Joint User Association, Content Caching and Recommendations in Wireless Edge Networks

L. E. Chatzieleftheriou, G. Darzanos, M. Karaliopoulos, I. Koutsopoulos*
Athens University of Economics and Business, Athens, Greece
{liviachatzi, ntarzanos, mkaralio, jordan}@aueb.gr

ABSTRACT

In this paper, we investigate the performance gains that are achievable when jointly controlling (*i*) in which Small-cell Base Stations (SBSs) mobile users are associated to, (*ii*) which content items are stored at SBS co-located caches and (*iii*) which content items are recommended to the mobile users who are associated to different SBSs. We first establish a framework for the joint user association, content caching and recommendations problem, by specifying a set of necessary conditions for all three component functions of the system. Then, we provide a concrete formulation of the joint problem when the objective is to maximize the total hit ratio over all caches. We analyze the problems that emerge as special cases of the joint problem, when one of the three functions is carried out independently, and use them to characterize its complexity. Finally, we propose a heuristic that tackles the joint problem. Proof-of-concept simulations demonstrate that even this simple heuristic outperforms an optimal algorithm that takes only caching and recommendation decisions into account and provide evidence of the achievable performance gains when decisions over all three functions are jointly optimized.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Network Operations; F.2 [**Analysis of Algorithms and Problem Complexity**]: General

Keywords

User association, recommender systems, edge caching, QoS

1. INTRODUCTION

Video traffic becomes more and more important for the current Internet, while the consumption of high quality multimedia content (*e.g.*, HD videos) is turning to a daily habit for the average mobile user. These trends increase the pressure on the edge of the network where the demand of multiple users is aggregated. According to [4], the monthly mobile data traffic is increasing with a CAGR (Compound Annual Growth Rate) of 47% and will reach the 49 exabytes by 2021. The need to accommodate this exponential growth of traffic and scale-up the capacity of mobile networks motivates ultra-dense architectures drawing on Small-cell Base Stations (SBSs). In turn, this densification of radio access points increases the capacity needs at the mobile backhaul.

To cope with the backhaul capacity limitations, especially during peak hours of content demand, the wireless networking community has been looking into the deployment of caches at the SBSs (*e.g.*, [12]). The idea is simple: by dynamically predicting the users' demand and storing content accordingly at SBS co-located caches, we could serve the demand locally and alleviate the traffic load at the backhaul.

On the other hand, according to [6,8], a significant portion of content downloads is the result of recommender systems. These are typically deployed by Content Provider platforms that interact with users through mobile apps. This demand-shaping capability of recommender systems renders them a powerful mechanism for implicitly optimizing the effectiveness of caching mechanisms and the network performance while taking into account the users' QoS requirements. Multiple studies in the recent literature (*e.g.*, [3, 7, 8, 10, 13]) explore models and algorithms in this direction. Common to them is the joint treatment of the content caching and recommendation problems, with the aim to improve the effectiveness (*i.e.*, cache hit ratio) of content caching at the network edge. We henceforth refer to this research thread as the Joint Caching and Recommendation (JCR) approach.

The main motivation for our work has been the remark that the content demand emerging locally within a small cell is highly dependent on the users who access the network through that cell and their content preferences. Therefore, the *user association* process, controlling how mobile users are associated with SBSs, could be viewed as another demand-shaping mechanism.

Most of the studies following the JCR approach assume that the underlying user association decisions do not take into account the content demand. This is in line with current user association algorithms that typically associate a device to the cell providing it with the best radio quality. However, the possibility to also *control* the association of mobile users to SBSs, when there are more than one alternatives for it, adds more degrees of freedom and optimization potential to the JCR approach. Besides which content to *store* and which to *recommend*, we may question how could mobile users be best associated to SBSs. We refer to this approach as the Joint Caching, Recommendation and As-

*This research has been co-financed by the Operational Program "Human Resources Development, Education and Lifelong Learning" and is co-financed by the European Union (European Social Fund) and Greek national funds. The authors wish to thank Prof. Daniel Sadoc Menasché for useful remarks and suggestions about the paper.

