Exploiting Caching and Multicast for 5G Wireless Networks

Konstantinos Poularakis, George Iosifidis, Vasilis Sourlas, *Member, IEEE*, and Leandros Tassiulas, *Fellow, IEEE*

***Abstract*—The landscape towards 5G wireless communication is currently unclear, and, despite the efforts of academia and industry in evolving traditional cellular networks, the enabling technology for 5G is still obscure. This paper puts forward a network paradigm towards next-generation cellular networks, targeting to satisfy the explosive demand for mobile data while minimizing energy expenditures. The paradigm builds on two principles; namely *caching* and *multicast*. On one hand, caching policies disperse popular content files at the wireless edge, e.g., pico-cells and femto-cells, hence shortening the distance between content and requester. On other hand, due to the broadcast nature of wireless medium, requests for identical files occurred at nearby times are aggregated and served through a common multicast stream. To better exploit the available cache space, caching policies are optimized with concerns on multicast transmissions. We show that the multicast-aware caching problem is NP-Hard and develop solutions with performance guarantees using randomized-rounding techniques. Trace-driven numerical results show that in presence of massive demand for delay tolerant content, combining caching and multicast can indeed reduce energy costs. The gains over existing caching schemes are 19% when users tolerate delay of three minutes, increasing further with the steepness of content access pattern.**

***Index Terms*—Content Caching, Multicast Delivery, Network Optimization, 5G Wireless Networks.**

I. Introduction

A. Motivation

We are witnessing an unprecedented worldwide growth of mobile data traffic that is expected to continue at an annual rate of 45% over the next years, reaching 30.5 exabytes per month by 2020 [2]. To handle this "data tsunami", the emerging 5^{th} generation (5G) systems need to improve the network performance in terms of energy consumption, throughput and user experienced delay, and at the same time make a better use of the network resources such as wireless bandwidth and backhaul link capacity. Two candidate solutions that have been investigated are *caching* and *multicast*.

On the first issue, there is an increasing interest for *in-network caching* architectures where operators cache popular content files at the Evolved Packet Core (EPC) or at the Radio Access Network (RAN), e.g., in dedicated boxes or at the cellular base stations. The common denominator is that they distribute storage resources near the end-user (rather than stored in data centers). In the context of heterogeneous cellular networks (HCNs) [3], caches can be installed at small-cell base stations (SBSs), e.g., pico-cells and femto-cells, targeting to offload traffic from the collocated macro-cell base station (MBS) [4]. Measurement studies have revealed up to 66% reduction in network traffic by using caching in 3G [5] and 4G [6] networks. Meanwhile, the wireless industry began to commercialize systems that support caching with examples including Altobridge's "Data at the Edge" solution [7], Nokia Siemens Networks' liquid application [8] and Saguna Networks' Open RAN platform [9].

On the second issue, many operators take advantage of *multicast* to efficiently utilize the available bandwidth of their networks in delivering the same content to multiple receivers [10]. For example, multicast is often used for delivering sponsored content, e.g., mobile advertisements in certain locations, downloading news, stock market reports, weather and sports updates [11]. Meanwhile, multicast has been incorporated in 3GPP specifications in which the proposed technology for LTE is called Evolved Multimedia Broadcast and Multicast Services (eMBMS) [12]. Commercial examples of eMBMS are Ericsson and Qualcomm LTE Broadcast solutions [13], [14]. This technology can be used across multiple cells where the transmission across them is synchronous using a common carrier frequency. Hence, multicast consumes a subset of the radio resources needed by a unicast service. The remaining resources can be used to support transmissions towards other users, thus enhancing network capacity.

Current proposals from academia and industry consider caching and multicast independently one from the other and for different purposes. On one hand, caching is used to shift traffic from peak to off-peak hours by exploiting the periodic pattern of traffic generation. This is realized by filling the caches with content during off-peak hours (e.g., nighttime), and serving requests for the stored content by the caches during peak-time (e.g., daytime). On other hand, multicast is used to reduce energy and bandwidth consumption by serving concurrent user requests for the same content via a single point-to-multipoint transmission instead of many point-to-point (unicast) transmissions.

Intuitively, caching should be effective when there is enough *content reuse*; i.e., many recurring requests for a few content files appear over time. Multicast should be effective when there is significant *concurrency* in accessing information across

Part of this work appeared in the proceedings of IEEE Wireless Communications and Networking Conference (WCNC), pp. 2300-2305, April 2014 [1]. This work was supported partly by the EC through the FP7 project FLEX (no. 612050), the Marie Curie project INTENT (grant no. 628360) and by the National Science Foundation Graduate Research Fellowship Program (grant no. CNS-1527090).

