440	(24Mbps, 1440px)
6	hd2160	(53Mbps, 2160px)

For each video content $v \in \mathcal{V}$, we generate a number N_v that indicates the number of representations of v in the catalog $\mathcal{I}$, with N_v sampled from a Uniform distribution $U(1, 6)$. Next, we insert in the catalog $\mathcal{I}$ all the video items $i_{v,r}$ with content v encoded in representation r, $r = 1, 2, \ldots, N_v$. The size of the video ($size(i_{v,r})$) is a function of its duration and representation, using a normalized scale compatible with the cache storage. After combining the MovieLens dataset ($\mathcal{V}$) to the set of representations ($\mathcal{R}$), we obtain a video catalog of almost 35,000 items ($|\mathcal{I}| = 35{,}000$). We also assume that all video items in the catalog are divided into 3-second chunks. The feature vectors $\mathbf{f}^v$ and $\mathbf{f}^u$ are calculated for each $v \in \mathcal{V}$ and for each $u \in \mathcal{U}$. Then, using equations (1) and (2), we estimate P_u^{Cont} for every $u \in \mathcal{U}$.

B. UE CAPABILITY AND CQI HISTORY

For each user, we sample the UE computing capability (ψ_u) from a Uniform distribution $U(1, 6)$, ensuring that $\psi_u = res(r)$ for some video representation r described in Table 1.

The CQI values are extracted from [27] and illustrated in Table 2 (column 1). Each CQI value $q \in \mathcal{Q}$ is therefore associated with a bitrate ϕ_q, also illustrated in Table 2 (column 4). Combining the information presented in Tables 1 and 2, we can determine the best video representation for each q, illustrated in Table 2 (column 5).

To obtain the historical time series $H(u, k)$ for each user u, we use the CQI value records extracted from a real-world dataset in [30]. The CQI values were obtained every second from static and moving users using Netflix, Amazon Prime video streaming, and file download applications in such a dataset. The dataset consists of approximately 80 files corresponding to different periods when the users used the applications.

Based on this dataset, we create $H(u, k)$ for each user u, using s_q as twice the size of the largest video watched by u. This history is then used to estimate the average wireless connection quality that the user will experience during the video content playback (q_u) using Equations 14 and 15 and 60 seconds as the size of the moving average ($m = 60$). This value of m was chosen experimentally. Figure 2 presents a user's CQI variation (in seconds) during a 30-minute history interval.

Using ψ_u and q_u we calculated the best representation for u ($best(u)$) and estimate P_u^{Pref}, $\forall u \in \mathcal{U}$. Given P_u^{Rec} and P_u^{Pref}, we calculate the ultimate video request distribution P_u^{Req} and $P_u^{Req\sim}$, $\forall u \in \mathcal{U}$. In accordance with [8], we sample α_u from a Uniform distribution $U(0.5, 0.7)$. Finally, using P_u^{Cont} and $tol(u)$, we compute $\mathcal{WC}_u$ and $\mathcal{WI}_u$, which serve as input to the second step of On-RAViR.

C. EVALUATION METRIC

To evaluate the user QoE during video playback, we developed a metric inspired by MOS. In the MOS, when watching a video, the user evaluates his/her experience from a minimum value (*minMOS*) to a maximum value (*maxMOS*), ranging from 1 to 5. Based on context information such as the level of Internet congestion, denoted by *ICI* (ranging from 0% to 99%), and the representation of each video chunk reproduced, denoted by *rep*, we identified some possible situations likely to have an impact on the user QoE:

- *best_cache*: the video chunk is cached with the representation expected by the user, i.e., $best(u)$;
- *under_cache*: the video chunk is cached with a representation lower than $best(u)$;
- *best_cloud*: the video chunk is not cached, and the Internet link is not congested, allowing to retrieve $best(u)$ from the Internet;
- *under_cloud*: the video chunk is not cached, and the Internet link is congested, requiring the retrieval of a lower representation than $best(u)$ from the Internet.

When the recommended video item has chunks cached with the best representation for the user, the user QoE is the best, i.e., *maxMOS*. Otherwise, the following penalties are applied to this value:

- *PI*: the recommended video chunk is not cached; Internet access is required with *ICI* used as penalty value;
- *PR*: the recommended video chunk is cached, but its value is lower than $best(u)$.

TABLE 2: Mapping between CQI values, bitrates, and video representations.

