Cache-aware User Association in Backhaul-Constrained Small Cell Networks

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

Invited Paper

[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]Electrical and Computer Engineering Department, University of Miami, Coral Gables, FL, USA, email: walid@miami.edu
[4] Alcatel-Lucent Chair in Flexible Radio, SUPÉLEC, Gif-sur-Yvette, France, email: merouane.debbah@supelec.fr

***Abstract*—Anticipating multimedia file requests via caching at the small cell base stations (SBSs) of a cellular network has emerged as a promising technique for optimizing the quality of service (QoS) of wireless user equipments (UEs). However, developing efficient caching strategies must properly account for specific small cell constraints, such as backhaul congestion and limited storage capacity. In this paper, we address the problem of devising a user-cell association, in which the SBSs exploit caching capabilities to overcome the backhaul capacity limitations and enhance the users' QoS. In the proposed approach, the SBSs individually decide on which UEs to service based on both content availability and on the data rates they can deliver, given the interference and backhaul capacity limitations. We formulate the problem as a one-to-many matching game between SBSs and UEs. To solve this game, we propose a distributed algorithm, based on the deferred acceptance scheme, that enables the players (i.e., UEs and SBSs) to self-organize into a stable matching, in a reasonable number of algorithm iterations. Simulation results show that the proposed cell association scheme yields significant gains, reaching up to 21% improvement compared to a traditional cell association techniques with no caching considerations.**

I. Introduction

Meeting the stringent quality-of-service (QoS) requirements of emerging wireless services such as multimedia streaming and mobile TV has led to the introduction of novel wireless cellular architectures. Among such architectures, the concept of small cell base stations (SBSs), such as picocells, microcells or femtocells overlaid on existing macro-cellular wireless systems, has emerged as a key solution for delivering high QoS, at low operational costs [1]. In order to reap the benefits of small cell deployments, a number of technical challenges must be addressed such as interference management, load balancing, and capacity limited backhaul links [2].

To overcome the backhaul capacity limitations, state-of-the-art SBS architectures propose the integration of offloading techniques and data storage units. In fact, as predicted by Moore's law (and, more recently, by Kryder's law), the capacity of modern-day storage units has increased exponentially over the past thirty years with consistently declining costs per stored bit [3]. Driven by this trend, the introduction of storage units within cellular architectures is now seen as an attractive solution to overcome the backhaul limitations of small cell networks [4], [5].

One promising technique for offloading data from the backhaul of small cell networks is data *caching*. Caching has been originally proposed in content distribution networks for enhancing data locality, i.e., by content replication at strategic nodes of the network (e.g., proxy servers), while balancing the network traffic during off-peak intervals [6], [7]. Similarly, an SBS can overcome the limitations of a congested backhaul, by downloading data contents and, subsequently, buffering them during the periods of time in which the backhaul is less congested. By doing so, the SBSs are able to boost the QoS of the users and reduce traffic over the limited-capacity backhaul links.

Most existing works on caching in cellular networks have focused on enhancing the users' QoS by leveraging decentralized cloud storage [6], [7], by offloading traffic to device to device (D2D) communication links [8], or by proactive techniques [9] (and references therein). Moreover, the benefits of data caching have been evaluated in terms of energy efficiency [10] or by exploring both spatial and social links among the users [11], [12]. This body of work sheds light on an important tradeoff in small cell networks with caching capabilities. On the one hand, in order to increase the probability of meeting the UEs' traffic demand, each SBS should download large amounts of diversified contents. On the other hand, the amount of cached data is ultimately limited by the backhaul bandwidth and the storage capacity at each SBS. As a result, the concept of caching is not uniformly applicable to all the SBSs in a network, as each SBS experiences unique network conditions due to the number of UEs currently serviced and the existing traffic load on the backhaul. In summary, leveraging caching in small cell networks demands novel, decentralized approaches in which each SBS decides on which UEs to service, based on its local file availability and the currently experienced network conditions.

The main contribution of this paper is to address the problem of UE-SBS association given the state of the small cell backhaul and the caching capacity at each SBS. By assuming coarse localization estimation, we propose a framework in which the SBSs make individual decisions on *which* UE they should service, based on the availability of cached files, as well as the backhaul congestion state. We model the problem as a one-to-many matching problem and we propose a deferred acceptance algorithm to find a stable matching between UEs and SBSs, given

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-1253731 and CNS-1406947, and the ERC Starting Grant 305123 MORE.

978-3-901882-63-0/2014 - Copyright is with IFIP

the storage, backhaul and interference limitations. Simulation results show that, in the proposed cache-based approach, the SBSs overcome the backhaul capacity limitations and improve the UE's QoS delivery of traditional UE-SBS associations, yielding gains of up to 21% that grow linearly with the SBSs' storage capacity.