K. Poularakis is with the Dept. of Electrical and Computer Engineering, University of Thessaly, Greece (e-mail: kopoular@uth.gr). G. Iosifidis and L. Tassiulas are with the Electrical Engineering Department & Institute for Network Science, Yale University, USA (e-mail: {georgios.iosifidis, leandros.tassiulas}@yale.edu). V. Sourlas is with the Electronic & Electrical Engineering Department, University College London, UK (e-mail: v.sourlas@ucl.ac.uk).

users; i.e., many users concurrently generate requests for the same content file. Such scenarios are more common during crowded events with a large number of co-located people that are interested in the same contents, e.g., during sporting games, concerts and public demonstrations with often tens of thousand attendees [15], [16]. In next generation 5G systems where the demand for mobile data is often massive, and a variety of new services such as social networking platforms and news services employ the one-to-many communication paradigm, e.g., updates in Tweeter, Facebook, etc, it is expected that multicast will be more often applied.

Clearly, it is of paramount importance to design caching and multicast mechanisms for servicing the mobile user requests with the minimum possible energy expenditures. For a given anticipated content demand, the caching problem asks for determining in which caches to store each content file. This becomes more challenging in HCNs where users are covered by multiple base stations and hence content can be delivered to requesters through multiple network paths [17]-[20]. Also, the caching problem differs when multicast is employed to serve concurrent requests for the same content file. Compared to unicast communication, multicast incurs less traffic as the requested file is transmitted to users only once, rather than with many point-to-point transmissions. Hence, the caching problem needs to be revisited to effectively tackle the following questions: *Can caching and multicast be combined to reduce energy costs of an operator? If yes, what is the condition and where the gains come from?*

B. Methodology and Contributions

In order to answer the above questions, we consider a HCN model that supports caching and multicast for the service of the mobile users. Requests for the same content file generated during a short-time window are aggregated and served through a single multicast transmission when the corresponding window expires (*batching multicast* [21]). To ensure that the user experienced delay will be limited, the duration of this window should be as small as possible. For example, users may tolerate a very small start-up delay for video streaming applications, whereas larger delay may be acceptable for downloading news, stock market reports, weather and sports updates. The multicast stream can be delivered either by a SBS that is in communication range with the requesters in case that the respective file is available in its cache, or by the MBS which has access to the entire file library through a backhaul link. Clearly, a MBS multicast transmission can satisfy requests generated within the coverage areas of *different* SBSs that have not cached the requested file. However, it typically induces higher energy cost than a SBS, since the distance to the receiver is larger and it also needs to fetch the file via its backhaul link.

First, we demonstrate through simple examples how multicast affects the efficiency of caching policies. Then, we introduce a general optimization problem (which we name MACP) for devising the multicast-aware caching policy that minimizes the overall energy cost. Our model explicitly takes into consideration: (i) the heterogeneity of the base stations which may have different cache sizes and transmission cost parameters (e.g., due to their different energy consumption profile [22]), and (ii) the variation of request patterns of the users which may ask for different content files with different intensity. We formally prove the intractability of the MACP problem by reducing it to the set packing problem, which is NP-Hard [23]. Following that, we develop an algorithm with performance guarantees under the assumption that the capacity of the caches can be expanded by a bounded factor. This algorithm applies linear relaxation and randomized rounding techniques. Then, we describe a simple heuristic solution that can achieve significant performance gains over existing caching schemes.

Using traffic information from a crowded event with over fifty thousand attendees [15], we investigate numerically the impact of various system parameters, such as delay tolerance of user application, SBS cache sizes, base station transmission costs and demand steepness. We find that the superiority of multicast-aware caching over traditional caching schemes is highly pronounced when: (i) the user demand for content is high and (ii) the user requests for content are delay-tolerant. The gains are 19% when users tolerate delay of three minutes, increasing further with the steepness of content access pattern. Our main technical contributions are as follows:

- *Multicast-aware caching problem (MACP)*. We propose a novel caching paradigm and an optimization framework building on the combination of caching and multicast techniques in HCNs. This is important, as content delivery via multicast is part of 3GPP standards and gains increasing interest.
- *Complexity Analysis*. We prove the intractability of the MACP problem by reducing it to the set packing problem [23]. That is, we show that MACP is NP-Hard even to approximate within a factor of $O(\sqrt{N})$, where N is the number of SBSs in a macro-cell. This result reveals how the consideration of multicast further perplexes the caching problem.
- *Solution algorithms*. Using randomized rounding techniques, we develop a multicast-aware caching algorithm that achieves performance guarantees under the assumption that the capacity constraints can be violated in a bounded way. Also, we describe a simple-to-implement heuristic algorithm that provides significant performance gains compared to the existing caching schemes.
- *Performance Evaluation*. Using system parameters driven from real traffic observations in a crowded event, we show the cases where the next generation HCN systems should optimize caching with concerns on multicast delivery. The proposed algorithms yield significant energy savings over existing caching schemes, which are more pronounced when the demand is massive and the user requests can be delayed by three minutes or more.