CQI	MCS	TBS	Bitrate	Video Representation
1	0	1384	2.768	(1.5Mbps,360px)
2	0	1384	2.768	(1.5Mbps,360px)
3	2	2216	4.432	(4Mbps,480px)
4	4	3624	7.548	(7.5Mbps,720px)
5	6	5160	10.320	(7.5Mbps,720px)
6	8	6968	13.936	(12Mbps,1080px)
7	11	8760	17.520	(12Mbps,1080px)
8	13	11448	22.896	(12Mbps,1080px)
9	16	15264	30.528	(24Mbps,1440px)
10	18	16416	32.832	(24Mbps,1440px)
11	21	21384	42.768	(24Mbps,1440px)
12	23	25456	50.912	(24Mbps,1440px)
13	25	28336	56.672	(53Mbps,2160px)
14	27	31704	63.408	(53Mbps,2160px)
15	27	31704	63.408	(53Mbps,2160px)

We then use the Weber-Fechner law [24] to calculate the two penalties PR and PI. This law states that the response (R) to a stimulus is proportional to the logarithm of the stimulus intensity (E), and it is described by the expression $R = k * ln(E)$. Considering rep and ICI as stimuli, we assume the best representation (best(u)) corresponds to a stimulus intensity of 100%, with maximum QoE. An inferior representation has the stimulus intensity (E_{pr}) as a percentage of the video representation delivered (rep) in relation to the best representation for the UE, i.e., $E_{pr} = rep/best(u) * 100)$. Likewise, regarding the ICI, a congestion-free link corresponds to $ICI = 0$ and a stimulus intensity of 100%, with maximum QoE. Internet congestion ($ICI > 0$) defines the stimulus intensity (E_{pi}) as the percentage of Internet access availability, i.e., $E_{pi} = (1 - ICI) * 100)$. Other stimulus intensities are subject to PR and/or PI penalties. Thus, we formulate the user QoE as:

$$QoE = maxMOS - PR - PI, \tag{17}$$

where:

$$PR = maxMOS - (k_{pr} * ln(E_{pr})) \tag{18}$$

$$PI = maxMOS - (k_{pi} * ln(E_{pi})). \tag{19}$$

We then find the constants k_{pr} and k_{pi}. When the intensity of the stimulus is 100%, the penalties are 0, and consequently, the response to the stimulus (the QoE) is maximum ($maxMOS$). Thus, $PR = 0 \Rightarrow k_{pr} = maxMOS/ln(100)$ and $PI = 0 \Rightarrow k_{pi} = maxMOS/ln(100)$.

Figure 3 shows the impact of penalties PR and PI (individually) on the user QoE under different delivered representations (Figure 3a) and Internet congestion levels (Figure 3b). Considering $PI = 0$ and that the UE and the CQI allow the video reproduction with the highest representation, we have $best(u) = 6$. As shown in Figure 3a, in this scenario, when the system delivers a video with the expected representation (6), $PR = 0$, and the QoE is maximum. As videos with lower representations are delivered, the penalty PR increases, reducing the user QoE. Figure 3b shows a similar analysis, considering

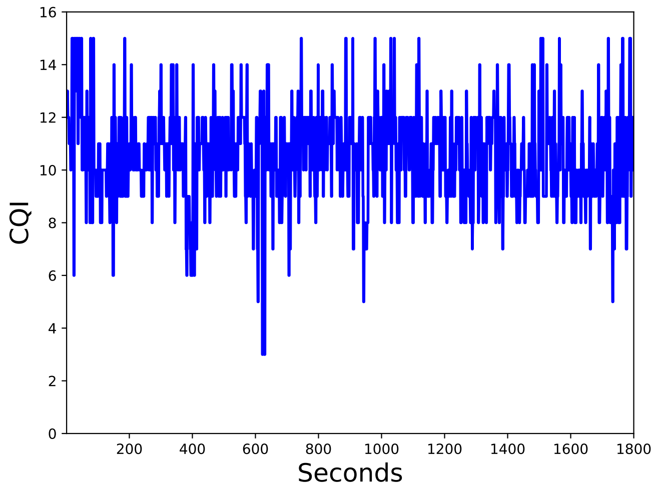


FIGURE 2: CQI variations for a given user over a period of 1800 seconds.

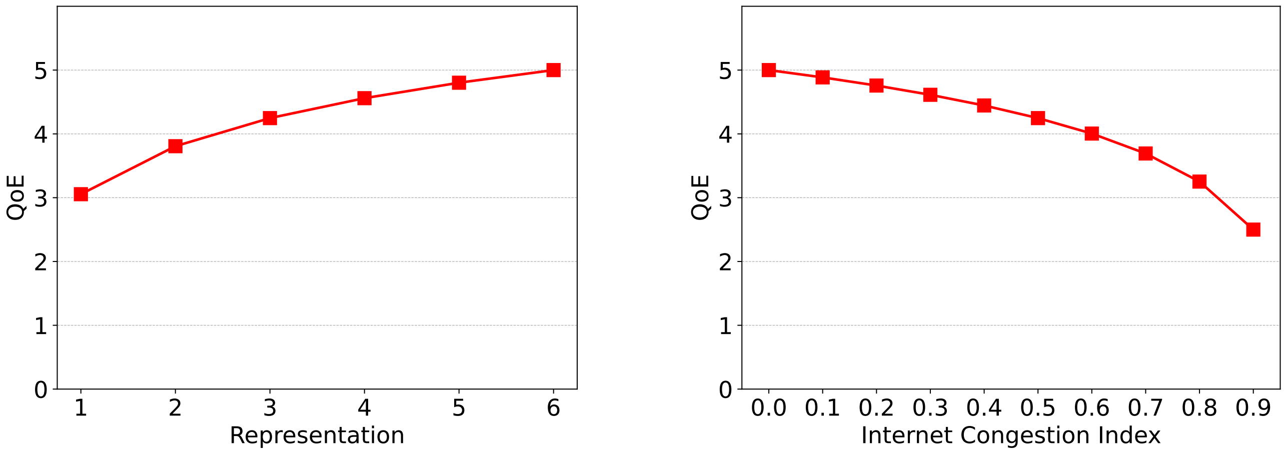
the penalty PI when $PR = 0$. As the ICI increases, the penalty PI increases, and the QoE is negatively impacted.

