Match to Cache: Joint User Association and Backhaul Allocation in Cache-aware Small Cell Networks

Francesco Pantisano[1], Mehdi Bennis[2], Walid Saad[3], and Mérouane Debbah[4]

[1] JRC - Joint Research Centre, European Commission, Ispra, Italy, email: francesco.pantisano@jrc.ec.europa.eu
[2] CWC - Centre for Wireless Communications, Oulu, Finland, email: bennis@ee.oulu.fi
[3] Wireless@VT, Bradley Department of Electrical and Computer Engineering, Blacksburg, VA, USA, email: walids@vt.edu
[4] Mathematical and Algorithmic Sciences Lab, Huawei France R&D, Paris, France, email: merouane.debbah@huawei.com

***Abstract*—Caching multimedia files at the network edge has been identified as a key technology for enhancing users' quality-of-service (QoS), while reducing redundant transmissions over capacity-constrained backhauls. Nevertheless, in small cell networks, the efficiency of a caching policy depends on the ability of small base stations (SBSs) to anticipate the requests from the user equipments (UEs). In this paper, we propose a collaborative filtering (CF) scheme for estimating the required backhaul usage at each SBS, by mining the *cacheability* of UEs' file requests. In the proposed approach, each SBS has a two-fold objective: update the bandwidth allocation based on the estimated backhaul utilization, and, given the current bandwidth availability, identify which UEs to service. We formulate the problem as a one-to many matching game between SBSs and UEs, and we propose a novel cache-aware user association algorithm that minimizes the backhaul usage at each SBS, subject to individual QoS requirements. Simulation results, based on real-world service request logs, have shown that the proposed CF-based solution can yield significant gains in terms of backhaul efficiency and cache hit-ratio, reaching up to 25%, with a maximum gap of 9% to an optimal cache-aware association technique.**

I. Introduction

The exponential growth in the demand for high data rates and quality-of-service (QoS) in wireless cellular networks has led to the introduction of ultra-dense architectures, based on the concept of small base stations (SBSs), such as picocells or femtocells. SBS deployments promise to deliver high QoS, at low operational costs [1], yet, in order to reap those benefits, a number of technical challenges stemming from the backhaul capacity limitations must be addressed [2]. In fact, due to high SBS network density, efficient and scalable backhaul management solutions are essential to harness traffic bottlenecks and deliver the desired performance.

To overcome the backhaul capacity limitations, state-of-the-art SBS architectures propose local *caching* of popular contents at SBS level, in order to reduce the overall traffic load from the core network and, thus, utilize the backhaul bandwidth more efficiently. Caching has been originally proposed in content distribution networks for decentralizing the availability of contents at strategic nodes of the network (e.g., proxy servers, gateways), while balancing the network traffic during off-peak intervals [3–5]. In essence, by decoupling the time instant during which a content is downloaded, from the one during which it is delivered to a UE, an SBS can boost the users' QoS and make a more efficient use of the backhaul resources.

The research leading to this paper has been partly supported by the Celtic-Plus project SHARING (proj. C2012/1-8), the U.S. National Science Foundation under grants CNS-1460316, CNS-1460333, CNS-1443914, and AST-1443913, and the ERC Starting Grant 305123 MORE.

As the efficiency of caching depends on the ability of each SBS to intelligently select which files to store, i.e., to enhance the *hit-ratio* of file requests, several works in the literature have aimed at maximizing the hit-ratio by adequately dimensioning the SBSs' caches [3], [6] or through proactive techniques [7–9]. Also, due to the limited storage capacity at each SBS, it is critical to properly decide which content to cache by taking into account the content *popularity*. Based on the popularity distribution, an SBS can estimate the probability of a file being requested and, accordingly, predict the backhaul bandwidth utilization, depending on whether such files are already available in the SBS' cache or have to be retrieved from the core network. In such studies, a common simplification is to assume that the global file popularity is known and modeled according to the Zipf distribution [7], [8], [10]. Although this assumption is valid for heterogeneous data sets, it fails to account for the different "cacheability" of certain file types [11]. For instance, in several multimedia streaming applications (e.g., YouTube, Netflix, Spotify), the contents are recommended to the users, or they are arranged in playlists, which ultimately create a logical link across a UE's file requests. Inferring such information allows to anticipate traditional network functions (e.g., pre-allocating UEs' bandwidth) which is a promising – and yet relatively unexplored – research field in wireless networks [7–9].

Making accurate predictions on future file requests demands the ability to mine the similarities across such requests and to combine them into probability distributions. In this respect, collaborative filtering (CF) has been one of the most successful techniques for building recommendation systems across large datasets. In practice, CF uses the known preferences of a group of users to make recommendations or predictions of the preferences for other unknown users. While being effective, CF suffers from scalability issues as the size of the file library grows. In case of large sets, user-based CF (based on individual requests from UEs) has been proven to be a more scalable option for deducing request similarity from users' past service requests, rather than from file libraries [12].