The rest of the paper is organized as follows: Section II describes the system model and defines the MACP problem formally. In Section III, we show the intractability of the problem and present algorithms with performance guarantees and heuristics. Section IV presents our trace-driven numerical

results, while Section V reviews our contribution compared to the related works. We conclude our work in Section VI.

II. System Model and Problem Formulation

In this section we introduce the system model, we provide a motivating example that highlights how multicast affects the efficiency of caching policies and, finally, we formally define the multicast-aware caching optimization problem.

A. System Model

We study the downlink operation of a heterogeneous cellular network (HCN) like the one depicted in Fig. 1. A set $\mathcal{N}$ of N small-cell base stations (SBSs), e.g., pico-cells and femto-cells, are deployed within a macro-cell coexisting with the macro-cell base station (MBS). The MBS can associate to any user in the macro-cell, while SBSs can associate only to users lying in their coverage areas. Each SBS n is equipped with a cache of size $S_n \geq 0$ bytes which can be filled in with content files fetched from the core network through a backhaul link. Since the SBS backhaul links are usually of low-capacity, e.g., often facilitated by the consumers' home networks such as Digital Subscriber Line (DSL) [24], they cannot be used to download content on demand to serve users. Instead, they are only used to periodically refresh the content stored in the caches [17]-[20]. In contrast, the backhaul link of the MBS is of sufficient capacity to download the content requested by users. Therefore, a user can be served either by the MBS or by a covering SBS provided that the latter has cached the requested content file.

The user demand for a set of popular files and within a certain time period is assumed to be known in advance, as in [17]-[20], [25]-[28] which is possible using learning techniques [29], [30]. Let $\mathcal{I}$ indicate that collection of files, with $I = |\mathcal{I}|$. For notational convenience, we consider all files to have the same size normalized to 1. This assumption can be easily removed as, in real systems, files can be divided into blocks of the same length [17], [27]. The SBS coverage areas can be overlapping in general, but each user can associate to only one SBS according to a best-server criterion (e.g., highest SNR rule). We denote with $\lambda_{ni} \geq 0$ (requests per time unit) the average demand for file i generated by the users associating to SBS n. Also, $\lambda_{0i} \geq 0$ denotes the average demand for file i generated by users who are not in the coverage area of any of the SBSs[1] .

The operator employs multicast (such as eMBMS) for transmission of the same content to multiple receivers. In this case, user requests within a short-time window are aggregated and served through a single multicast stream when the corresponding window expires. We denote with d (time units) the time duration of this window, also called *multicast period*. Clearly, it is important to identify which SBSs receive file requests within the multicast period. To this end, we denote with p_{ni} the probability that *at least one* request for file i is generated by

[1] Notice that the current practice of operators is to deploy SBSs to certain areas with high traffic. Hence, other less congested areas may be covered only by the MBS.

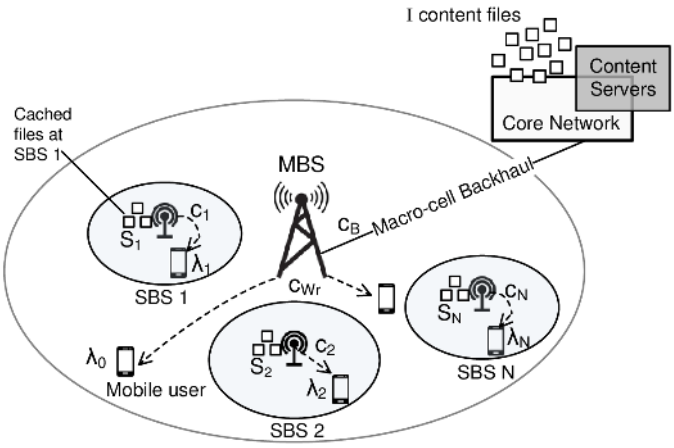


Fig. 1. Graphical illustration of the discussed model. The circles represent the coverage areas of the MBS and the SBSs. To ease presentation, the backhaul links of the SBSs are not depicted.

users associating to SBS n (*area* n)[2] during a multicast period. Similarly, p_{0i} indicates the respective probability for the users that are not in the coverage area of any of the SBSs (*area* n_0). For example, if the number of requests for file i associated to SBS n follows the *Poisson* probability distribution with rate parameter λ_{ni}, it becomes:

$$p_{ni} = 1 - e^{-\lambda_{ni} d}. \tag{1}$$

We then define the collection of all subsets of areas excluding the empty set as follows:

$$\mathcal{R} = (r : r \subseteq \mathcal{N} \cup n_0, r \neq \emptyset). \tag{2}$$

We also define with q_{ri} the probability that at least one request for the file $i \in \mathcal{I}$ is generated within each one of the areas $r \in \mathcal{R}$ during a multicast period. For example, if requests are generated *independently* among different areas, then the following equation holds:

$$q_{ri} = \prod_{n \in r} (p_{ni}) \cdot \prod_{n \notin r} (1 - p_{ni}). \tag{3}$$

Our model is generic, since it allows for any probability distributions p_{ni} and q_{ri}.