D. RESULTS

Initially, we compare the quality of the solutions produced by our framework (On-RAViR) against those generated by the optimal algorithm (RAViR). For this evaluation, we consider 3 recommendations for each user ($N = 3$) and an Internet congestion index of 60% ($ICI = 60\%$) with varying cache size. Figure 4 presents the results. Figure 4a shows the effective cache hit rate achieved by both algorithms, while Figure 4b presents the average user QoE provided by them. The average QoE is normalized and presented in the interval [0,1]. As shown in both figures, the heuristic consistently performs close to the optimal for all evaluated cache sizes. Regarding the execution time of the experiments, as previously mentioned, RAViR takes approximately 24 minutes for a catalog of 35,000 videos with 610 users, making 3 recommendations for each user. In contrast, On-RAViR takes a few seconds for the same scenario and the same computing environment. Thus, On-RAViR meets the response time requirements and offers good-quality solutions.

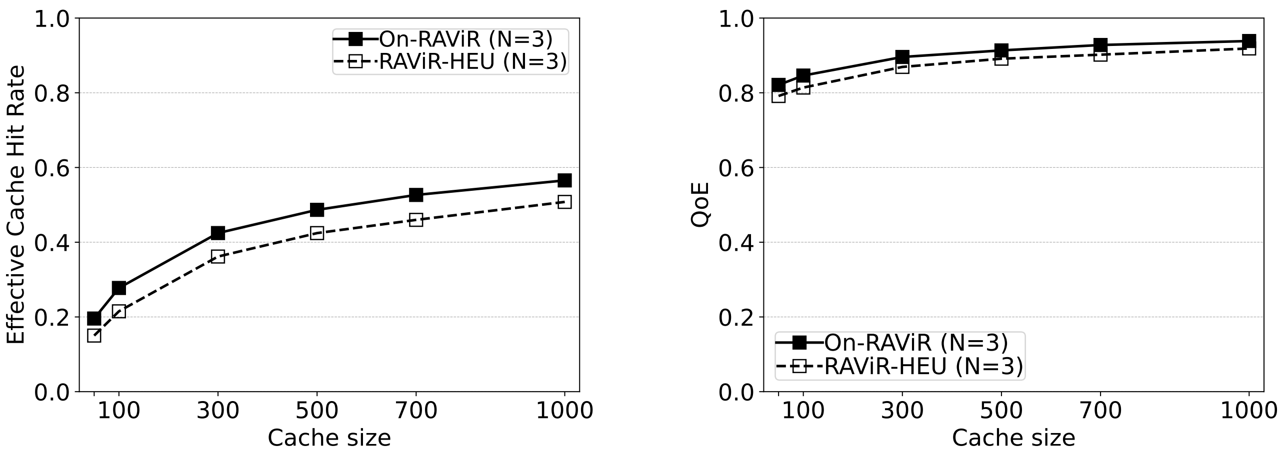
We also evaluate the performance of On-RAViR against JCRP-HEU, a heuristic we proposed based on JCRP [8]. Originally, JCRP does not account for computing and network resources, and users' preferences are based solely on content. To make a fair comparison, we design JCRP-HEU to select the representation with the lowest resolution to be included in the user $\mathcal{WI}_u$ whenever it faces a set of video items with the same content and different representations. For this evaluation, we assume that each user plays a recommended video, and we dynamically evaluate the user QoE every 3 seconds (chunk duration) during the playback. This means that every 3 seconds, we check the user's actual CQI, calculate the best representation for the user ($best(u)$), and, according to the Internet congestion index, we compute the user QoE as

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2025.3622056



(a) PR (b) PI

FIGURE 3: Impact of PR and PI penalties on the QoE metric, when the user can reproduce the video with the highest representation (6): (a) shows the impact of the PR penalty on the QoE as the video representation increases and $PI = 0$; (b) shows the impact of the PI penalty on QoE as ICI increases and $PR = 0$.