The rest of this paper is organized as follows. In Section II, we introduce the system model and network setting. In Section III, we formulate the UE-SBS association problem as a matching game, and we propose an algorithm to obtain a stable UE-SBS matching. Simulation results are analyzed in Section IV. Finally, conclusions are drawn in Section V.

II. System Model

Consider the *downlink* transmission of a single orthogonal frequency division multiple access (OFDMA) macro-cell. In this network, M mobile UEs and N SBSs are deployed, respectively denoted by the sets $\mathcal{M} = \{1, ..., M\}$ and $\mathcal{N} = \{1, ..., N\}$. Each SBS i can service at most q_i UE. We let $\mathcal{L}_i$ be 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. The SBSs are connected to the core network via a backhaul of limited capacity B_i. Over a time period T, each UE m requests a number of files f from a set $\mathcal{F}$. For simplicity, we assume that all files have the same size s. The backhaul bandwidth B_i is scheduled over time to accommodate the UEs' traffic requests. The files $f \in \mathcal{F}$ are requested based on their popularity, which is assumed to follow a Zipf distribution with parameter ψ [8]. Thus, each UE requests file f with probability $\frac{f^{-\psi}}{\sum_x^{|\mathcal{F}|} \frac{1}{x}^{\psi}}$, $x \in \mathcal{F}$. Let $\mathcal{F}_m = \{1, ..., F_m\}$, $\mathcal{F}_m \subset \mathcal{F}$ denote the files requested by UE m during T. For the transmission of the files in $\mathcal{F}_m$, the instantaneous capacity between each SBS i and UE m is given by:

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

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 $g_{j,m}$ denote, respectively, the transmit power and the channel gain between SBS j and UE m.

The UEs are considered to be mobile at a speed ν_m within the macro-cell modeled as a Manhattan grid map [13], as shown in Fig. 1. In such a grid model, a *path* is defined by a polyline with a start and an end point. Thus, a UE's mobility is fully described by its speed ν_m and its path, which are both chosen to be i.i.d.. While moving along its trajectory, we assume that each UE i reports its channel gain $g_{i,m}(t)$ to its serving SBS i. This channel state information (CSI) feedback is reported once per coherence time and is used for deciding on the associations between SBSs and mobile UEs. In fact, while the path loss only depends on the distance between the UE's location and the serving SBS, two

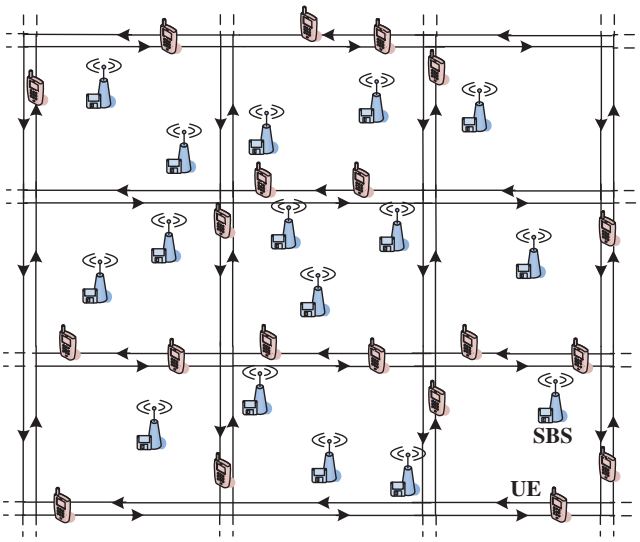

Fig. 1. Network scenario based on the Manhattan mobility model.

UEs on the same path are likely to experience different fading components, depending on their speed.

In classical networks, an SBS retrieves the UE's files $f \in \mathcal{F}_m$ only once an explicit request is made by the UE. In such a reactive protocol, the quality of the transmission stream depends on the wireless channel conditions (e.g., received interference) and on the backhaul capacity $B_{i,m}$ that SBS i allocates to the UE's traffic requests. As a result, in a traditional reactive approach, the maximum data rate at which the files in $\mathcal{F}_m$ can be delivered, from an SBS i to a UE m, is:

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

Note that, in case of backhaul traffic congestion, the backhaul capacity B_i is insufficient for keeping up with the transmission data rate $r_{i,m}(t)$ (i.e., $B_i < r_{i,m}(t)$). As a result, 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 assume that each SBS is equipped with a data storage unit having a capacity of K_i bytes, that are used to download data files (e.g., popular video files) in $\mathcal{F}$ to be stored at the SBS level, prior to a UE's requests. Hence, when an SBS i is not servicing any UE[1], it can cache a set of files $\mathcal{D}_i = \{1, ..., D_i\}$, $\mathcal{D}_i \subset \mathcal{F}$, by downloading them from the core network via the backhaul. Note that, by locally caching the files $\mathcal{D}_i$, an SBS can enhance the UE's QoS, by transmitting at data rates that are no longer affected by the backhaul status, since the constraint in (2) no longer applies.