IFIP WG 7.3 Performance 2018. Dec. 5–7, 2018, Toulouse, France

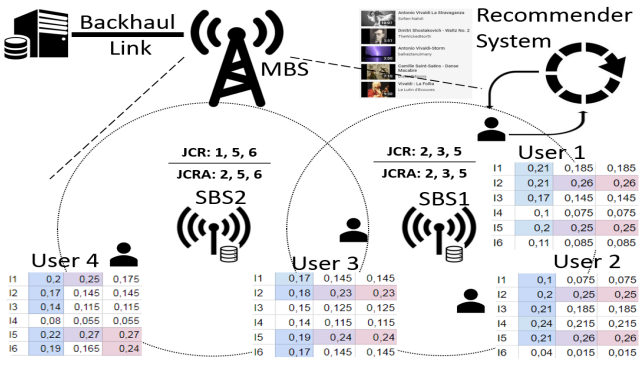

Figure 1: Illustration of the toy example. Caches are co-located with MBS/SBSs and serve users with content of their preference. Users can be associated only to one SBS in their range or to the MBS. A recommendation system issues content recommendations to the users in a centralized manner.

sociation (JCRA) approach and employ a toy example to motivate it.

Toy Example. Figure 1 shows a scenario with two SBSs, one Macro-cell Base Station (MBS) with a coverage area that is a superset of the two small cells, and four users. A user can be associated to only one of the SBSs within her range, or to the MBS, if none of the SBSs can serve her at the required quality. In our example, users 1 and 2 are in the range of SBS1 only, user 4 is in the range of SBS2 only, while user 3 is within range of both and closest to SBS1. We consider a content catalogue of six items. Users who are associated to the MBS can access any of those items through the cache of a back-end server. On the other hand, caches that can store up to three items are co-located with the two small cells. Users generate content requests that are based on their interests and are served by the associated SBS as far as the requested content is stored there. A recommendation system issues recommendations to all four users for two items that belong to their personal interests. For the sake of simplicity, we assume here a naive rule for the impact of the recommendations, namely that recommending an item to a user boosts the user's probability of requesting it by 0.05 and not doing so decreases it by 0.025.

Figure 1 includes four user tables and two SBSs tables. Each SBS table shows the placement of content items in a specific SBS cache under both the JCR and JCRA approaches. On the other hand, each user table has four columns. The second column reports the inherent content preference distribution of the user over the six items (I1-I6) presented in the 1st column. The third column presents the modified content preference distribution when we apply the JCR approach (the two recommended items are shaded). Finally, the fourth column illustrates the impact of the JCRA approach (recommended items are shaded).

Under the JCR approach, we assume that each user is associated to the closest SBS. Hence, users 1, 2 and 3 are served by SBS1 so that items I2, I3, I5 are stored in the SBS1 cache, items I1, I5, I6 are stored in the SBS2 cache, and the aggregate cache hit ratio for both SBSs is 0.65. On the other hand, under the JCRA approach, users 1 and 2 are associated with SBS1 and users 3 and 4 to SBS2. SBS1 stores items I2, I3, I5, whereas SBS2 stores I2, I5, I6, with recommendations being directed to different items. Consequently, the total cache hit ratio grows from 0.62 for JCR to 0.80 for JCRA.

1.1 Related Work

There is a growing interest in using recommender systems as demand shaping tools for network-related goals. Works [5, 7] consider generic network settings, while [2, 3, 13] consider wireless environments with wireless SBSs and static [2, 3] or mobile [13] users. However, neither of these studies considers association or routing decisions. Authors in [10] consider P2P systems and account for the dissemination cost of content. In [2, 3, 5, 7, 8], the authors explicitly recommend content to users to shape demand distributions, while in [13] they consider scenarios where alternative content is delivered or recommended to users *after* a request for non-cached content. The impact of recommendations is experimentally described in [8] and mathematically modeled in [2, 3]. These works consider recommendations as a demand-shaping tool that can be engineered under constraints as to how much they distort personal preferences. In [5], the authors draw on the PageRank model to provide sequential recommendations to users under given cache placement. The application in [7] retrieves YouTube videos that are thematically relevant to requested content and, at the same time, cached at the server, and recommends them to users. Their results suggest that a "recommendation window", such as that in [3], could yield significant cache hit ratio improvements.