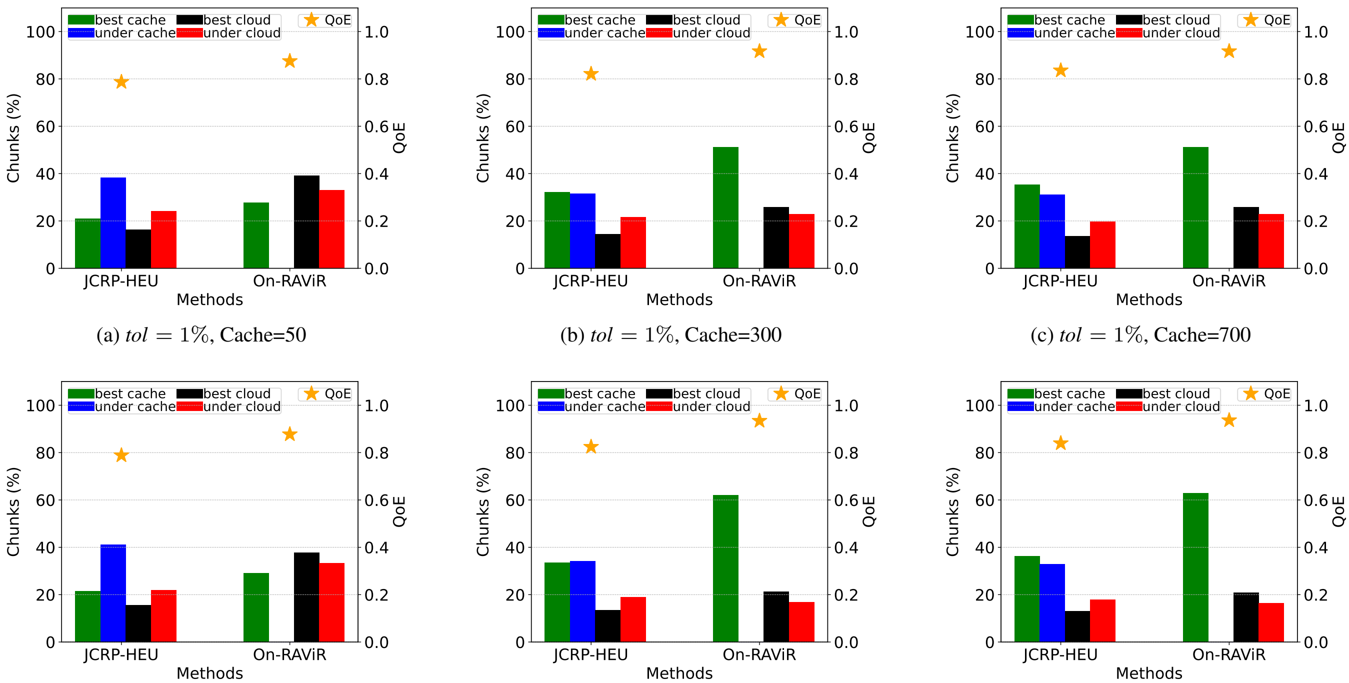
(a) Effective cache hit rate (b) User QoE

FIGURE 4: Effective cache hit rate and user QoE, achieved by RAViR and On-RAViR.

follows:

- If $best(u)$ is cached, a *best_cache* will occur and $QoE = maxMOS$;
- If $best(u)$ is not cached, the following situations are possible depending on the ICI:
  - if no better representation of the video chunk is available in the catalog, a *under_cache* will occur, the video chunk is delivered from the cache with an inferior representation, and $QoE = maxMOS - PR$;
  - If $ICI = 0$, a *best_cloud* will occur, the video chunk will be delivered using the Internet connection with the best representation, and $QoE = maxMOS$;
  - If $ICI > 0$, ICI will represent the probability of not delivering the best representation. In this case, a *best_cloud* may occur, and the video chunk will be delivered using the Internet link with the best representation, and $QoE = maxMOS - PI$. An *under_cloud* may also occur, and the video chunk will be delivered using the Internet connection with an inferior representation, and $QoE = maxMOS - PR - PI$.

Experiments were conducted in different scenarios, but due to space constraints, we present the results for the scenarios where each user receives a list of $N = 3$ recommended videos, according to distortion tolerance $tol(u) = 1\%$ and $tol(u) = 10\%$, with $ICI = 60\%$, and a total cache capacity (C) of 50, 300, and 700. Figure 5 shows the percentage of chunks transmitted during recommended video playback for each method (On-RAViR and JCRP-HEU) and the impact on the average QoE, considering the different scenarios. We observe that On-RAViR is consistently superior to JCRP-HEU, as it delivers more video chunks from the cache with the best representation for the users. JCRP-HEU can also deliver many chunks from the cache but of inferior quality. On



(a) $tol = 1\%$, Cache=50 (b) $tol = 1\%$, Cache=300 (c) $tol = 1\%$, Cache=700

(d) $tol = 10\%$, Cache=50 (e) $tol = 10\%$, Cache=300 (f) $tol = 10\%$, Cache=700

FIGURE 5: Percentage of video chunks received from cache or Internet and the average QoE of each method with 3 recommendations per user, $tol(u) = 1\%$ (a,b,c), $tol(u) = 10\%$ (d,e,f), different cache capacities, and $ICI = 60\%$.