The main contribution of this paper is to exploit user-based CF predictions to improve the backhaul efficiency, in the downlink of small cell networks. By inferring on the popularity distribution of multimedia files, the SBSs can make better informed decisions on *which* UE should be serviced for a target QoS requirement, and *how* to allocate the backhaul bandwidth, accordingly. We address this problem in two phases. First, we model the file popularity through the *cacheability* metric [11], which quantifies the benefits of caching files with large revisit rate. Next, based

on the estimated incoming file requests and the current cache composition, the SBSs individually devise a UE-SBS association that minimizes the backhaul bandwidth allocation. In summary, the proposed CF-based approach enables each SBS to perform more precise backhaul bandwidth allocations, by harnessing storage and backhaul capacity limitations.

The rest of this paper is organized as follows. In Section II, we introduce the system model and the CF framework. In Section III, we formulate the UE-SBS association problem and propose a decentralized algorithm that converges to the targeted solution. Simulation results are analyzed in Section IV. Finally, conclusions are drawn in Section V.

II. System Model

A. Network Setting

Consider the *downlink* transmission of a single orthogonal frequency division multiple access (OFDMA) macro-cell. In this network, M UEs and N SBSs are deployed, respectively denoted by the sets $\mathcal{M} = \{1, ..., M\}$ and $\mathcal{N} = \{1, ..., N\}$. Let $\mathcal{L}_i$ denote the set of UEs serviced by SBS i. The macro-cell spectrum is divided in orthogonal frequency subbands, and each SBS i allocates one subband $w_{i,m}$ to each UE $m \in \mathcal{L}_i$. The transmit power of each SBS $i \in \mathcal{N}$ is denoted by p_i. Each SBS i is connected to the core network via a backhaul of capacity B_i. Each UE m requests a set of files $\mathcal{F}_m = \{1, ..., F_m\}$, $\mathcal{F}_m \subset \mathcal{F}$. For simplicity, we assume that all files have the same size s[1]. Finally, for the transmission of the files in $\mathcal{F}_m$, the instantaneous capacity between each SBS i and UE m is given by:

$$r_{i,m}(t) = w_{i,m} \log(1 + \gamma_{i,m}(t)), \tag{1}$$

where $g_{i,m}(t)$ is the channel gain between UE m and SBS i, at time t, $\gamma_{i,m}(t) = \frac{p_i g_{i,m}(t)}{\sigma^2 + I_{i,m}(t)}$ is the instantaneous signal-to-interference-plus-noise ratio (SINR) between SBS i and UE m and σ^2 the variance of the Gaussian noise. Moreover, the interference component $I_{i,m}(t) = \sum_{j \neq i} p_j g_{j,m}(t)$, denotes the interference produced by the transmissions from other SBSs j to their respective UE n, which takes place on the same frequency band $w_{i,m}$ allocated to UE m. Here, p_j, and $g_{j,m}(t)$ denote, respectively, the transmit power and the channel gain between SBS j and UE m. Finally, each UE m has a minimum data rate requirement r^*_m.

In order to accommodate the users' traffic requests, at time instant[2] t, each SBS i allocates a backhaul bandwidth $B_{i,m}(t)$ to each UE $m \in \mathcal{L}_i$, such that $\sum_{m \in \mathcal{L}_i} B_{i,m}(t) \leq B_i$. In such a setting, the quality of the transmission stream depends on the wireless channel conditions (e.g., interference) and on the backhaul capacity $B_{i,m}(t)$ that SBS i allocates to the UE's traffic requests. As a result, the maximum data rate at which the files in $\mathcal{F}_m$ can be delivered, from an SBS i to a UE m, is:

$$C_{i,m}(t) = \min\{B_{i,m}(t), r_{i,m}(t)\}. \tag{2}$$

[1]For example, in a video application such as YouTube, each file corresponds to a short segment of a video, a few seconds long. For instance, 100 different popular video clips with a length of 30 seconds each would correspond to more than 3000 different files.

[2]In the considered setting, time can be discretized into scheduling intervals, during which file requests are sent and radio resource management operations performed.

Note that, if the backhaul capacity B_i is insufficient for keeping up with the transmission data rate $r_{i,m}(t)$ (i.e., $B_{i,m}(t) < r_{i,m}(t)$), UE m can experience a considerable QoS degradation (e.g., low resolution or playback, for video applications), for reasons that are independent from the quality of the wireless transmission. To overcome such limitations, we consider that each SBS i is equipped with a data storage unit having a capacity of K_i bytes in which it can locally store a subset of the files in $\mathcal{F}$, denoted by $\mathcal{D}_i = \{1, \ldots, D_i\}$. This caching procedure continues until the storage capacity K_i is exhausted. Upon reaching the maximum storage capacity K_i, the least requested files are systematically dropped to accommodate new file entries.

Introducing caching capabilities at SBS level yields two noteworthy considerations. First, the files in local caches can be transmitted at data rates that are no longer affected by the backhaul status, since the constraint in (2) no longer applies. Second, the backhaul allocation $B_{i,m}(t)$ exclusively caters for the files that are not locally available at SBSs i. As a result, dimensioning the backhaul allocation, based on the current cache composition, directly affects the number of UEs an SBS can service and their respective QoS. In the next sections, we will discuss how the popularity of multimedia files can be exploited for devising smarter UE-SBS associations and optimizing the backhaul usage.