The power consumption is typically higher for MBS compared to SBSs, while it depends on the channel conditions and the distance between transmitter and receiver. Let P_n (watts) denote the minimum transmission power required by MBS for transmitting a file to a user in area n. According to SINR criteria this is given by [31], [32]:

$$P_n = P_s - G_n - G_m + L_{mn} + \Psi_n + 10 \log_{10} M_n. \tag{4}$$

In the above equation P_s is the receiver sensitivity for the specific service, parameter G_n represents the antenna gain of a user in area n and G_m represents the antenna gain of MBS. L_{mn} is the path loss between MBS and a user in area n which depends on the channel characteristics and the distance between MBS and user, Ψ_n is the shadow component derived

[2] With a slight abuse of notation we use the same index for base stations and their covering areas.

by a lognormal distribution and M_n is the number of resource blocks assigned to a user in area n. A similar definition holds for the transmission power of the SBSs.

We consider the more general case in which both the MBS and the SBSs employ multicast. Namely, a multicast transmission of SBS $n \in \mathcal{N}$ satisfies the requests for a cached file generated in area n, while a MBS multicast transmission satisfies the requests generated in different areas (and requests in area n_0) where the associated SBSs have not cached the requested file. Let n^* denote the area that requires the highest transmission power in a subset $r \in \mathcal{R}$, i.e., $n^* = \text{argmax}_{n \in r} P_n$. Then, to multicast a file to all the users in r, the power consumption required by MBS is given by [33]:

$$c_{Wr} = P_{n^*} = max_{n \in r} P_n. \quad (5)$$

Similarly, c_n denotes the power consumption required by SBS n for multicasting a cached file to its local users, where in general $c_n \leq c_{Wr}$, $\forall n, r$. Finally, we denote with $c_B \geq 0$ the power consumed for transferring a file via the backhaul link of the MBS [34].

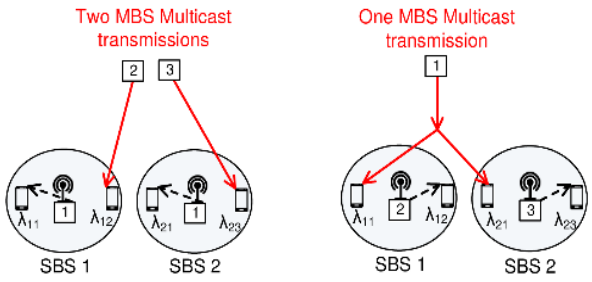
Before we introduce formally the problem, let us provide a simple example that highlights how the consideration of multicast transmissions perplexes the caching problem.

B. Motivating Example

Let us consider a multicast service system with two SBSs ($\mathcal{N} = \{1, 2\}$) and three files ($\mathcal{I} = \{1, 2, 3\}$). Each SBS can cache at most one file because of its limited cache size. We set $c_B + c_{Wr} = 1$ $\forall r$, $c_1 = c_2 = 0$ and $d = 1$. We also set the generation of request to follow a Poisson probability distribution. Finally, we set $\lambda_{11} = 0.51$, $\lambda_{12} = 0.49$, $\lambda_{13} = 0$, $\lambda_{21} = 0.51$, $\lambda_{22} = 0$, and $\lambda_{23} = 0.49$, which imply that $p_{11} = 0.3995$, $p_{12} = 0.3874$, $p_{13} = 0$, $p_{21} = 0.3995$, $p_{22} = 0$ and $p_{23} = 0.3874$ (cf. equation (1)).

In a conventional system, each user request is served via a point-to-point unicast transmission. It is well known that placing the most popular files with respect to the local demand in each cache is optimal (in terms of the overall energy cost) in this setting. Hence, the optimal caching policy places file 1, which is the most popular file, to both SBS caches. By applying the above caching policy to the multicast service system that we consider here, all the requests for file 1 will be satisfied by the accessed SBSs at zero cost. The requests within SBS 1 for file 2 and the requests within SBS 2 for file 3 will be served by the MBS with $c_B + c_{Wr} = 1$ cost each (Fig. 2(a)). Assuming independent generation of requests, the total energy cost will be: $(c_B + c_{W1}) \cdot p_{12} \cdot (1 - p_{23}) + (c_B + c_{W2}) \cdot (1 - p_{12}) \cdot p_{23} + (c_B + c_{W1} + c_B + c_{W2}) \cdot p_{12} \cdot p_{23} = 0.7747$, where in the last term the cost is 2 instead of 1 because two *different* files are requested for download and thus two MBS transmissions are required for serving the requests.