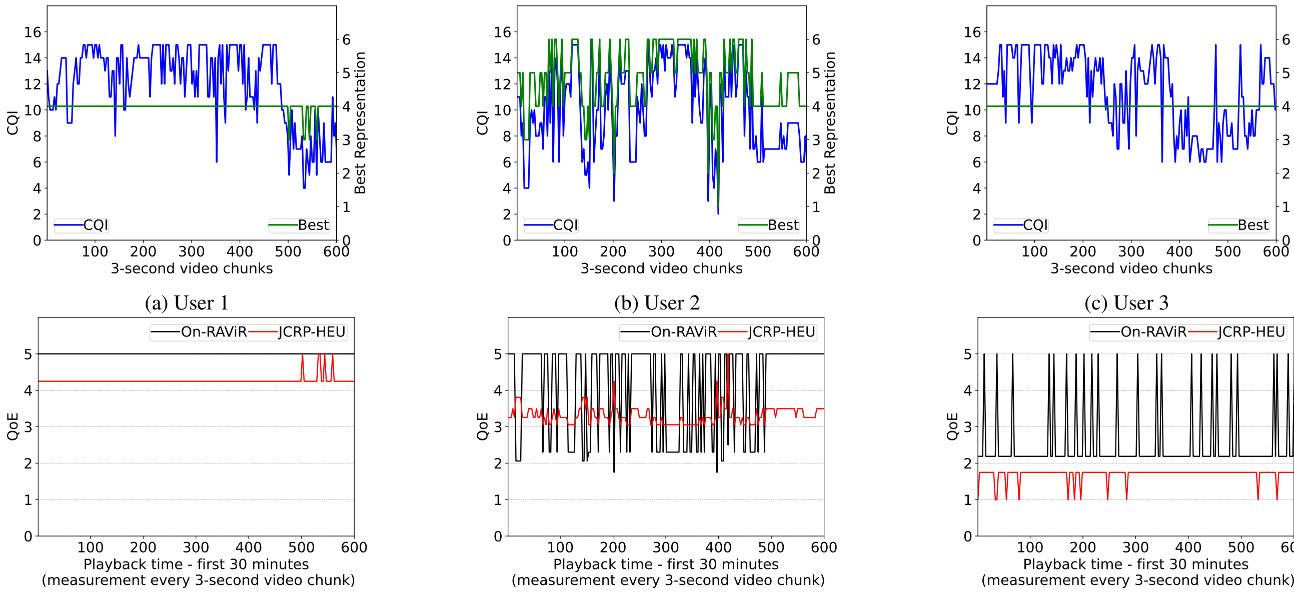
average, On-RAViR increases user QoE by 20% compared to JCRP-HEU.

We also analyze the QoE behavior individually per user during the recommended videos' playback. To this end, we randomly select two users (User 1 and User 2) and assume they choose and play a recommended video that is cached. Another user (User 3) chooses and plays a recommended video that is not cached. Figures 6a, 6b, and 6c show the variation of the CQI (blue line) and the best representation (green line) at every 3 seconds during the first 30 minutes playing a recommended video. Figures 6d, 6e and 6f show how this variation impacts the users' QoE when On-RAViR and JCRP-HEU are used under a ICI of 90%.

We can see in Figures 6d-6f that On-RAViR performs better than JCRP-HEU, despite the fluctuations. This happens because On-RAViR can predict each user's CQI, caching videos with the representations that best serve the users during the video playback period. For User 1, On-RAViR recommends and caches a video with a representation that serves the user for the entire video playback time. In contrast, JCRP-HEU recommends and caches a lower video resolution, negatively impacting the user QoE. JCRP-HEU only achieves maximum QoE for some chunks in the last few seconds, when the average CQI drops significantly. For User 2, who has a UE capable of playing a video at maximum resolution (hd2160), On-RAViR can achieve maximum QoE in many moments of the video playback, which is not the case with JCRP-HEU. It is worth noting that QoE fluctuations, very frequent, for example, for User 2, are not negative. These fluctuations will generally be imperceptible to the user, who normally uses a DASH-based client application. In the case of User 3, even when the recommended video is not cached and the Internet link is congested, On-RAViR can perform better than JCRP-HEU as it recommends videos with representation that suit the network limitation.