B. Cache-Aware Backhaul Allocation via Collaborative Filtering

In cache-aware networks, the backhaul allocation exclusively caters for the uncached files. Thus, an efficient backhaul allocation policy needs to estimate the backhaul utilization by inferring the probability of *uncached* files being requested. However, each SBS' knowledge is limited to the file requests sent from its previously serviced UEs. Thus, obtaining reliable popularity statistics from a limited set of information requires robust and systematic prediction approaches. To this aim, we propose a CF-based approach to estimate the file popularities [12].

In the proposed CF setting, each SBS i maintains a matrix with the UEs $m \in \mathcal{L}_i$ and the number of requests for all known files $\mathcal{F}$ over time. Such information can be represented in form of matrix as shown in Table I. The f-th column of Table I refers to the number of requests for file f, made by all the UEs $m \in \mathcal{L}_i$. The m-th row of Table I encompasses the number of requests of user m for the files in $\mathcal{F}$. Finally, the dashes denote unavailable information, for file requests which still have not occurred. The goal of the proposed CF approach is to exploit the correlations across different file requests (i.e., across columns), and users (across rows), so as to compute the probability of a file being requested by another UE, i.e., its *cacheability*. Based on that, each SBS has a two-fold objective: update the backhaul allocation based on the estimated backhaul utilization, and, given the current bandwidth availability, identify which UEs can be serviced.

In practice, when a UE m requests a file $f \in \mathcal{F}$, an SBS i constructs two sets of information on f, called *neighborhoods*, that serve as baseline for predicting the cacheability of file f. The first neighborhood $\mathcal{S}_f \subseteq \mathcal{M}$ is composed by the number of requests by other UEs $n \neq m$ previously serviced by SBS i for the same file f (equivalent to a subset of the f-th column of Table I). The second neighborhood $\mathcal{S}_m \subseteq \mathcal{F}_m$ is defined by

TABLE I
REPRESENTATION OF THE NUMBER OF FILE REQUESTS AT SBS i.

	$f=1$	$f=2$	...	$f=\lvert\mathcal{F}\rvert$
$m=1$	2	-	...	-
$m=2$	-	4	...	8
...	...	...	...	...
$m=\lvert\mathcal{L}_i\rvert$	-	14	...	-

the number of requests of other files in $\mathcal{F}_m$ requested by UE m (subset of the m-th row of Table I). Finally, the estimated number of requests of a UE m for a file $f \in \mathcal{F}_m$, associated to an SBS i, at time t can be expressed as [12]:

$$\hat{x}^f_{i,m}(t) = (\bar{x}_i + b_f + b_m) + \frac{\sum_{n\in\mathcal{S}_f} d_{m,n}\,\hat{x}^f_{i,n}}{\sum_{n\in\mathcal{S}_f}|d_{m,n}|} + \frac{\sum_{\dot{f}\in\mathcal{S}_m} d_{f,\dot{f}}\,\hat{x}^{\dot{f}}_{i,m}}{\sum_{\dot{f}\in\mathcal{S}_m}|d_{f,\dot{f}}|}, \tag{3}$$

where, $\bar{x}_i$ is the average number of requests received by SBS i by all known UEs and for all files in $\mathcal{F}$; $b_f = \frac{\sum_{m\in\mathcal{S}_f}\hat{x}^f_{i,m}}{|\mathcal{S}_f|} - \bar{x}_i$ denotes the average number of requests for file f, made by the UEs $m \in \mathcal{L}_i$, relative to the average $\bar{x}_i$; $b_f = \frac{\sum_{f\in\mathcal{S}_m}\hat{x}^f_{i,m}}{|\mathcal{S}_m|} - \bar{x}_i$ denotes the average number of requests by UE m over all the files $\dot{f}$, with respect to the average $\bar{x}_i$. Finally, $d_{m,n} = |\hat{x}^f_{i,m} - \hat{x}^f_{i,n}|$ and $d_{f,\dot{f}} = |\hat{x}^f_{i,m} - \hat{x}^{\dot{f}}_{i,m}|$ respectively denote the absolute error (also referred to as distance) between the number of requests of different UEs serviced by SBS i (i.e., the neighborhood of file f), and different files for a given UE m (neighborhood of UE m). The nominal value of the unavailable entries (denoted by dashes) is set to zero.