Despite the similar network setting, our work significantly differs from the studies described above. In particular, these studies focus on optimizing content placement by taking joint decisions either on (i) content caching and recommendation, *or* (ii) by simply optimizing recommendations for given cache placement. Our work takes a first step in exploring the resulting performance gains when *all three decisions about content caching, content recommendation, and user association are jointly controlled.*

2. SYSTEM MODEL

Our model considers SBSs and an MBS that work in conjunction forming a two-layer heterogeneous network. Each SBS can serve a set of mobile users within its range, while the MBS can serve the users within range of any SBS. The mobile users generate content item requests by accessing a Content Provider platform (*e.g.*, Netflix). We assume that all SBSs are equipped with caches that can store content items provisioned by the Content Provider platform. A user can be either associated to *one* SBS or the MBS. The association of a user to an SBS is the preferred alternative since it can exploit the capacity and caching capability of the SBSs more efficiently, preserving the radio resources of the macro cell. Nevertheless, a user can always associate with the MBS, as a fallback solution, when the association to an SBS is not feasible. A centralized recommender system issues recommendations for content items to the users. Its aim is to shape the content demand across the different SBSs and boost the effectiveness of content placement decisions. We assume that user locations fall within the range of at least one SBS, and that user demands are stationary stochastic processes in the time intervals between two decision epochs.

Content demand and placement. We consider that a content catalog $\mathcal{I}$ is made available to mobile users $\mathcal{U}$. The items of the catalog are classified into one or more thematic categories and have different sizes $L_i, i \in \mathcal{I}$ in bytes. The content items have different sizes, ranging from large entire movie files to small advertisements. We use L_i to denote the size of item i in bytes.

Moreover we make the following assumption:

Assumption 2.1 (Resource Limitation) *Each SBS has limited service capacity and each SBS co-located cache has finite storage capacity.*

Replicas of each content item $i \in \mathcal{I}$ may be stored in any set of the SBS caches, besides the MBS cache, depending on the actual caching decisions. We denote as S_c the storage capacity of each SBS $c \in \mathcal{C}$ and measure it in bytes. At any point in time, each cache c stores a finite set of files, referred to as the cache placement $\mathcal{P}_c$. On the other hand, we assume that the back-end server cache M has enough capacity to store copies of the entire catalog.

Each user is characterized by a preference distribution that expresses her inherent preferences for each content item. This distribution can be extracted by taking into account the user preferences and the thematic categories each item is related to. In our work, we assume that the system is aware of this distribution, *i.e.*, each user u is described by a content preference distribution, $p_u(i),\ i \in \mathcal{I}$, with $\sum_{i \in \mathcal{I}} p_u(i) = 1$, which captures her original preferences over all items.

Recommender System. We consider a time slotted system and assume that content demand predictions are made once every time slot. User inherent preferences change slowly within a time slot but they are significantly affected by the recommendations.

Assumption 2.2 (Impact of Recommendations) *If we assume that item i adequately matches user's u preferences, the recommendation of item i to user u boosts its request probability.*

In our model especially, we assume that the content request distribution is a convex combination of the distributions p_u, which reflects the *inherent preferences* of the users, and r_u, which reflects the *impact of the recommender system*. In particular, the content request distribution is given by

$$d_u(i) = w_u \cdot r_u(i) + (1 - w_u) \cdot p_u(i) \tag{1}$$

for each of the R items that are recommended to user u, and by

$$\widetilde{d}_u(i) = (1 - w_u) \cdot p_u(i) \tag{2}$$

for each one of the $(|\mathcal{I}| - R)$ items not recommended to user u. The recommendation weights w_u in (1) and (2) express the importance user u attaches to recommendations and in practice can be found from historical data, *e.g.*, as in [6, 8].