However, if we take into consideration the fact that the user requests are aggregated and served via multicast transmissions every $d = 1$ time unit, then *the optimal caching policy changes*; it places file 2 to SBS 1 and file 3 to SBS 2. In this case, all the requests for file 1 will be served by the MBS via a single multicast transmission of cost $c_B + c_{Wr} = 1$ (Fig. 2(b)). The requests for the rest files will be satisfied by the accessed SBSs at zero cost. Hence, the total energy cost will be: $(c_B + c_{W1}) \cdot p_{11} \cdot (1 - p_{21}) + (c_B + c_{W2}) \cdot (1 - p_{11}) \cdot p_{21} + (c_B + c_{W12}) \cdot p_{11} \cdot p_{21} = 0.6394 < 0.7747$.



(a) Conventional caching. (b) Multicast-aware caching.

Fig. 2. An example with two SBSs and three files when (a) conventional and (b) multicast-aware caching is applied. The labels below SBSs represent the cached files. The labels on the top represent the files delivered by MBS.

This example demonstrated the inefficiency of conventional caching schemes that neglect multicast transmissions when determining the file placement to the caches. Novel schemes are needed that combine caching with multicast to better exploit the available cache space.

C. Problem Formulation

Let us introduce the binary optimization variable x_{ni} that indicates whether file $i \in \mathcal{I}$ is stored in the cache of SBS $n \in \mathcal{N}$ ($x_{ni} = 1$) or not ($x_{ni} = 0$). These variables constitute the *caching policy* of the operator:

$$\boldsymbol{x} = (x_{ni} \in \{0, 1\} : n \in \mathcal{N}, i \in \mathcal{I}). \quad (6)$$

We recall that the files will be transferred to the SBS caches through the backhaul links at the beginning of the period of study. Clearly, this operation consumes power. Power is also consumed by the caches themselves, with the exact value depending on the caching hardware technology, e.g., solid state disk (SSD) or dynamic random access memory (DRAM) [35]. We capture the above cost factors by the term c_S which denotes the power consumed by storing a file in a SBS cache amortized over a multicast period.

We also use the binary optimization variable y_{ri} to indicate whether a MBS multicast transmission will occur when a subset of areas $r \in \mathcal{R}$ receive requests for a file $i \in \mathcal{I}$ ($y_{ri} = 1$) or not ($y_{ri} = 0$). These variables constitute the *multicast policy* of the operator:

$$\boldsymbol{y} = (y_{ri} \in \{0, 1\} : r \in \mathcal{R}, i \in \mathcal{I}). \quad (7)$$

Clearly, a MBS multicast will occur ($y_{ri} = 1$) when at least one requester cannot find i in an SBS cache. This implies that *at least one* of the following conditions holds: (i) a request for file i is generated within an area that is not in the coverage area of any of the SBSs, i.e., $n_0 \in r$, or (ii) a request for file i is generated by a user associated to an SBS $n \in r \setminus n_0$, but the latter has not stored in its cache the requested file. Hence,

y_{ri} should satisfy the following inequalities:

$$y_{ri} \geq \mathbf{1}_{\{n_0 \in r\}}, \ \forall r \in \mathcal{R}, \ i \in \mathcal{I}, \tag{8}$$

$$y_{ri} \geq 1 - x_{ni}, \ \forall r \in \mathcal{R}, i \in \mathcal{I}, n \in r, \tag{9}$$

where $\mathbf{1}_{\{.\}}$ is the indicator function, i.e., $\mathbf{1}_{\{b\}} = 1$ iff condition b is true; otherwise $\mathbf{1}_{\{b\}} = 0$.

Let us now denote with $J_i(\boldsymbol{y})$ the energy cost for servicing the requests for file i that are generated within a multicast period, which clearly depends on the multicast policy $\boldsymbol{y}$ of the operator. For each subset of areas r that may generate requests for file i within a time period, a single MBS multicast transmission of cost $c_B + c_{Wr}$ occurs, if a requester cannot find i in an accessed SBS ($y_{ri} = 1$). In other case ($y_{ri} = 0$), all the requests are satisfied by the accessed SBSs, where the requests in area n incur cost c_n. Hence:

$$J_i(\boldsymbol{y}) = \sum_{r \in \mathcal{R}} q_{ri} \cdot \Big(y_{ri} \cdot (c_B + c_{Wr}) + (1 - y_{ri}) \cdot \sum_{n \in r} c_n \Big). \tag{10}$$

Table I summarizes the key notation used throughout the paper.