This caching procedure can continue until the storage capacity K_i is exhausted. Upon reaching the maximum storage capacity K_i, the least popular files are systematically dropped to accommodate new file entries, while verifying the storage capacity constraint:

$$D_i \cdot s \leq K_i \ [bits]. \quad (3)$$

[1]Equivalently, an SBS can keep copies of the files that have been transmitted to its UEs over time.

When applying caching techniques to small cell networks, it must be noted that the proportion of cached data is not equal at all SBSs, as it depends on the backhaul conditions experienced by each SBS, and on their storage capabilities. As a result, data caching techniques cannot be applied uniformly to each SBS and, thus, they require novel decentralized approaches in which each SBS selects its own caching strategy, by accounting for both the local storage capacity and the network properties (i.e., backhaul capacity, received interference).

In the following section, we will describe how each SBSs can devise an individual caching strategy, while accounting for the mobility pattern of the incoming users and the network properties.

III. Cache-aware User Association as a Matching Game

A. Problem formulation

Given the system model presented in the previous section, our key goal is to study the problem of UE-SBS association, by focusing on *which UE* should be serviced by each SBS, given a set of locally available files $\mathcal{D}_i$. We consider that each SBS keeps track of the CSI feedbacks $g_{i,m}(t)$ that are periodically reported by each UE m in its vicinity. Based on the CSI sequence, an SBS can learn the UEs' speed and direction of arrival, and thus infer[2] the time instants $t_{i,m}^{IN}$ and $t_{i,m}^{OUT}$, at which user m will arrive and leave cell i. Once a user is associated (at time $t_{i,m}^{IN}$), the files that are available in the local storage units are transmitted first, at an instantaneous transmission data rate of $r_{i,m}(t)$ bps. The amount of data cached at SBS i transmitted to a UE m is: $|\mathcal{F}_m \cap \mathcal{D}_i| \cdot s$, and the estimated time to accomplish that is:

$$\hat{\tau}_{i,m}(t) = \frac{|\mathcal{F}_m \cap \mathcal{D}_i| \cdot s}{r_{i,m}(t)} \ [sec]. \tag{4}$$

Note that the estimated time in (4) for delivering the files in the cache of SBS i depends on the instantaneous data rate $r_{i,m}(t)$. Therefore, in case $\hat{\tau}_{i,m}(t) \geq t_{i,m}^{OUT}$, only a portion of the cached data can be delivered to UE m, precisely, until UE m leaves cell i at time $t_{i,m}^{OUT}$. As a result, the data cached at SBS i can be transmitted a UE m starting from $t_{i,m}^{IN}$ until time limit $\hat{\tau}_{i,m}^{max}(t)$, defined as:

$$\hat{\tau}_{i,m}^{max} = \min\{\hat{\tau}_{i,m}(t), t_{i,m}^{OUT}\} \ [sec]. \tag{5}$$

When a UE m requests a file f that is not locally available in the SBS cache, that file is retrieved from the core network, via the backhaul. In this case, the files are delivered to the UE m at a transmission rate $C_{i,m}(t)$ as per (2), depending on whether the bottleneck is represented by the backhaul capacity or the transmission data rate. In order to formalize the UE-SBS association problem, we define a suitable utility function for each UE $m \in \mathcal{M}$ seeking a set of files $f \in \mathcal{F}_m$, and being serviced by SBS $i \in \mathcal{N}$, as the amount of bits that SBS i delivered to UE m during the service time $[t_{i,m}^{IN}, t_{i,m}^{OUT}]$:

$$U_{i,m}(t_{i,m}^{IN}, t_{i,m}^{OUT}, \hat{\tau}_{i,m}^{max}) = \tag{6}$$
$$= \frac{\sum_{t=t_{i,m}^{IN}}^{\hat{\tau}_{i,m}^{max}} r_{i,m}(t)\Delta(g_{i,m}(t)) + \sum_{t=\hat{\tau}_{i,m}^{max}}^{t_{i,m}^{OUT}} C_{i,m}(t)\Delta(g_{i,m}(t))}{t_{i,m}^{OUT} - t_{i,m}^{IN}},$$

where $\Delta(g_{i,m}(t))$ is the interval duration between two consecutive time instants t and it is assumed to be equal to the coherence time at time t. In other words, $\Delta(g_{i,m}(t))$ is the duration during which the channel is unchanged, starting from time instant t.