Although the issued recommendations are normally expected to follow exactly the users' preferences, we allow them to deviate from exactly matching the preference distribution. Hence, the recommended items may not be the top R items in the user preferences since the recommender system may also take into account the caching decisions and network conditions in determining the items to recommend. Nevertheless, the issued recommendations should not deviate too much from the user preference distribution, otherwise the system will cause dissatisfaction to the users. In order to address the user dissatisfaction concerns, we introduce the term "Quality of Recommendations" (QoR) and make the following assumption:

Assumption 2.3 (QoR guarantee) *The personalized recommendations that are issued by the system to users must satisfy certain quality guarantees, i.e., not deviate from the original user preferences beyond a well-specified quality threshold.*

One example of implementing the QoR guarantee is the model in [2]. The two relevant model provisions are:

- the introduction of the *recommendation window* $\mathcal{W}_u$, setting a bound on the preference value for items that could be recommended instead of the nominally recommended R most preferred items;
- the preservation of the ranking order by recommended items, *i.e.*, the provision that the order of items in the recommendation list follows the order of items in the inherent content preference distribution of the user.

An extended discussion about both the recommendation window and the trade-offs that its size implies is presented in [2].

User-Cache Association. At any given time slot, each user $u \in \mathcal{U}$ is located within the range of a different subset of SBSs. We define as $\mathcal{N}(u)$ the "neighborhood" of user u, *i.e.*, the set of cells user u can be associated with. However, a user can only be associated to one SBS or the MBS, hence having access to either one of the SBS co-located caches or the larger server with the whole content catalog that is accessible through the MBS. Each SBS has limited service rate. We assume that this rate is split among all user devices associated with it so that their QoS requirements can be satisfied.

Assumption 2.4 (QoS guarantee) *The association of a user to a certain SBS should guarantee the minimum QoS that the user has acquired from the Content Provider platform.*

In our approach, we assume a minimum downlink data rate should be guaranteed to each user, based on acquired product (*e.g.*, 720p, 1080p *etc.*). Consequently, the number of platform users that can be associated to an SBS is limited since a congested SBS would not be able to satisfy the specified QoS requirements. The association of user $u \in \mathcal{U}$ to a certain SBS $c \in \mathcal{C}$, at a guaranteed service rate, generates *association cost* b_{cu} for the SBS. The number of users that an SBS can serve is determined by the aggregate *association cost*, which should not be higher than B_c, for all cache-enabled SBSs $c \in \mathcal{C}$.

The association cost is strongly related to the physical distance of the user to the associated small cell. In fact, a more distant user generates higher association cost in terms of transmit power for the SBS, network interference or even power consumption (for her own device). Then, b_{cu} corresponds, for instance, to the power that SBS c assigns to the device of user u for achieving a downlink data rate that satisfies the QoS requirements of that user.

3. PROBLEM FORMULATION AND ANALYSIS

3.1 Problem formulation

The objective of the joint caching, recommendation, and association (JCRA) problem is to maximize the portion of total demand that can be satisfied by all SBS caches. This can be measured through the aggregate cache hit ratio among all SBS caches

$$H = \frac{\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{P}_{c_u}} x_{ui}d_u(i) + (1-x_{ui})\widetilde{d}_u(i)}{\sum_{u\in\mathcal{U}}\sum_{i\in\mathcal{I}} x_{ui}d_u(i) + (1-x_{ui})\widetilde{d}_u(i)}, \quad (3)$$

where $\{x_{ui}\}$ is a set of binary decision variables, with $x_{ui} = 1$ if item i is recommended to user u and $x_{ui} = 0$ otherwise. c_u is the SBS co-located cache user u is associated with and $\mathcal{P}_{c_u}$ denotes the placement of items in SBS co-located cache c. As we highlighted also in section 2, maximizing H is beneficial for both the network performance and the users' QoS.

Let $\{y_{ic}\}$ and $\{z_{cu}\}$ be two sets of binary decision variables, with $y_{ic} = 1$ if item i is cached in cache c and $y_{ic} = 0$ otherwise; $z_{cu} = 1$ if user u is associated to BS c and $z_{cu} = 0$ otherwise. Then, our objective is to solve the following maximization problem:

$$\max_{\mathbf{y},\mathbf{x},\mathbf{z}} \quad \sum_{u\in\mathcal{U}}\sum_{c\in\mathcal{C}}\sum_{i\in\mathcal{I}} y_{ic}z_{cu}(x_{ui}d_u(i) + (1-x_{ui})\widetilde{d}_u(i)) \quad (4)$$

$$s.t. \quad \sum_{i\in\mathcal{I}} y_{ic}L_i \le S_c, \quad \forall c \in \mathcal{C} \quad (5)$$