The *Multicast-Aware Caching Problem* (MACP) determines the caching and multicast policies that minimize the expected energy cost within a multicast period[3] :

$$\text{minimize}_{\boldsymbol{x},\boldsymbol{y}} \ \sum_{n \in \mathcal{N}} \sum_{i \in \mathcal{I}} (c_S \cdot x_{ni}) + \sum_{i \in \mathcal{I}} (J_i(\boldsymbol{y})), \tag{11}$$

$$\text{subject to: } (8), (9),$$

$$\sum_{i \in \mathcal{I}} x_{ni} \leq S_n, \ \forall n \in \mathcal{N}, \tag{12}$$

$$x_{ni} \in \{0, 1\}, \ \forall n \in \mathcal{N}, \ i \in \mathcal{I}, \tag{13}$$

$$y_{ri} \in \{0, 1\}, \ \forall r \in \mathcal{R}, \ i \in \mathcal{I}, \tag{14}$$

where the first term in the objective function is the caching cost, and the second is the servicing cost. Inequalities in (12) ensure that the total amount of data stored in a cache will not exceed its size. Constraints in (13), (14) indicate the discrete nature of the optimization variables.

MACP is an integer programming problem, and hence, is in general hard to solve. Also, its objective function in (11) has an exponentially long description in the number of SBSs N, since the summation in $J_i(\boldsymbol{y})$ is over all subsets $r \in \mathcal{R}$. As we formally prove in the next section, MACP is an NP-Hard problem.

TABLE I
KEY NOTATIONS

Symbol	Physical Meaning
n_0	Area that is out of coverage of all SBSs
n	SBS (area) belonging to the set $\mathcal{N}$
r	Subset of areas belonging to the collection $\mathcal{R}$
i	File belonging to the set $\mathcal{I}$
S_n	Cache capacity of SBS n
c_S	Energy cost for storing a file in a SBS cache
c_B	Energy cost for multicasting a file via MBS backhaul
c_{Wr}	Energy cost for multicasting a file from MBS to areas r
c_n	Energy cost for multicasting a file from SBS n
λ_{ni}	Average demand in area n for file i
d	Duration of multicast period
p_{ni}	Probability that requests for file i appear in area n within d
q_{ri}	Probability that requests for file i appear in areas r within d
x_{ni}	Caching decision for file i to SBS n
y_{ri}	Indicator of MBS multicast for serving file i in areas r
$J_i(\boldsymbol{y})$	Energy cost for servicing the requests for file i

III. Complexity and Solution Algorithms

In this section, we prove the high complexity of the MACP problem and present solution algorithms with performance guarantees and heuristics.

A. Complexity

In this subsection, we prove that the MACP problem cannot be approximated within any ratio better than the square root of the number of SBSs. The proof is based on a reduction from the well known NP-Hard set packing problem (SPP) [23]. In other words, we prove that SPP is a special case of MACP. Particularly, the following theorem holds:

Theorem 1. It is NP-Hard to approximate MACP within any ratio better than $O(\sqrt{N})$.

Theorem 1 is of high importance, since it reveals how the consideration of multicast transmissions further perplexes the caching problem. In order to prove Theorem 1 we will consider the corresponding (and equivalent) decision problem, called Multicast-Aware Caching Decision Problem (MACDP). Specifically:

MACDP: Given a set $\mathcal{N}$ of SBSs, a set $\mathcal{I}$ of files, the cache sizes $S_n \ \forall n \in \mathcal{N}$, the costs c_S, c_B, c_{Wr} and $c_n \ \forall r \in \mathcal{R}, n \in \mathcal{N}$, the multicast period d, the probabilities $q_{ri} \ \forall r \in \mathcal{R}, i \in \mathcal{I}$, and a real number $Q \geq 0$, *we ask the following question*: do there exist caching and multicast policies $\boldsymbol{x}, \ \boldsymbol{y}$, such that the value of the objective function in (11) is less or equal to Q and constraints (8),(9),(12),(13),(14) are satisfied?

The set packing decision problem is defined as follows:

SPP: Consider a finite set of elements $\mathcal{E}$ and a list $\mathcal{L}$ containing subsets of $\mathcal{E}$. We ask: do there exist k subsets in $\mathcal{L}$ that are pairwise disjoint?

Lemma 1. *SPP problem is polynomial-time reducible to the MACDP.*

Proof: Let us consider an *arbitrary* instance of the SPP decision problem and a *specific* instance of MACDP with $N = |\mathcal{E}|$ SBSs, i.e., $\mathcal{N} = \{1, 2, \ldots, |\mathcal{E}|\}$, $I = |\mathcal{L}|$ files, i.e., $\mathcal{I} = \{1, 2, \ldots, |\mathcal{L}|\}$, unit-sized caches ($S_n = 1 \ \forall n \in \mathcal{N}$), $c_S = 0$, $c_B + c_{Wr} = 1$ and $c_n = 0 \ \forall r \in \mathcal{R}, \ n \in \mathcal{N}$. Parameter d is any positive number, and the question is if we can satisfy all the user requests with energy cost $Q = 1 - \frac{k}{|\mathcal{L}|}$, where k is the parameter from the SPP. The important point is that we define the q_{ri} probabilities as follows:

$$q_{ri} = \begin{cases} 1/|\mathcal{L}|, & \text{if } r = \mathcal{L}(i) \\ 0, & \text{else} \end{cases} \tag{15}$$

where $\mathcal{L}(i)$ is the i^{th} component of the list $\mathcal{L}$. Notice that with the previous definitions, $\mathcal{L}(i)$ contains a certain subset

[3] We emphasize that our model is focused on the energy consumed for caching and transmitting data to users. Hence, other factors such as cooling [22] are left outside the scope of our study.