To quantify the benefits of caching, we adopt a metric similar to the *cacheability* metric proposed in [11] for quantifying the popularity of HTTP requests. This metric $\hat{\rho}_{m,i,f}(t)$ represents the probability of a cached file f being requested by UE m to SBS i, given the past requests of UE m and all the other UEs $n \in \mathcal{L}_i$ serviced by SBS i:

$$\hat{\rho}_{m,i,f}(t) = \frac{(\hat{x}^f_{i,m}(t) - 1)}{\sum_{n\in\mathcal{L}_i}\hat{x}^f_{i,n}(t)},\ \forall i \in \mathcal{N}. \tag{4}$$

In such a setting, a suitable way for exploiting the file cacheability inferred from the above model is to estimate the backhaul bandwidth required for accommodating the users' file requests. In fact, if a UE is likely to request a cached file, it can be assigned a minimum backhaul bandwidth $\hat{B}_{i,m}(t)$. Conversely, instantaneous requests demand larger allocations, according to the data rate requirement r^*_m. To capture such aspects, we consider that the estimated backhaul bandwidth requirement for a UE m associated to SBS i is given by:

$$\hat{B}_{i,m}(\mathcal{L}_i, \mathcal{F}_m, \hat{\rho}_{m,i,f}, t) = \frac{1}{|\mathcal{F}_m|}\sum_{f\in\mathcal{F}_m}\Big(\frac{\hat{\rho}_{m,i,f}(t)\,r^*_m}{\sum_{n\in\mathcal{L}_i}\hat{\rho}_{n,i,f}(t)\,r^*_n}\Big)\cdot B_i. \tag{5}$$

Note that, the allocation in (5) depends on the probability $\hat{\rho}_{m,f}(t)$ that a UE m might request file f, and on the probability that other UEs $n \in \mathcal{L}_i$ might do so, as well. Both probabilities are computed as per (4), i.e., based on CF-based estimations. In case of even probability, the bandwidth allocations only depend on the transmission data rate requirements r^*_m and r^*_n.

III. CELL ASSOCIATION AS A MATCHING GAME

A. Problem Formulation

In this work, we aim at solving the problem assigning each UE $m \in \mathcal{M}$ to the SBS $i \in \mathcal{N}$, through a matching $\eta : \mathcal{M} \to \mathcal{N}$, that minimizes the long-term backhaul bandwidth allocation, given an initial cache composition $\mathcal{D}_i$ and the respective file request predictions. Essentially, this yields the following optimization problem:

$$\underset{(i,m)\in\eta}{\arg\min}\ \frac{1}{T}\sum_{t}^{T}\sum_{i\in\mathcal{N}}\sum_{m\in\mathcal{M}}\hat{B}_{i,m}(\mathcal{L}_i,\mathcal{F}_m,\hat{\rho}_{m,i,f},t) \tag{6}$$

$$\text{s.t.},\quad D_i \cdot s \le K_i, \tag{7}$$

$$\sum_{m\in\mathcal{L}_i}\hat{B}_{i,m}(\mathcal{L}_i,\mathcal{F}_m,\hat{\rho}_{m,i,f},t) \le B_i, \tag{8}$$

$$C_{i,m}(t) = \min\{\hat{B}_{i,m}(t), r_{i,m}(t)\}, \tag{9}$$

$$C_{i,m}(t) \ge r^*_m,\ \forall i \in \mathcal{N}. \tag{10}$$

where (7) is a constraint on the maximum storage capacity of each SBS, constraint (8) represents a limit on the backhaul bandwidth allocation, constraint (9) denotes the transmission capacity bottleneck, and constraint (10) indicates a minimum data rate requirement. In terms of complexity, the optimization problem in (6)-(10) is NP-complete, and depends on the number of SBSs and UEs in the network. Even by relaxing some of the constraints, the exponential complexity makes a centralized approach intractable, notably in small cell networks in which the number of UEs and SBSs can considerably grow. This complexity coupled with the need for self-organizing solutions mandates a distributed approach in which UEs and SBSs autonomously decide on the best UE-SBS association.

For solving the SBS-UE association problem in (6), one suitable framework is that of *matching theory* [13]. Matching theory provides a computationally tractable set of tools for solving a combinatorial problem such as (6). Essentially, a matching game is defined as follows:

Definition 1. *A* matching *game is defined by two sets of players* $(\mathcal{M},\mathcal{N})$ *and a function* $\eta : \{\mathcal{M}\cup\mathcal{N}\} \to \{\mathcal{M}\cup\mathcal{N}\}$*, such that:*

- $|\eta(m)| = 1$*, for every UE* $m \in \mathcal{M}$*,*
- $|\eta(i)| \le q_i$*, (or equivalently* $|\mathcal{L}_i| \le q_i$*) for every SBS* $i \in \mathcal{N}$*,*
- $\eta(m) = i$ *if and only if* $i = \eta(m)$*, or equivalently,* $m \in \mathcal{L}_i$*.*

As the UEs are unaware of the files stored at the SBS side, their preference is exclusively based on the transmission data rate $C_{i,m}(t)$. Thus, for a UE m, we define a preference relation $\succ_m$ over the set of SBSs $\mathcal{N}$ as:

$$i \succ_m j \Leftrightarrow C_{i,m}(t) > C_{j,m}(t). \tag{11}$$

At the SBS side, the preferences are mainly based on the backhaul bandwidth requirements of each UE, which are inferred by estimating the cacheability of the UE's incoming file requests, as discussed in Section II-B. Accordingly, for each SBS i, we define a preference relation $\succ_i$ over the set of UEs $\mathcal{M}$ as:

Algorithm 1: UE-SBS Cell Association Algorithm

Data: r^*_m, $\mathcal{D}_i$, B_i.