In the utility (6), we can see that the files $f \in \{\mathcal{F}_m \cap \mathcal{D}_i\}$ requested by UE m, and already available at SBS i, are transmitted at data rate $r_{i,m}$ during $[t_{i,m}^{IN}, \hat{\tau}_{i,m}^{max}]$. In addition, the files $f \in \{\mathcal{F}_m \setminus \mathcal{D}_i\}$, that have to be downloaded from the core network, are transmitted during $[\hat{\tau}_{i,m}^{max}, t_{i,m}^{OUT}]$ and subject to the constraints in (2) and (5). As a result, while the QoS of cached files delivery only depends on the wireless channel properties, the files that are not in local caches are also exposed to a possible QoS degradation, due to the backhaul capacity limitations.

Finally, we aim at finding a *matching* $\eta : \mathcal{M} \rightarrow \mathcal{N}$ that maximizes the utility $U_{i,m}$, by considering the limitations on the backhaul capacity and storage size. Essentially, this yields the following optimization problem:

$$\underset{\eta\,:\,(i,m)\in\eta, f\in\mathcal{D}_i}{\arg\max} \ \sum_{i\in\mathcal{N}} \sum_{m\in\mathcal{L}_i} U_{i,m}(t_{i,m}^{IN}, t_{i,m}^{OUT}, \hat{\tau}_{i,m}^{max}), \tag{7}$$

$$\text{s.t.,} \quad D_i \cdot s \leq K_i, \ \forall i \in \mathcal{N}. \tag{8}$$

In terms of complexity, solving the UE-SBS association using classical optimization techniques is an NP-hard problem, which depends on the number of SBSs and UEs in the network [17]. Such an exponential complexity makes a centralized approach intractable, especially in dense network deployments in which the number of UEs and SBSs significantly grows. As a result, solving the UE-SBS association problem in (7) mandates a decentralized approach in which UEs and SBSs autonomously decide on the UE-SBS association based on their caching capabilities. The formulation and implementation of such a decentralized solution are discussed in the following section.

B. Matching game formulation

For solving the SBS-UE association problem in (7), one suitable framework is that of *matching theory* [17]. Matching theory provides a computationally tractable set of tools for solving a combinatorial problem such as (7). 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* $\eta : \{\mathcal{M} \cup \mathcal{N}\} \rightarrow \{\mathcal{M} \cup \mathcal{N}\}$*, such that:*

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

Specifically, we consider a one-to-many matching that assigns to each UE $m \in \mathcal{M}$, an SBS $i = \eta(m)$, $i \in \mathcal{M}$, and to each SBS $i \in \mathcal{M}$, a set of UEs $\eta(i) \subset \mathcal{M}$, such that $|\eta(i)| \leq q_i$, where q_i denotes a maximum quota. Both UEs and SBSs define

[2]For example, an SBS can estimate the incoming users based on mobility tracking [14], [15], or based on the received signal strength indicators that a UE periodically broadcasts [16].

Algorithm 1: UE-SBS Cell Association Algorithm.

Data: Each UE m is initially associated to a randomly selected SBS j, $(j, m) \in \eta'$.
Result: Convergence to a stable matching η.
Phase I - Incoming UE discovery;
• At time t: each SBS i tracks the CSI feedbacks $g_{i,m}(t)$ of the UE m in the vicinity;
• Each SBS estimates the arrival time $t_{i,m}^{IN}$ and $t_{i,m}^{OUT}$ of user m;
• At time $t_{i,m}^{IN}$: UE m notifies $\mathcal{F}_m$ to SBS i, and the utility $U_{i,m}$ is updated;
Phase II - UE-SBS matching proposal ;
for *all the discovered UEs* m **do**
  • Incoming users m are sorted by $\succ_i$;
  • SBS i sends a proposal to the UE i at the top of the preference list and notifies B_i;
  • UE m computes the data rate $C_{i,m}(t)$ and sorts the SBSs by $\succ_m$;
  if $i \succ_m j$ **then**
    • UE m accepts the proposal of SBS i;
    • SBS i will start the transmissions at $t_{i,m}^{IN}$.
  else
    • UE m refuses the proposal, and SBS i sends a proposal to the next preference.
  end
end
• At time $t_{i,m}^{IN}$: UE m gets associated to SBS m, $\mathcal{L}_i \leftarrow \mathcal{L}_i \cup \{m\}$.
Phase III - Cache management;
• During $[\hat{t}_{i,m}, t_{i,m}^{IN}]$, the SBSs update the cached data sets based on the file popularity;
• Caching procedure continues until the memory capacity is reached. Beyond that point, least popular files are systematically dropped.