$$\sum_{u\in\mathcal{U}} b_{cu}z_{cu} \le B_c, \quad \forall c \in \mathcal{C} \quad (6)$$

$$\sum_{c\in\mathcal{N}(u)\cup M} z_{cu} = 1, \quad \forall u \in \mathcal{U}, \quad (7)$$

$$z_{cu} = 0, \quad u \in \mathcal{U}, c \in \mathcal{C} \setminus \mathcal{N}(u) \quad (8)$$

$$x_{ui} = 0, \quad \forall i \notin \mathcal{W}_u, u \in \mathcal{U}, \quad (9)$$

$$\sum_{i\in\mathcal{I}} x_{ui} = R, \quad \forall u \in \mathcal{U} \quad (10)$$

$$y_{ic}, x_{ui}, z_{cu} \in \{0,1\}, u \in \mathcal{U}, i \in \mathcal{I}, c \in \mathcal{C}, \quad (11)$$

Constraints (5) and (6) reflect the cache storage and service cost constraints for each cache, respectively. Constraint (7) captures the fact that users will be either associated with an SBS in their neighbourhood or to the MBS and constraint (8) indicates that users can receive content only from SBSs in their neighbourhood. Constraints (9) and (10) ensure that the content items recommended to each user are always within her recommendation window and that exactly R items will be recommended to her, respectively. Finally, constraint (11) denotes the binary nature of the decision variables y_{ic}, x_{ui} and z_{cu}.

3.2 Special Cases and Complexity Analysis

As we already illustrated through our toy example, the JCRA approach exhibits greater optimization potential than JCR or any other approach that does not involve joint decisions over all three caching, recommendation and association. However, due to the intractability of the JCRA problem, we approach and prove the hardness of the JCRA problem through three special cases.

Table 1: Notation table

Notation	Context
$\mathcal{I},\mathcal{U},\mathcal{C}$	Items, Users, SBS co-located caches
M	Back-end server cache
p_u	Inherent content preference distribution of user u
r_u	Probability distribution due to recommendation
d_u	Content item request probability distribution of user u (recommended items)
$\widetilde{d}_u$	Content item request probability distribution of user u (non-recommended items)
R	Number of recommended items
$\mathcal{W}_u$	Recommendation window of user u
$N(u)$	Set of caches that user u can be associated to
b_{cu}	Cost of associating user u to cache c
B_c	Maximum aggregate association cost that cache c can handle
S_c	Storage capacity of cache c in bytes
$\mathcal{P}_c$	Items placement in cache c
H	Aggregate cache hit ratio

***i)* Fixed user associations:** We assume that a Content Provider has control over the placement of content items to the caches co-located with the SBSs and the recommendations issued by its platform. On the other hand, the association of users to the SBSs and MBS is performed independently by a Mobile Network Operator that associates the users in content-agnostic manner.

PROPOSITION 3.1 *Under fixed user association decisions, the JCRA Problem is NP-Complete.*

PROOF. When we fix the variables z_{cu}, constraints (6) and (8) of the JCRA Problem trivially hold and the JCRA problem reduces to the JCR problem, which is shown to be NP-Complete [2]. □

***ii)* Fixed content recommendations:** The underlying assumption is that the Content Provider does not nudge users towards requesting cached content but rather issues recommendations that match the users' inherent content preferences.

PROPOSITION 3.2 *Under fixed content recommendation decisions, the JCRA Problem is NP-Hard.*

PROOF. We can prove the JCRA Problem NP-hardness by generalization. When we fix the content item recommendations, *i.e.*, the values of the variables x_{ui}, it is $x_{ui} = 1, \forall i \in \mathcal{W}_u, \forall u \in \mathcal{U}$ and $x_{ui} = 0, \forall i \notin \mathcal{W}_u, \forall u \in \mathcal{U}$. the constraints (9) and (10) of the JCRA Problem trivially hold. The JCRA Problem reduces to a problem that is itself a generalization of the Generalized Assignment Problem (GAP). Fixing the caching decision variables y_{ic}, we can map the users and SBSs in our case to the jobs and agents, respectively, in GAP. Since GAP is an NP-Hard combinatorial optimization problem [1], the special case of the JCRA Problem where recommendations are fixed is NP-Hard too. □

***iii)* Fixed cache placements:** When we fix the content placement at the SBS co-located caches, the system only issues recommendations and associates users to SBSs. This scenario becomes relevant when the content is cached according to prediction of demand over longer time scales but

shorter-term demand variations (*e.g.*, due to users' high mobility) are addressed by the joint content recommendation and user association decisions.