Begin;
Phase I - Backhaul bandwidth requirements and interference estimation;
• Each UE m discovers the interfering SBSs in the vicinity and measures the received interference;
• Each SBS estimates the cacheability $\hat{\rho}_{m,i,f}(t)$ of the file requests from the UEs within transmission range;
Phase II - UE-SBS matching negotiations;
repeat
  • The bandwidth allocation $\hat{B}_{i,m}(t)$ is updated based on $\hat{\rho}_{m,i,f}(t)$ and the current η;
  • UEs and SBSs are sorted by $\succ_m$ and $\succ_i$;
  if $j \succ_m i$ **then**
    • UE m sends a proposal to SBS j;
    • SBS j computes $\hat{B}_{j,m}(t)$ for the new link (j,m);
    if *(7)-(10) are satisfied* **then**
      • the new link (j,m) is created;
    else
      • SBS j refuses the proposal, and UE m sends a proposal to the next preference.
    end
  end
until $\nexists n, j \ : n \succ_i m$ *and* $m \succ_j n$;
Outcome: Stable matching η;

$$m \succ_i n \Leftrightarrow \hat{B}_{i,m}(t) < \hat{B}_{i,n}(t). \tag{12}$$

To solve the problem in (6) in a decentralized approach, the SBSs and UEs can individually rank one another, based on the preference relations $\succ_m$, $\succ_i$. The aim of each SBS is to maximize its own utility, or equivalently, to accommodate most file requests, given its backhaul bandwidth availability and the set of cached files $\mathcal{D}_i$. Similarly, the aim of each UE m is to be associated with the SBS delivering the largest data rate $C_{i,m}(t)$ for its requested files.

B. Proposed algorithm and properties

To find a UE-SBS matching for the problem in (6), we propose a new approach, shown in Algorithm 1, inspired by the deferred acceptance scheme proposed by Gale and Shapley [13].

In this regard, a sufficient condition for the stability of the proposed matching is given by Gale and Shapley [13] and hereby adapted to the problem in (6):

Definition 2. *A UE-SBS association is* stable *if there does not exist two UEs m, n, that are respectively serviced by two SBSs i and j, although m prefers j to i, and n prefers i to j.*

Finally, at the end of Phase II of Algorithm 1, the SBS have converged to a final matching, whose stability is guaranteed by the following proposition:

Proposition 1. *The proposed Algorithm 1 is based on the deferred acceptance algorithm, thus, it is guaranteed to converge to a stable matching in a finite number of iterations, as per [13].*

IV. Performance Evaluation

For our simulations, we consider a single cell of a macro-cellular network with a spectrum bandwidth of 20 MHz. In this cell, $M = [40, 500]$ UEs and $N = [20, 230]$ SBSs are uniformly deployed. The transmit power of each SBS i is $p_i = 33$ dBm.

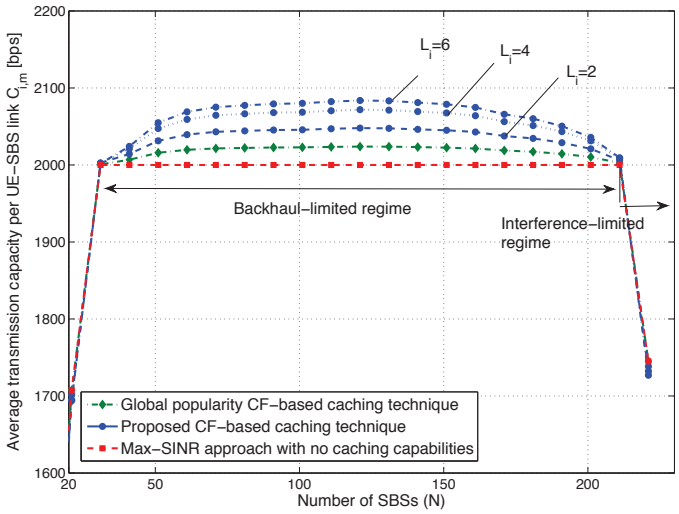

Fig. 1. Average transmission capacity per UE-SBS link vs number of SBSs in the network (N). $|\mathcal{L}_i| = \{2, 4, 5, 6\}$ UEs, $D_i = 0.45 \cdot |\mathcal{F}|$, $B = 2$ Gbps.

Transmissions are affected by distance dependent path loss, with path loss exponent 3, and shadowing according to 3GPP specifications [14]. The files in $\mathcal{F}$ have a size of $s = 10$ MB, each with a bitrate of $r^*_m = 450$ Kbps. Each SBS $i \in \mathcal{N}$ has a memory capacity chosen from an interval $K_i = [0.1, 0.6]$ TB. Similarly, the backhaul capacity is chosen from an interval $B_i = [0.5, 2]$ Gbps.