individual preference relations $\succ$, that are complete, reflexive, and transitive binary relation between the players in $\mathcal{M}$ and $\mathcal{N}$. Accordingly, the preference profile of an SBS i, over the set of UEs $\mathcal{M}$ is defined by an ordered list $\Pi(i) = \{m, n, \dots\}$, denoting that SBS i prefers to service UE m, rather than UE j, or briefly $m \succ_i n$. Similarly, $\Pi(m) = \{i, j, \dots\}$ represents the preferences of UE m over the set of SBS $\mathcal{N}$, indicating that that UE m prefers being associated to SBS i, rather than to SBS j, i.e., $i \succ_m j$.

When defining a preference for an SBS, a UE has no knowledge of the files stored at the SBS side. As a result, a UE can only define a preference based on the properties of the SBSs' transmitted signals. Hence, for any UE m, we propose a preference relation $\succ_m$ defined over the set of SBSs $\mathcal{N}$, based on the transmission data rate of SBS i:

$$i \succ_m j \Leftrightarrow C_{i,m}(t) > C_{j,n}(t). \tag{9}$$

Next, we define an analogous preference relation $\succ_i$ for any SBS i over the set of UE $\mathcal{M}$, based on the utility in (6). Such a preference relation accounts for a UE's time of arrival and departure from cell i, and the amount of files requested by UE m, that are currently available at the SBS side:

$$m \succ_i n \Leftrightarrow U_{i,m}(t_{i,m}^{IN}, t_{i,m}^{OUT}, \hat{\tau}_{i,m}^{max}) > U_{i,n}(t_{i,n}^{IN}, t_{i,n}^{OUT}, \hat{\tau}_{i,m}^{max}) \tag{10}$$

To solve the problem in (7) 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 become associated with the UE, for which the requested files are likely to be locally available, in $\mathcal{D}_i$, or can be retrieved through a backhaul of capacity B_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.

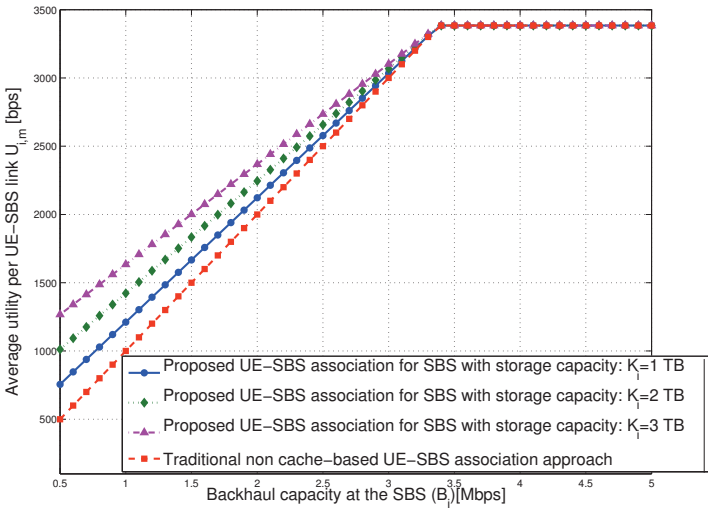

Fig. 2. Average utility per UE-SBS link as a function of the backhaul capacity B_i, for different storage capacities $K_i = \{1, 2, 3\}$, $M = 120$ UEs , $N = 120$ SBSs, $\nu_m = 1$ m/s.

C. Proposed game solution

In order to find a UE-SBS matching and the problem in (7), we propose a new approach, shown in Algorithm 1, inspired by the deferred acceptance scheme proposed by Gale and Shapley for the stable marriage problem [17]. Algorithm 1 is composed of three main phases: incoming UE discovery, UE-SBS matching proposal, and cache management. Initially, each UE is associated to a randomly selected SBS j[3]. Then, each SBS i discovers the incoming UE $i \in \mathcal{M}$ in the vicinity, using standard tracking techniques such as in [2]. At this stage, SBS i learns the time of arrival $t_{i,m}^{IN}$, and the time left for caching additional contents. Next, based on the current matching, UEs and SBSs update their respective utilities and individual preferences over one another. In the second phase, each SBS sends a proposal to the most preferred UE m, by notifying the available backhaul capacity B_i. Upon receiving a proposal, UE m updates its preference list and accepts the request of SBS i only if it is the most preferred SBS, among the available ones. Otherwise, if rejected, SBS m proposes to the next UE in its preference list. Both UEs and SBSs periodically update their respective utilities and preferences according to the current utilities and ensure that they are associated to their respective first preference.