In this scenario, constraints (5) of the JCRA problem trivially hold. Moreover, considering the case that the recommender system issues recommendations for the items that rank top in the user's content preferences, for every SBS cache that is association possibility of user u, we can compute the part of her content demand that can be served by it. To show that the JCRA Problem under fixed caching decisions is NP-Hard, we consider the corresponding decision problem of this special case.

P1: *Association Decision Problem.* We use $\mathcal{C}$ to denote the set of SBSs and $\mathcal{U}$ the set of users in the system. Under fixed caching and recommendation decisions, each user has to be assigned to only one SBS. We consider the set $\mathcal{D} = \{k_{u,c}(i)\}_{u,c,i}$ that captures the final demand distribution of all users u for all items i when associated to every SBS c, including the impact of recommendations. In other words, each element of $\mathcal{D}$ captures the probability for an item i to be requested by user u when associated to SBS c. Let $\mathcal{B}$ be the set whose elements denote the aggregate association cost each SBS can handle and b be the set whose elements denote the association costs for each pair (u, c) of user u and SBS c. Consider also a real number $Q \geq 0$. Then, the following question arises:

" Is there any assignment of users to SBSs such that the association cost constraints are satisfied for all caches and the aggregate cache hit ratio over all SBSs is higher than Q? In other words, is there a set of values for all z_{cu} such that

$$\sum_{u\in\mathcal{U}} z_{cu} b_{cu} \leq B_c, \forall c \in \mathcal{C}, \text{ and } \sum_{u\in\mathcal{U}}\sum_{c\in\mathcal{C}} z_{cu} \sum_{i\in\mathcal{P}_c} k_{u,c}(i) > Q?"$$

Denoting the problem instance by **P1**($\mathcal{C}$, $\mathcal{U}$, $\mathcal{I}$, $\mathcal{D}$, $\mathcal{B}$, b, Q), we can easily see that **P1** is in NP.

To show NP-Hardness, we take advantage of the Generalized Assignment Problem (GAP) and perform a polynomial time reduction of it to **P1**.

PROPOSITION 3.3 $GAP \leq_L$ **P1**.

PROOF. Since the cache placement is fixed for every SBS, the value v_{uc} for associating user u to SBS c is the aggregate cache hit ratio over all cached items, *i.e.*, $v_{uc} = \sum_{i\in\mathcal{P}_c}(x_{ui} d_u(i) + (1 - x_{ui})\widetilde{d}_u(i))$, where $\mathcal{P}_c$ the cache placement of SBS c. Moreover, each user u has a different association cost b_{cu} for being associated to SBS c. Finding the association of users to SBSs under fixed caching decisions and truthful recommendations leads to the solution of the Generalized Assignment Problem. In particular, we map users $\mathcal{U}$ and SBS co-located caches $\mathcal{C}$ of the P1 to the objects and machines of GAP respectively. Accordingly, we use v_{uc} and cost b_{cu} for denoting the profit and cost for assigning user u to SBS c. □

COROLLARY 3.1 *Under fixed caching decisions, the JCRA Problem is NP-Hard.*

***iv)* Complexity of the JCRA problem:** From the analysis in *(i)-(iii)* we can easily conclude that:

COROLLARY 3.2 *The Joint Caching, Recommendation and Routing Problem is NP-Hard.*

4. OUR HEURISTIC

In this section we present a heuristic scheme that simultaneously takes decisions for all three types of decision variables in the JCRA Problem. The heuristic proceeds in the three steps that are listed and described in what follows.

Step 1: Associate users to small cells accounting for their inherent content preferences.