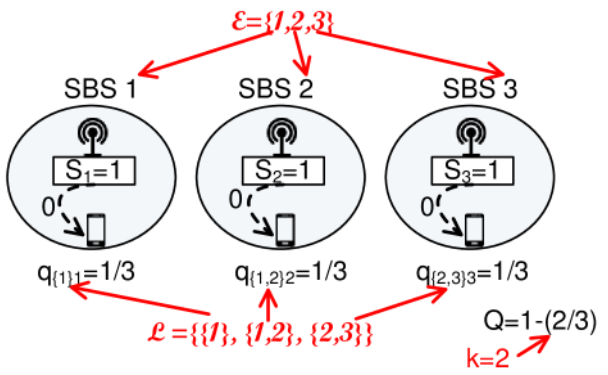


Fig. 3. An example of the reduction from SPP with $\mathcal{E} = \{1, 2, 3\}$, $\mathcal{L} = \{\{1\}, \{1, 2\}, \{2, 3\}\}$ and $k = 2$. In the MACDP instance there are $N = |\mathcal{E}| = 3$ SBSs and $I = |\mathcal{L}| = 3$ files. There is a solution to MACDP of cost $Q = 1 - \frac{2}{3}$ that places file 1 to SBS 1 and file 3 to SBSs 2 and 3. Accordingly, the solution to SPP picks $k = 2$ subsets: $\mathcal{L}(1) = \{1\}$ and $\mathcal{L}(3) = \{2, 3\}$.

of elements of $\mathcal{E}$. For the MACDP, under the above mapping, this corresponds to a subset of SBSs asking with a non-zero probability file i. Moreover, with (15) we assume that these probabilities are equal for all files $i \in \mathcal{I}$ and have value $1/|\mathcal{L}|$.

If the MBS serves all the requests, then the MACDP problem has a value (cost) of $c_B + c_{Wr} = 1$ (the worst case scenario). For each file i that the operator manages to serve completely through local caching at the SBSs, the operator reduces its cost by $(c_B + c_{Wr}) \cdot q_{ri} = 1/|\mathcal{L}|$. This reduction is ensured only if the file is cached in all the SBSs $n \in r$ for which $q_{ri} = 1/|\mathcal{L}|$. Therefore, in order to achieve the desirable value $Q = 1 - \frac{k}{|\mathcal{L}|}$, we need to serve locally the requests for k files. That is, to find k subsets of SBSs where the file requested by these SBSs will be cached (so as to avoid MBS multicasts).

Notice that each cache can store up to one file. Hence, the caching decisions should be *disjoint* with respect to the SBSs. For example, in Fig. 3, SBS 1 cannot store both files 1 and 2, because $S_1 = 1$. This ensures that the subsets $\{1\}$ and $\{1, 2\}$ in the SPP problem will not be both selected. In other words, the value of the objective function in (11) can be less or equal to $1 - \frac{k}{|\mathcal{L}|}$, if there exist k subsets in $\mathcal{L}$ that are pairwise disjoint.

Conversely, if a Set Packing for some k exists, then for each subset $\mathcal{L}(i)$ that is picked in it, one can place the file i to the cache of each one of the SBSs $n \in \mathcal{L}(i)$ corresponding to this subset. At most one file is placed in each cache, since the selected subsets in the list are pairwise disjoint. The cost will be equal to $1 - \frac{k}{|\mathcal{L}|}$. ■

SPP is NP-Hard and moreover it is inapproximable within $O(\sqrt{|\mathcal{E}|})$ [23]. According to the reduction, we create a SBS for each one of the elements in $\mathcal{E}$, and hence it holds $|\mathcal{E}| = N$, which completes the proof of Theorem 1.

B. Algorithm with performance guarantees

In this subsection, we present a caching algorithm with performance guarantees. We first note that, based on Theorem 1, it is unlikely to find a tight approximate solution to the MACP problem. Hence, we follow an alternative approach by letting the solution to violate the cache capacity constraints in equation (12) by a bounded factor. Such a constraint violation turns out to greatly facilitate the solution of the problem. Following that, we present a provably near-optimal solution algorithm applying linear relaxation and randomized rounding techniques, variants of which have been also used for optimizing graph cuts [36].