To evaluate our approach, we design an experiment in which we simulate the real situation of invoking a multimedia streaming service. In this regard, to model the UE's file requests, we consider the listening logs of the Last.fm dataset [15]. The considered file library $\mathcal{F}$ is composed by the top 10000 songs, each one assigned to a non-unique tag denoting the music genre. The probability of a UE requesting a file f is proportional to the number of access on Last.fm, yet files are arranged in playlists, based on genre or theme. Each UE requests a playlist of $F_m = 20$ files, out of a set of $|\mathcal{F}| = 10000$ files. The SBSs' caches are composed by a set of randomly selected files, and at each iteration, the least requested files are systematically dropped. T

For comparison purposes, we consider three additional UE-SBS association schemes, used as benchmarks. In the first scheme, each UE is associated to the respective SBS that maximizes the received SINR. Also, in this approach, the SBSs have no caching capabilities and the the contents are retrieved directly from the core network (i.e., subject to the constraint in 2. The second scheme is also based on caching and CF, but it considers the global file popularity (i.e., the number of requests averaged over all UEs and all files)[3], thus it is based on an average request distribution, which converges to the Zipf one. Finally, the third approach is a cache-aware association scheme in which both wireless transmission and cache consistency are optimized. The UE-SBS association of such an approach is computed in a centralized fashion, by numerically solving instances of the optimization problem in (6).

Figure 1 shows the average transmission capacity $C_{i,m}$ per UE-SBS link as a function of the SBS network size N, for different number of UEs $|\mathcal{L}_i|$ serviced by each SBS. Figure 1

[3]This approach is also known as baseline predictor, in the CF literature [12].

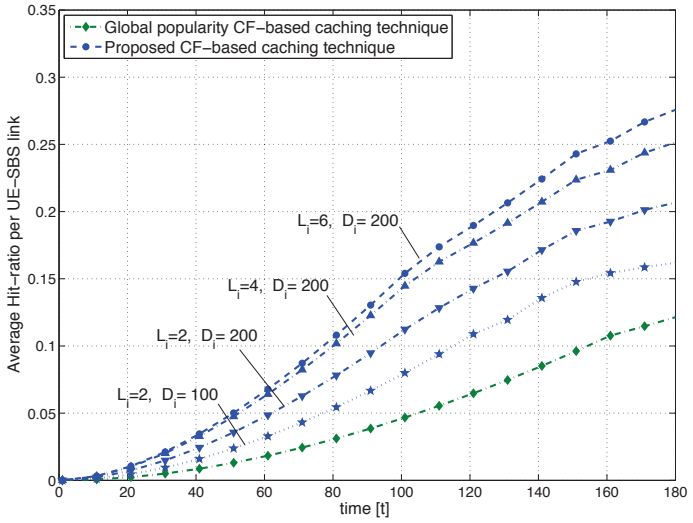

Fig. 2. Time evolution of the hit-ratio per UE-SBS link. $|\mathcal{L}_i| = \{2, 4, 6\}$ UEs, $D_i = \{0.22, 0.45\} \cdot |\mathcal{F}|$ files, $B = 1$ Gbps, N=100 SBSs.

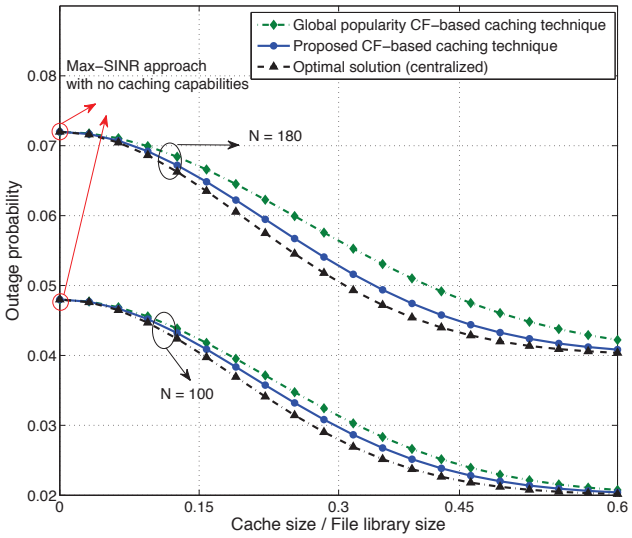

Fig. 3. Average outage probability ($Prob\ \{C_{i,m}(t) < r_m^*\}$) vs normalized cache size. $r_m^* = 450$ Kbps, $B = 1.8$ Gbps.

demonstrates that the backhaul capacity is the main limitation for association techniques with no caching capabilities. In such a regime, Figure 1 shows that the proposed caching strategy can overcome the backhaul capacity limitations by decentralizing selected files at SBS level. Also, Figure 1 demonstrates that exploiting the cacheability inferred from local UEs leads to more precise predictions of the UE's files request, ans such gains increase with $|\mathcal{L}_i|$ (i.e., the CF neighborhood). For example, Figure 1 shows that the performance gap between the proposed approach and the global popularity scheme is 4.5%, for SBSs with a backhaul capacity of $B_i = 1$ Mbps and a cache of $0.45 \cdot |\mathcal{F}|$ files. Finally, the gains stemming from caching saturate for larger networks, when the co-channel interference becomes the main hindrance for QoS delivery (i.e., $N \geq 220$ SBSs). Therefore, Figure 1 demonstrates that the proposed cache-based approach yields significant utility gains by exploiting local content availability, notably in networks with a limited-capacity backhaul.