In order to study the stability of the proposed matching, we use the stability concept used by Gale and Shapley [17], by adapting it to the problem in (7):

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.*

For our proposed game, the scheme shown in Algorithm 1 will reach a stable matching as follows:

[3]Equivalently, the UE can be initially associated to the closest SBS.

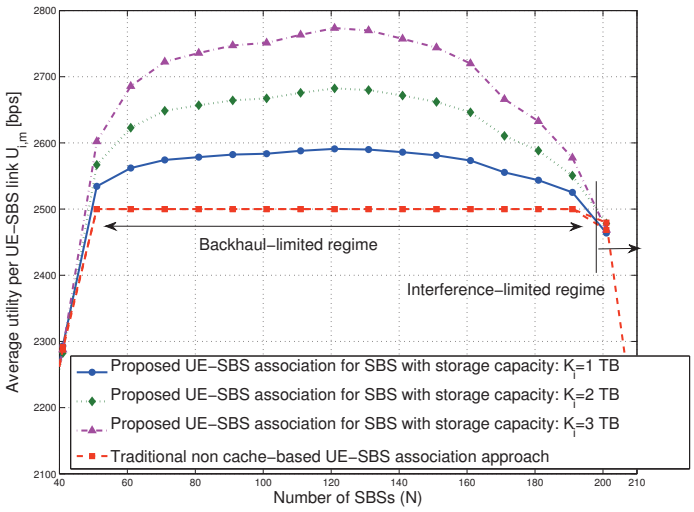

Fig. 3. Average utility per UE-SBS link as a function of the number of SBS N, for different storage capacities $K_i = \{1, 2, 3\}$ TB, $M = 100$ UEs , $B_i = 2.5$ Mbps, $\nu_m = 1$ m/s.

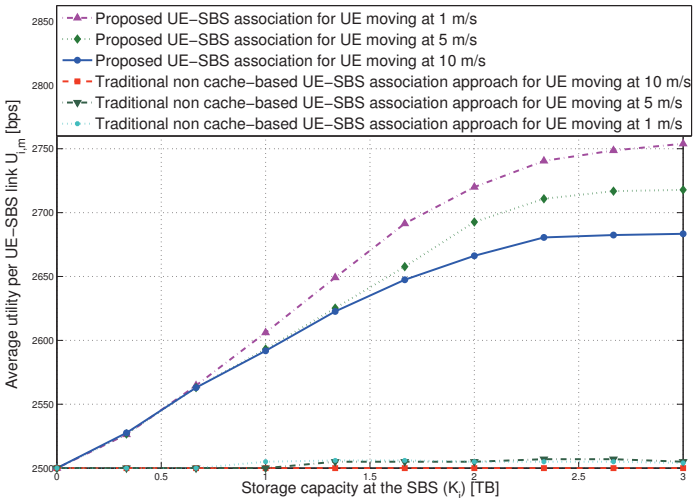

Fig. 4. Average utility per UE-SBS link as a function of the storage capacity per SBS, K_i, for different UEs' speeds $\nu_m = \{1, 5, 10\}$ m/s, $M = 120$ UEs, $B_i = 2.5$ Mbps.

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 [17].*

IV. Simulation Results

For our simulations, we consider a single cell of a macro-cellular network, modeled as a Manhattan map of 500×500 m with a bandwidth of 20 MHz. In this cell, M UEs and N SBSs are uniformly deployed. The UE's speed is chosen as i.i.d in the interval $\nu_m = [1, 10]$ m/s. The transmit power of each SBS i is $p_i = 33$ dBm and the assigned bandwidth per UE is $w_{i,m} = 720$ KHz. Transmissions are affected by distance dependent path loss, with path loss exponent 3, and shadowing according to 3GPP specifications [18]. The files in $\mathcal{F}$ have a size of $s = 2$ KB. The file requests follow a Zipf distribution with parameter $\psi = 0.4$. Each UE requests $D_i = 1500$ files, out of a set of $|\mathcal{F}| = 1.5 \cdot 10^9$ files. Each SBS $i \in \mathcal{N}$ has a memory capacity chosen from an interval $K_i = [0.2, 3]$ TB. Similarly, the backhaul capacity is chosen from an interval $B_i = [0.5, 5]$ Mbps.

Prior to the performance evaluation, we considered a training phase of the duration of 600 seconds, in which each SBS has downloaded a set of popular files directly through the backhaul, according the Zipv file popularity distribution.