Finally, we analyze the impact of the cache size and ICI on the average user QoE in both methods. Figure 7a shows the results for a fixed ICI of 60% and varying cache size, considering 3 and 5 recommendations per user and a distortion tolerance of 1%. Since JCRP-HEU caches videos with lower resolutions, it can recommend and deliver video segments directly from the cache to more users. However, as Figure 7a shows, On-RAViR maintains a higher QoE than JCRP-HEU, even when the cache size is smaller (50). Figure 7b shows the results for a cache size of 50 and varying ICI, considering 3 and 5 recommendations per user and a distortion tolerance of 1%. Since the cache size is small, the QoE provided by both methods is significantly affected by the increase in the ICI value. However, even in the worst scenario (cache size=50 and ICI = 90%), On-RAViR maintains a higher QoE than JCRP-HEU.

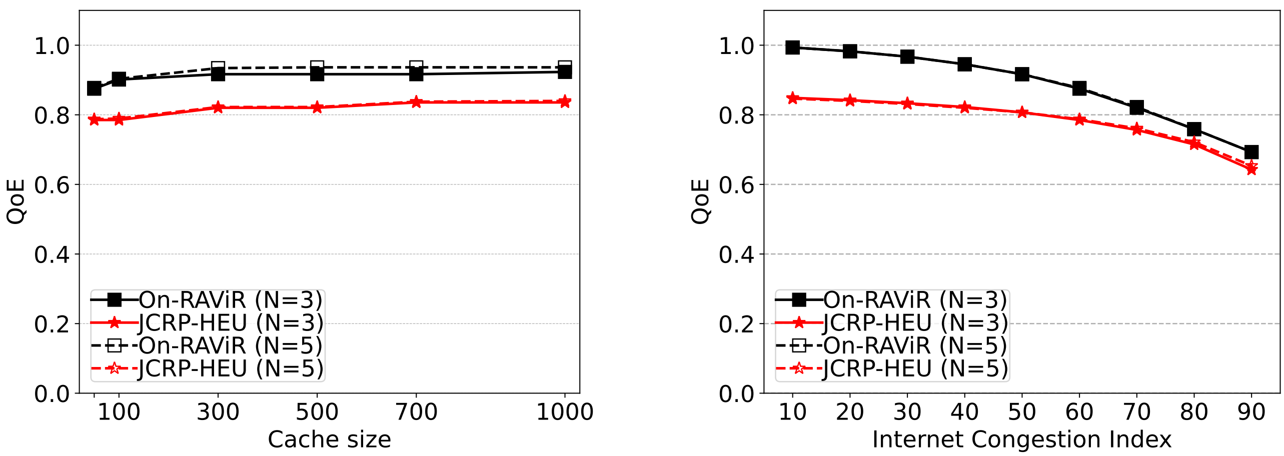
In summary, the results show that On-RAViR archives fast and high-quality solutions close to the optimal ones. Compared to JCRP-HEU, On-RAViR offers a higher average QoE, serving more requests with adequate representations. However, it is worth mentioning that although promising, our work presents some limitations. First, implementing On-RAViR in real-world deployment might face some challenges.



(a) User 1 (b) User 2 (c) User 3

(d) User 1 (e) User 2 (f) User 3

FIGURE 6: (a), (b), and (c) present the variation in the CQI of three users during the first 30 minutes of playing a recommended video; (d), (e), and (f) show the QoE of these users while playing a recommended video with an *ICI* of 90%.



(a) Cache size (b) *ICI*

FIGURE 7: Impact of total cache capacity and *ICI* on the average user QOE, considering the two methods: On-RAViR and JCRP-HEU. The solid lines represent the experiment with 3 recommendations per user ($N = 3$), and the dashed lines represent the experiment with 5 recommendations per user ($N = 5$), both with a distortion tolerance of 1% ($tol(u) = 1\%$).

In particular, the "cold start" problem may affect the efficiency of our framework as new users may not have enough information on the system to estimate their preferences and the CQI history. Additionally, integrating RNIS into existing network management and control systems or network services can be complicated. This requires close integration between network providers and content services. Second, the QoE metric proposed in this work, although providing a step towards unifying QoE, QoS, and QoR into a single metric, does not account for traditional QoR metrics used in recommendation systems such as precision and recall.

V. CONCLUSIONS AND FUTURE WORK

In this work, we presented an optimization model representing a recommendation system aware of the resources available on the user's device, the mobile network, and the edge cache combined with the caching decision. We developed an online framework to address this problem, evaluated our proposal, and compared it to a state-of-the-art heuristic. Our framework outperformed the heuristic in all evaluated scenarios, demonstrating that taking into account the device limitations, the video representation, and the network conditions when making joint decisions on video recommendation and caching offers more personalized service and leads to significant improvements in QoE. In addition, our online framework showed that it is possible to make good recommendations and cache decisions timely, even considering multi-dimension such as user preferences and computing and network availability. As future work, we intend to investigate the potential of ML techniques to deliver better solutions while considering that these techniques generally require more computing time, mainly for training. In our evaluation of ML techniques for time series forecasting, we did not observe improvement that justifies the higher computing cost compared to simple techniques such as the moving average. However, this result is related to the dataset and simple forecasting involved. We believe that ML techniques, such as deep reinforcement learning, may learn optimal policies for joint recommendation and resource allocation in the context of our problem.