In Figure 2, we observe the time evolution of the average hit-ratio per UE-SBS link for different number of serviced UEs per SBS, and cache size. Figure 2 shows that the accuracy of the proposed approach increases over time, since, at each iteration, the CF prediction is based on a larger number of requests (i.e., on a larger CF neighborhood S_m). Using uncoded caching techniques (i.e., storing complete files), the hit-ratio grows proportionally with the number of cached files. However, Figure 2 demonstrates that, by exploiting the file cacheability inferred from *local* UEs's requests, the predictions are more accurate and the hit-ratio can be further improved. For instance, the proposed approach achieves a hit-ratio of 25% by caching 20% of all the available files, after a simulation time of 160 s, in a network composed by $N = 100$ SBS, each one serving $|\mathcal{L}_i| = 6$ UEs. In summary, Figure 2 demonstrates that the proposed algorithm achieves good predictions with a reasonable initial delay, by reducing the bias in the estimated popularity distribution.

In Figure 3, we evaluate the average outage probability as a function of the cache size K_i, normalized to the file library size $|\mathcal{F}|$. In this figure, we can observe that the outage probability both depends on the received interference (which increases with the number of SBSs in the network) and the cache size. With respect to the latter, by exploiting local file availability, it becomes possible to overcome the transmission capacity limitations due to backhaul bottlenecks (e.g., Eq. (2), by reaching up to 43% reduction, with respect to a max-SINR association scheme. Also note that, for all the considered caching approaches, the probability of outage exhibits a floor, due to fact that SBSs with larger caches tend to service more UEs. This exposes the UEs to a larger received interference, which cannot resolved through caching optimization, rather with interference management techniques. Finally, Figure 3 demonstrates that more accurate predictions on the bandwidth utilization, obtained in the proposed CF-based approach are able to further the gap to the optimal performance, which does not exceed 8%, for a network of $N = 180$ SBSs, equipped with cache units of $0.4 \cdot |\mathcal{F}|$ files.

Figure 4 shows the cumulative distribution function of the transmission capacity $C_{i,m}$ per UE-SBS link, for all the considered schemes. Figure 4 shows that the benefits of caching are more significant for UEs experiencing severe backhaul bottlenecks. For example, at the tenth percentile (i.e., 10% of all UEs with the smallest backhaul bandwidth allocations), the performance of the max-SINR association technique yields 1550 bps (over a backhaul of capacity $B_i = 1800$ bps), while the proposed approach gains up to 19.8%, by reaching 1845 bps per UE-SBS link. Moreover, note that the max-SINR-based and the optimal caching approaches (respectively denoted by the red and the black curves) reflect two opposite optimization criteria: the first, based on wireless properties; the second, content-centric. Hereby, on average, the gap between the proposed (decentralized) CF-based approach and an optimal (centralized) cache-enabled association scheme is 7%. This gap can be further reduced by optimizing the CF neighborhood size $\mathcal{S}_f$ or by exploting additional context information on the users' preferences.

Finally, in Figure 5, we evaluate the average transmission

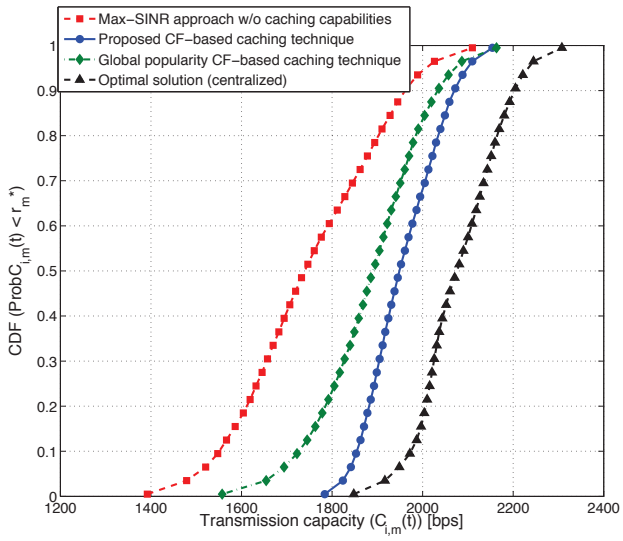

Fig. 4. Cumulative distribution function of the transmission capacity $C_{i,m}(t)$ per UE-SBS link. $N = 100$ SBSs, $r_m^* = 450$ Kbps, $B = 2$ Gbps.