For comparisons, we consider a traditional non cache-based approach, in which the SBSs accommodate the UE's data requests by downloading the respective files directly from the backhaul whose capacity is given by B_i. In practice, the utility of such an approach is still expressed by (6), while setting $\hat{\tau}_{i,m}^{max} = t_{i,m}^{IN}$, since no files are kept at the SBSs' side.

Figure 2 shows the average utility per UE-SBS link as a function of the backhaul capacity B_i, for different storage capacities K_i at the SBSs, in a network with $N = 120$ SBSs, and $M = 120$ UEs. Figure 2 shows that the proposed caching strategy is mostly beneficial during a regime of limited backhaul capacity (i.e., $B_i \leq 3.4$ Mbps). In fact, the proposed approach yields a utility gain which is proportional to the probability of having the UE's files in the serving SBS' storage unit. For example, Figure 2 shows that the performance gap between the proposed approach and a non cache-based association scheme is 21%, for SBSs with a backhaul capacity of $B_i = 2$ Mbps and storage units of 3 TB. Finally, the gains stemming from caching saturate when the backhaul capacity no longer represents a bottleneck for QoS delivery (i.e., $B_i \geq 3.4$ Mbps). Therefore, Figure 2 demonstrates that the proposed cache-based approach yields significant utility gains by exploiting local content availability, notably in networks with a limited-capacity backhaul.

Figure 3 shows the average utility per UE-SBS link as a function of the number of SBSs N, for different storage capacities, in a network with $M = 100$ UEs, and a backhaul capacity of $B_i = 2.5$ Mbps. Figure 3 shows that, for all the considered approaches, the average utility of an UE grows with the probability of being serviced by a nearby SBS, yielding higher SINR. Note that, even when higher transmission data rates are possible, delivering the UE's files in a traditional non cache-based approach is ultimately limited by the backhaul capacity, as per constraint (2). In such a backhaul-limited regime (i.e., $45 \leq N \leq 199$ SBSs), locally cached files can be transmitted at data rates larger than the backhaul capacity. For example, Figure 3 shows that the performance gains of the proposed cache-based approach increase with the storage capacity K_i, reaching up to 11% and 6% relative to a non cache-based approach, respectively for $K_i = 3$ TB, and $K_i = 1$ TB, in a network with $N = 120$ SBSs. Finally, for all the considered techniques, the utility eventually decreases as the received interference grows with the size of the small cell tier. In summary, Figure 3 shows that by locally caching UE's file, the SBSs are able to overcome the backhaul capacity limitations and improve the QoS delivery, yielding gains that grow linearly with the SBSs' storage capacity K_i.

In Figure 4, we evaluate the average utility per UE-SBS link as a function of the caching capabilities at the SBS sides, for different UEs' speeds $\nu_m = \{1, 5, 10\}$ m/s. Figure 4 shows that, for the considered cases, the UE-SBS utility grows linearly with the storage capacity K_i, while depending on the average time spent by a UE within an SBS' coverage. In fact, the longer

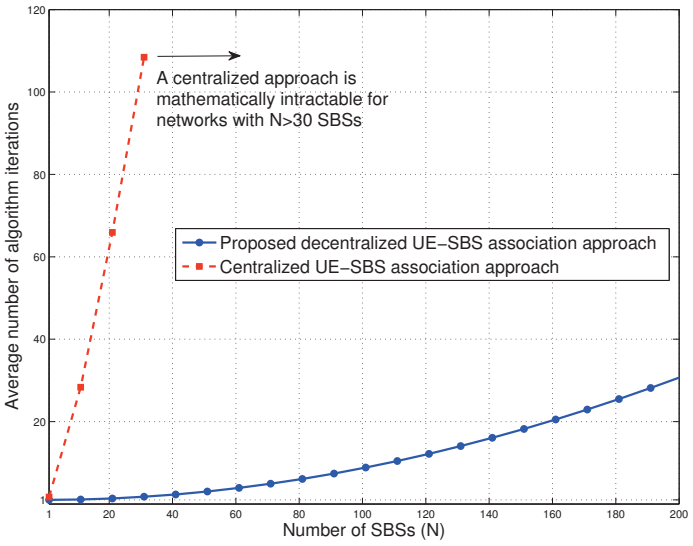

Fig. 5. Average number of algorithm iteration as a function of the network size N, $M = 120$ UEs, $B_i = 2.5$ Mbps, $\nu_m = 1$ m/s.

a UE is associated to an SBS, the larger is the amount of cached data that an SBS can deliver. For example, the gain of the proposed solution with respect to a traditional UE-SBS association approach is 9% and 7.1%, respectively for a UE's speed of $\nu_m = 1$ m/s and $\nu_m = 10$ m/s, for SBSs with a storage capacity $K_i = 2$ TB. In a nutshell, Figure 4 shows that the proposed cache-aware user association approach can enhance the UE's QoS in a wide range of UEs' mobility patterns.