(i) Initially, for each small cell cache c we compute the maximum potential demand that can emerge for each item $i \in \mathcal{I}$ as:

$$\hat{p}_c(i) = \sum_{u:c\in\mathcal{N}(u)} p_u(i). \tag{12}$$

This formula takes into consideration all users that could be potentially be served by SBS c. Note that the inherent content preferences of user u are factored in all SBSs in her neighbourhood $\mathcal{N}(u)$.

(ii) Once these upper bounds for the aggregate demand at each SBS c are determined for all content items, we compute for each user $u \in \mathcal{U}$ the similarity s_{cu} between this aggregate demand distribution and her individual user content preference distribution:

$$s_{cu} = \sum_{i\in\mathcal{I}} p_u(i)\ \hat{p}_c(i). \tag{13}$$

(iii) The association of each user u to SBS c is assigned a utility v_{cu} that combines both the similarity s_{cu} *and* the association cost b_{cu}:

$$v_{cu} = \frac{s_{cu}}{b_{cu}}. \tag{14}$$

(iv) Finally, users are associated to SBSs by solving an instance of the Generalized Assignment Problem, where users are mapped to jobs, cells to agents, and each possible assignment (user association) bears a profit s_{cu} and a cost b_{cu}. We solve the resulting GAP with an approximation algorithm, *e.g.*, the Martello-Toth approximation scheme in [9]. By A_c we denote the set of users that the algorithm associates to SBS c.

Step 2: Determine cache placements accounting for the impact of recommendations. In this step, we derive cache placements that maximize the cache hit ratio under the aggregate demand values $\{\hat{p}_c(i)\}$ computed in the first step.

(i) We first compute the content request probabilities of users, *as if* truthful recommendations $\mathcal{R}_u^t$ are issued to each user u, according to equations (1) and (2) for recommended and non-recommended items, respectively. Note that these are not the actual recommendations issued to the end-users; they only serve as intermediate estimates.

(ii) We then solve a 0-1 Knapsack Problem (KSP) for each SBS. The value v_{ic} that item i brings when cached in SBS c is the aggregate demand probability over all users $u \in A_c$

$$v_{ic} = \sum_{u\in A_c} d_u(i), \tag{15}$$

and the cost for caching it is its length L_i. A Dynamic Programming algorithm with pseudopolynomial complexity can be used [9] to solve this 0-1 KSP instance. The relation of caching and knapsack problems is quite standard in literature, *e.g.*, [11, 12].

Step 3: Align recommendations with cache placements to boost the aggregate cache hit ratio. As stated in Section 2, the recommendations that will be issued to users may deviate controllably from those corresponding to their implicit content preferences.

Hence, in this step we adjust the provisional recommendations determined in step 2, so that for each user u as many recommended items as possible lie both within the user's recommendation window $\mathcal{W}_u$ and the cache placement $\mathcal{P}_c$ at the SBS she is associated to.

4.1 Preliminary Results

Our preliminary numerical results show that even a naive greedy algorithm that takes into account the three problem dimensions is better by up to 5% than an optimal algorithm that considers only recommendations and caching. Fig. 2 depicts the performance of our heuristic for an instance of the problem with $|\mathcal{C}| = 15$ SBSs, $|\mathcal{U}|$=100 users, $|\mathcal{I}|$=300 items and 8 thematic categories. We assume that $R = 3$ items are recommended to each user, giving each recommended item an equal boost $r_u(i) = 1/R$, which fades out with R. Note that the maximum considered cache capacity reaches the 20% of the total content catalog size. Finally, we generate the recommendation weights w_u by sampling a uniform distribution defined in the interval [0.5, 0.7], in line with the experimental findings in [8].

We observe that our heuristic outperforms the upper bound of the JCRP solution, even under very strict storage capacity constraints, *i.e.*, less than 4% of the total catalog size. Moreover, our heuristic outperforms the reference scheme even in terms of required storage capacity, in order to obtain a specific cache hit ratio. Although the gains of the proposed scheme appear to be limited, it is important to observe that the proposed algorithm exhibits better performance than what is feasible when jointly controlling only the content caching and recommendations.