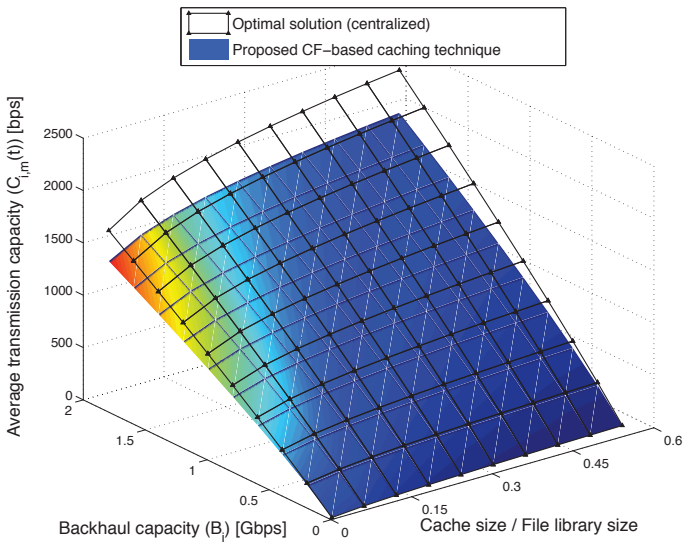

Fig. 5. Average transmission capacity as a function of the normalized cache size $D_i/|\mathcal{F}|$ and backhaul capacity B_i. $N = 100$ SBSs, $r_m^* = 450$ Kbps.

capacity as a function of the normalized cache size and the backhaul capacity, for the proposed CF-based caching scheme and an optimal centralized solution. The color map indicates the gradient of the transmission capacity as a function of the normalized cache size. From Figure 5, we observe that the gains of caching scale linearly for increasing cache sizes and backhaul capacities. Nevertheless, for large backhaul capacities ($B_i = 2$ Gbps), with less stringent bottlenecks, the gains saturate more quickly as the cache size grows, mainly reflecting the caches limitedness. This evidence suggests that, at architectural level, caching techniques should be carefully tuned (in terms of cache size and content placement), by accounting for local backhaul constraints, rather than network-averaged conditions. Finally, the comparison with an optimal centralized solution shows a maximum performance gap of 9%. In summary, Figures 4 and 5 demonstrate that the proposed CF-based UE-SBS association scheme can yield near-optimal performance in terms of transmission capacity increase, in a decentralized, uncoordinated fashion.

V. Conclusions

In this paper, we have presented a joint approach to the UE-SBS association and backhaul bandwidth management, in wireless small cell networks. We have proposed a CF-based recommendation scheme that enables each SBS to estimate the probability of its cached files being requested, based on the history of its past serviced UEs. We have combined such information into a UE-SBS association scheme by accounting for the backhaul utilization and individual data rate requirements. In the analytical formulation, we have modeled the problem as a one-to-many matching game, in which the SBS and UE devise individual preferences over one another. We have proposed an algorithm that enables the UEs and SBSs to generate a list of preferences that are respectively based on the transmission capacity and the cacheability of the UEs' file requests. Simulation results have shown that, by exploiting the correlations across local UEs' requests, the proposed CF-based solution can yield significant gains in terms of hit-ratio, reaching up to 25%, with a maximum gap of 9% to an optimal cache-aware association technique.

References

[1] J. Andrews, H. Claussen, M. Dohler, S. Rangan, and M. Reed, "Femtocells: Past, present, and future," *IEEE Journal on Sel. Areas in Comm.*, vol. 30, no. 3, pp. 497 –508, Apr. 2012.

[2] T. Q. S. Quek, G. de la Roche, I. Guvenc, and M. Kountouris, *Small Cell Networks: Deployment, PHY Techniques, and Resource Management*. New York, USA: Cambridge University Press, Sept. 2012.

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

[4] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch, and G. Caire, "Wireless video content delivery through coded distributed caching," in *In Proc. of IEEE Int'l Conf. on Communications (ICC)*, June 2012, pp. 2467–2472.

[5] ——, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *In Proc. of IEEE INFOCOM*, March 2012, pp. 1107–1115.

[6] G.-M. Chiu and C.-R. Young, "Exploiting in-zone broadcasts for cache sharing in mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 3, pp. 384–397, March 2009.

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

[8] E. Bastug, J.-L. Guenego, and M. Debbah, "Proactive small cell networks," in *In Proc. of Int'l Conf. on Telecommunications (ICT)*, May 2013, pp. 1–5.

[9] F. Pantisano, M. Bennis, W. Saad, S. Valentin, and M. Debbah, "Matching with externalities for context-aware user-cell association in small cell networks," in *In Proc. of Asilomar Conf. on Signals, Systems, and Computers*, Nov 2013, pp. 1–6.

[10] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in *In Proc. of IEEE INFOCOM*, vol. 1, Mar 1999, pp. 126–134 vol.1.

[11] B. Ramanan, L. Drabeck, M. Haner, N. Nithi, T. Klein, and C. Sawkar, "Cacheability analysis of HTTP traffic in an operational LTE network," in *Wireless Telecommunications Symposium (WTS), 2013*, April 2013, pp. 1–8.

[12] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in Artificial Intelligence*, vol. 8, no. 3, pp. 1–19, March 2009.

[13] A. Roth and M. A. O. Sotomayor, *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. New York, USA: Cambridge Press, 1992.

[14] S. Sesia, I. Toufik, and M. Baker, *LTE - the UMTS long term evolution*. A John Wiley and Son publication - UK, Aug. 2009.

[15] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The million song dataset," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.