In Figure 5, we show the average number of algorithm iterations (Phase II of Algorithm 1) required at each SBS to converge to a stable matching, as a function of the number of SBSs in the network. Figure 5 shows that the complexity of a decentralized approach depends on the number of SBSs that can service a given UE. For instance, the average number of algorithm iterations is 17, for a network with $M = 120$ UEs and $N = 120$ SBSs, while it grows up to 20 for a larger network with $N = 160$ SBSs. Figure 5 also shows the number of iterations required to devise an optimal UE-SBS matching in a centralized fashion. Here, although the deferred acceptance scheme has a polynomial complexity, a centralized approach requires a brute-force search, whose complexity grows exponentially with N [19]. As a result, a centralized solution is computationally intractable for networks larger than $N = 35$ SBSs. In summary, Figure 5 shows that the proposed distributed approach converges to a stable matching by performing a reasonable number of algorithm iterations at each SBS.

V. Conclusions

In this paper, we have presented a novel cache-aware UE-SBS association approach for wireless small cell networks. The proposed scheme enables each SBS to select the UEs to be serviced, by accounting for the local availability of cached files, as well as the backhaul and interference limitations. 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, based on the deferred acceptance scheme, that enables the UEs and SBSs to generate a list of preferences that are respectively based on the transmission capacity and a utility that accounts for the SBSs' data storage capabilities and the UE's mobility pattern. We have shown that, by using the proposed algorithm, the SBSs and the UEs reach a stable matching in a reasonable number of iterations. Simulation results have shown that, by exploiting local files availability at the SBSs, the proposed cache-based solution enables the SBSs to overcome the limitations of a congested backhaul, and yield significant gains in terms of data delivered to the UEs, reaching up to 21%, with respect to a traditional UE-SBS association approach with no cache considerations.

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] M. Kryder and C. S. Kim, "After hard drives - what comes next?" *IEEE Transactions on Magnetics*, vol. 45, no. 10, pp. 3406–3413, Oct 2009.

[4] 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.

[5] S. Woo, E. Jeong, S. Park, J. Lee, S. Ihm, and K. Park, "Comparison of caching strategies in modern cellular backhaul networks," in *In Proc. of the Int'l Conf. on Mobile Systems, Applications, and Services (MobiSys)*, 2013, pp. 319–332.

[6] 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.

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

[8] N. Golrezaei, A. Dimakis, and A. Molisch, "Wireless device-to-device communications with distributed caching," in *In Proc. of IEEE Int'l Symp. on Information Theory Proceedings (ISIT)*, July 2012, pp. 2781–2785.

[9] 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.

[10] V. Siris and M. Anagnostopoulou, "Performance and energy efficiency of mobile data offloading with mobility prediction and prefetching," in *In Proc. of IEEE Int'l Symp. on World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2013, pp. 1–6.

[11] E. Bastug, M. Bennis, and M. Debbah, "Social and spatial proactive caching for mobile data offloading," in *In Proc. of IEEE ICC Workshop on Small Cell and 5G Networks (SMALLNETS)*, June 2014, p. (to appear).

[12] K. Hamodouche, W. Saad, and M. Debbah, "Many-to-many matching games for proactive social caching in small cell networks," in *In Proc. of 12th Int'l Symp. on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, May 2014, pp. 1–5.

[13] F. Bai, N. Sadagopan, and A. Helmy, "Important: a framework to systematically analyze the impact of mobility on performance of routing protocols in adhoc networks," in *IEEE INFOCOM conference*, 2003, pp. 825 –835.

[14] X. Chen, F. Meriaux, and S. Valentin, "Predicting a user's next cell with supervised learning based on channel states," in *In Proc. of IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, June 2013, pp. 36–40.

[15] Y. Yuan, Y. Tang, and C. Lin, "A novel mobility prediction mechanism in heterogeneous networks," in *In Proc. of Int'l Conf. on Communications and Mobile Computing (CMC)*, vol. 3, April 2010, pp. 536–540.

[16] J. Kim, S. Kim, N. Y. Kim, J. Kang, Y. Kim, and K.-T. Nam, "A novel location finding system for 3gpp lte," in *In Proc. of IEEE Int'l Symp. on Personal, Indoor and Mobile Radio Communications*, Sept 2009.

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

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

[19] T. Roughgarden, "Computing equilibria: A computational complexity perspective," *Stanford University*, vol. 1, no. 09, Jan 2009.