ACKNOWLEDGEMENTS

This work was supported by CAPES, MCTIC/CGI.br/São Paulo Research Foundation (FAPESP) through the Project Smart 5G Core And MUltiRAn Integration (SAMURAI) under Grant 2020/05127-2 and Project Slicing Future Internet Infrastructures (SFI2) under Grant 2018/23097-3, by CNPq through the Project Universal under Grant 405111/2021-5.

REFERENCES

[1] Ericsson, "Streaming video: from megabits to gigabytes," https://www.ericsson.com/en/reports-and-papers/mobility-report/articles/streaming-video, 2020.

[2] B. Jedari, G. Premsankar, G. Illahi, M. D. Francesco, A. Mehrabi, and A. Ylä-Jääski, "Video Caching, Analytics, and Delivery at the Wireless Edge: A Survey and Future Directions," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 431–471, 2021.

[3] Y. Fu, Y. Zhang, Q. Zhu, H.-N. Dai, M. Li, and T. Q. S. Quek, "A New Vision of Wireless Edge Caching Networks (WECNs): Issues, Technologies, and Open Research Trends," *IEEE Network*, vol. 38, no. 1, pp. 247–253, 2024.

[4] F. Malandrino, C.-F. Chiasserini, G. Avino, M. Malinverno, and S. Kirkpatrick, "From Megabits to CPU Ticks: Enriching a Demand Trace in the Age of MEC," *IEEE Transactions on Big Data*, vol. 6, no. 1, pp. 43–50, 2020.

[5] T. V. Doan, V. Bajpai, and S. Crawford, "A Longitudinal View of Netflix: Content Delivery over IPv6 and Content Cache Deployments," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020, pp. 1073–1082.

[6] C. A. Gomez-Uribe and N. Hunt, "The Netflix Recommender System: Algorithms, Business Value, and Innovation," *ACM Trans. Manage. Inf. Syst.*, vol. 6, no. 4, Dec. 2016.

[7] P. Sermpezis, T. Spyropoulos, L. Vigneri, and T. Giannakas, "Femto-Caching with Soft Cache Hits: Improving Performance with Related Content Recommendation," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–7.

[8] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Jointly Optimizing Content Caching and Recommendations in Small Cell Networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 1, pp. 125–138, 2019.

[9] S. Kastanakis *et al.*, "Network-Aware Recommendations in the Wild: Methodology, Realistic Evaluations, Experiments," *IEEE Transactions on Mobile Computing*, vol. 21, no. 7, pp. 2466–2479, 2022.

[10] A. C. B. L. Moncao *et al.*, "Combining Resource-Aware Recommendation and Caching in the Era of MEC for Improving the Experience of Video Streaming Users," *IEEE Transactions on Services Computing*, vol. 16, no. 3, pp. 1698–1712, 2023.

[11] M.-C. Lee and Y.-W. P. Hong, "Socially-Aware Joint Recommendation and Caching Policy Design in Wireless D2D Networks," in *ICC 2021 - IEEE International Conference on Communications*, 2021, pp. 1–6.

[12] Q. Ning, M. Yang, C. Tang, and L. Huang, "Optimizing Network Performance Through Joint Caching and Recommendation Policy for Continuous User Request Behavior," *IEEE Transactions on Network and Service Management*, vol. 21, no. 2, pp. 1490–1502, 2024.

[13] D. Yu, T. Wu, C. Liu, and D. Wang, "Joint Content Caching and Recommendation in Opportunistic Mobile Networks Through Deep Reinforcement Learning and Broad Learning," *IEEE Transactions on Services Computing*, vol. 16, no. 4, pp. 2727–2741, 2023.

[14] D. Liu and C. Yang, "A Learning-Based Approach to Joint Content Caching and Recommendation at Base Stations," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–7.

[15] M. S. Nogueira, J. I. D. Pinheiro, F. Assis, D. S. Menasché, P. Sermpezis, and T. Spyropoulos, "When Should Recommenders Account for Low QoS?" *IEEE Access*, vol. 11, pp. 132 014–132 036, 2023.

[16] X. Zhao, F. Hou, Y. Kuang, and P. Yuan, "Cache Strategy for Joint Content Recommendation and D2D Collaboration," in *2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2024, pp. 915–920.

[17] Y. Fu, L. Salaün, X. Yang, W. Wen, and T. Q. S. Quek, "Caching Efficiency Maximization for Device-to-Device Communication Networks: A Recommend to Cache Approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 10, pp. 6580–6594, 2021.

[18] M. Nogueira *et al.*, "Can Recommenders Compensate for Low QoS?" in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 4185–4190.

[19] Y. Fu, Y. Zhang, A. K. Y. Wong, and T. Q. S. Quek, "Revenue Maximization: The Interplay Between Personalized Bundle Recommendation and Wireless Content Caching," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 4253–4265, 2023.