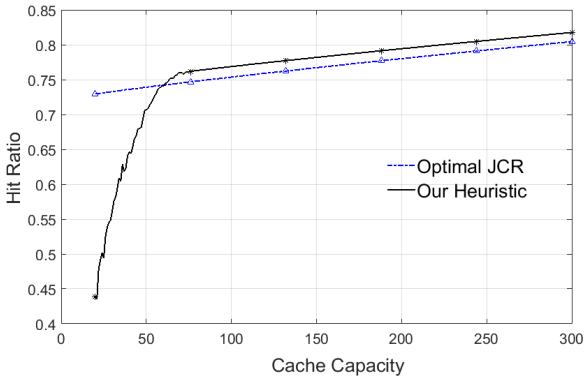

Figure 2: Aggregate cache hit ratio of our heuristic *vs.* upper bound achieved with JCRP [2].

5. CONCLUSION AND FUTURE WORK

In this work, we provided a system framework that jointly considers caching at the network edge, users requesting content items and a recommendation system that shapes the content preferences of users. We formulated the problem of exercising joint control over the user association to SBSs, the content caching and the recommendations and showed that it is NP-Hard. We also proposed a heuristic for the problem and showed preliminary results about its effectiveness.

Future Work. The main direction of our future work consists in the derivation of smarter algorithms for solving the JCRA Problem and their approximability analysis. The latter also involves the asymptotic behavior of the algorithm as different problem parameters scale up. In addition, we plan to devise an online version of our heuristic in order to reap the benefits of jointly controlling the three functions of the JCRA Problem in highly dynamic environments.

6. REFERENCES

[1] D. G. Cattrysse and L. N. V. Wassenhove. A survey of algorithms for the generalized assignment problem. *European Journal of Operational Research*, 60(3):260 – 272, 1992.

[2] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos. Jointly Optimizing Content Caching and Recommendations in Small Cell Networks. *IEEE Transactions on Mobile Computing*, pages 1–1, 2018.

[3] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos. Caching-Aware Recommendations: Nudging User Preferences towards better Caching Performance. In *Proc. IEEE INFOCOM*, pages 784–792, Atlanta, USA, May 2017.

[4] V. N. I. Cisco. Global mobile data traffic forecast update, 2015–2020 white paper. *Document ID*, 958959758, 2016.

[5] T. Giannakas, P. Sermpezis, and T. Spyropoulos. Show me the Cache: Optimizing Cache-Friendly Recommendations for Sequential Content Access. In *Proc. IEEE WoWMoM*, pages 1–9, Chania, Greece, June 2018.

[6] C. A. Gomez-Uribe and N. Hunt. The Netflix Recommender System: Algorithms, Business Value, and Innovation. *ACM Trans. on Management Information Systems*, 6(4):6(4):13:1–13:19, 2016.

[7] S. Kastanakis, P. Sermpezis, V. Kotronis, and X. Dimitropoulos. CABaRet: Leveraging Recommendation Systems for Mobile Edge Caching. In *ACM SIGCOMM workshops: Workshop on Mobile Edge Communications (MECOMM')*, pages 1–7, Budapest, Hungary, Aug. 2018. ACM.

[8] D. K. Krishnappa, M. Zink, C. Griwodz, and P. Halvorsen. Cache-Centric Video Recommendation: An Approach to Improve the Efficiency of YouTube Caches. *ACM Trans. on Multimedia Computer Communications Applications*, 11(4):11(4):1–20, June 2015.

[9] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, Inc., New York, NY, USA, 1990.

[10] D. Munaro, C. Delgado, and D. S. Menasché. Content Recommendation and Service Costs in Swarming Systems. In *Proc. IEEE ICC*, pages 5878–5883, June 2015.

[11] G. Neglia, D. Carra, and P. Michiardi. Cache Policies for Linear Utility Maximization. *IEEE/ACM Transactions on Networking*, 26(1):302–313, 2018.

[12] K. Poularakis, G. Iosifidis, A. Argyriou, and L. Tassiulas. Video Delivery Over Heterogeneous Cellular Networks: Optimizing Cost and Performance. In *Proc. IEEE INFOCOM*, pages 1078–1086, Toronto, Canada, April 2014.

[13] P. Sermpezis, T. Spyropoulos, L. Vigneri, and T. Giannakas. Femto-Caching with Soft Cache Hits: Improving Performance through Recommendation and Delivery of Related Content. *IEEE Journal on Selected Areas in Communications*, Feb. 2018.