[20] M. Plakia, E. Tzamousis, T. Asvestopoulou, G. Pantermakis, N. Filippakis, H. Schulzrinne, Y. Kane-Esrig, and M. Papadopouli, "Should I Stay or Should I Go: Analysis of the Impact of Application QoS on User Engagement in YouTube," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 5, no. 2, Apr. 2020.

[21] J. Nightingale, P. Salva-Garcia, J. M. A. Calero, and Q. Wang, "5G-QoE: QoE Modelling for Ultra-HD Video Streaming in 5G Networks," *IEEE Transactions on Broadcasting*, vol. 64, no. 2, pp. 621–634, 2018.

[22] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, "Toward A Practical Perceptual Video Quality Metric," https://netflixtechblog.com/toward-a-practical-perceptual-video-quality-metric-653f208b9652, 2016, accessed: March-10-2021.

[23] Y. Ben Mazziane, S. Alouf, G. Neglia, and D. S. Menasche, "Ttl model for an lru-based similarity caching policy," *Comput. Netw.*, vol. 241, no. C, 2024.

[24] H. R. Schiffman, *Sensation and Perception: An Integrated Approach, 5th Ed.* New Jersey: John Wiley & Sons, 2000.

[25] European Telecommunications Standards Institute (ETSI), "Multi-access Edge Computing (MEC); Radio Network Information API," 2019, eTSI GS MEC 012 V2.1.1.

[26] K. Saija, S. Nethi, S. Chaudhuri, and R. M. Karthik, "A Machine Learning Approach for SNR Prediction in 5G Systems," in *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2019, pp. 1–6.

[27] S.-R. Yang, Y.-J. Tseng, C.-C. Huang, and W.-C. Lin, "Multi-Access Edge Computing Enhanced Video Streaming: Proof-of-Concept Implementation and Prediction/QoE Models," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1888–1902, 2019.

[28] V. Cacchiani, M. Iori, A. Locatelli, and S. Martello, "Knapsack problems - An overview of recent advances. Part I: Single knapsack problems," *Computers & Operations Research*, vol. 143, pp. 1–14, 2022.

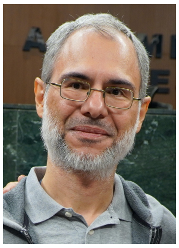
[29] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, 2015.

[30] D. Raca *et al.*, "Beyond Throughput, the next Generation: A 5G Dataset with Channel and Context Metrics," in *Proceedings of the 11th ACM Multimedia Systems Conference*, ser. MMSys '20, 2020, pp. 303–308.

ANA CLAUDIA B. L. MONÇÃO is an assistant professor at the Institute of Informatics - Universidade Federal de Goiás (UFG) since 2013. She holds a degree in Data Processing Technologist from Pontifical Catholic University of Rio de Janeiro (1992), has MSc in Computer Sciences from Universidade Federal de Goiás (2013) and PhD (2023) in Universidade Federal de Goiás. Her main research interests include recommender systems and data science.

KARLLA B. CHAVES RODRIGUES received the degree in Information Systems (2022) and a M.Sc. degree in Computer Science (2025), both from the Universidade Federal Goiás (UFG), where she is currently a Ph.D. student. She has contributed to international research initiatives through BR-EU collaboration calls and participated in national projects supported by the Brazilian government. Her research interest include cloud and edge computing, software-defined networking , performance evaluation of distributed systems and characterization of computational resources.

SAND LUZ CORRÊA received the degree in computer science from Universidade Federal de Goiás (UFG), in 1994, the M.Sc. degree in computer science from the University of Campinas (UNICAMP), in 1997, and the Ph.D. degree in informatics from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio), in 2011. In 2015, she spent her sabbatical at Virginia Tech, USA, and in 2020, at Inria Saclay Research Centre, France. She is an Associate Professor at the Institute of Informatics, UFG, where she has been a Professor and a Researcher, since 2010. She has participated in some international research projects (including two from joint calls BR-EU). Her research interests include cloud computing, SDN, data plane programmability, sensor systems, and smart city platforms.



KLEBER VIEIRA CARDOSO received the degree in computer science from Universidade Federal de Goiás (UFG), in 1997, and the M.Sc. and Ph.D. degrees in electrical engineering from COPPE—Universidade Federal do Rio de Janeiro, in 2002 and 2009, respectively. In 2015, he spent his sabbatical with Virginia Tech, USA, and in 2020, with the Inria Saclay Research Centre, France. He has participated in some international research projects (including two from joint calls BR-EU) and coordinated several national-sponsored research and development projects. He is an Associate Professor with the Institute of Informatics—UFG, where he has been a Professor and a Researcher, since 2009. His research is focused on the following topics: wireless networks, software-defined networks, virtualization, resource allocation, and performance evaluation.