Algorithm 1: Randomized rounding algorithm with parameter $\mu \in (0, \frac{1}{2})$

Let $(\boldsymbol{x}^\dagger, \boldsymbol{y}^\dagger)$ be the optimal solution to LR(MACP);
Choose $m \in [\frac{1}{2} - \mu, \frac{1}{2} + \mu]$ uniformly at random;
Let $\mathcal{A} = \{(r, i) : r \in \mathcal{R},\ i \in \mathcal{I},\ y^\dagger_{ri} \geq m\}$;
Let $\mathcal{B} = \{(r, i) : r \in \mathcal{R},\ i \in \mathcal{I},\ y^\dagger_{ri} < m\}$;
Set $y_{ri} = 1\ \forall (r, i) \in \mathcal{A}$, and $y_{ri} = 0\ \forall (r, i) \in \mathcal{B}$;
for $n \in \mathcal{N},\ i \in \mathcal{I}$ **do**
  if $\exists\ r :\ y_{ri} = 0$ *and* $n \in r$ **then**
    $x_{ni} \leftarrow 1$;
  else
    $x_{ni} \leftarrow 0$;
  end
end
Output $\boldsymbol{x}, \boldsymbol{y}$;

To start, we introduce the *linear relaxation* of the MACP problem, which we refer to as LR(MACP). This differs from MACP in that the variables in $\boldsymbol{x}$ and $\boldsymbol{y}$ can take any real value within $[0, 1]$, i.e., constraints (13) and (14) are replaced by $x_{ni} \in [0, 1],\ \forall n \in \mathcal{N}, i \in \mathcal{I}$ and $y_{ri} \in [0, 1],\ \forall r \in \mathcal{R}, i \in \mathcal{I}$. The objective function and the constraints of the LR(MACP) problem are linear with respect to the optimization variables. Hence, it can be solved using standard linear optimization techniques [37]. We need to emphasize at this point that the number of optimization variables in the LR(MACP) problem is non-polynomial to the number of SBSs N, since there is a variable for each subset $r \in \mathcal{R}$ (equation (9)). In practice though, the number of SBSs in a macro-cell is small (e.g., a few tens), and hence we can apply software toolboxes like CPLEX and Mosek [38] to efficiently solve LR(MACP).

Having found a fractional solution to the LR(MACP) problem, denoted with $(\boldsymbol{x}^\dagger, \boldsymbol{y}^\dagger)$, the proposed algorithm applies randomized rounding techniques to approximate the (integer) solution of the MACP problem. Specifically, given an input parameter value $\mu \in (0, \frac{1}{2})$, the algorithm decides uniformly at random a threshold value $m \in [\frac{1}{2} - \mu, \frac{1}{2} + \mu]$. Then, iteratively it rounds each y_{ri} variable to 1 if its (fractional) value exceeds m (subset $\mathcal{A}$); otherwise it takes the 0 value (subset $\mathcal{B}$). Finally, a variable x_{ni} will take the value 1, if there exists y_{ri} variable with $n \in r$ that was rounded to 0; otherwise it takes the 0 value. The procedure is summarized in Algorithm 1. Then, we prove the following theorem.

Theorem 2. *Given that $c_S = 0$, Algorithm 1 outputs a solution of energy cost at most $\frac{2}{1-2\mu}$ times the optimal. The expected amount of data placed in each cache is at most $\frac{1}{2\mu}$ times its capacity.*

Proof: Let V_{opt} and V_1 indicate the optimal solution value

for the MACP problem and the one achieved by Algorithm 1 respectively. Then, it holds that:

$$
\begin{aligned}
&V_{\text{opt}} \geq \\
&\geq \sum_{r\in\mathcal{R}}\sum_{i\in\mathcal{I}} q_{ri}\Big(y^{\dagger}_{ri}(c_B+c_{Wr})+(1-y^{\dagger}_{ri})\sum_{n\in\mathcal{N}} c_n\Big) \\
&\geq \sum_{(r,i)\in\mathcal{A}} q_{ri}y^{\dagger}_{ri}(c_B+c_{Wr})+\sum_{(r,i)\in\mathcal{B}} q_{ri}(1-y^{\dagger}_{ri})\sum_{n\in\mathcal{N}} c_n \\
&\geq \sum_{(r,i)\in\mathcal{A}} q_{ri}(\frac{1}{2}-\mu)(c_B+c_{Wr})+\sum_{(r,i)\in\mathcal{B}} q_{ri}(\frac{1}{2}-\mu)\sum_{n\in\mathcal{N}} c_n \\
&= (\frac{1}{2}-\mu)V_1, \qquad (16)
\end{aligned}
$$

where the first inequality is because the optimal solution of the linear relaxed problem provides a lower bound to the optimal solution value of the initial problem. The second inequality is because we kept in the summation only a subset of the terms and all the terms are positive, i.e., $q_{ri} \geq 0$, $y^{\dagger}_{ri} \geq 0$, $1-y^{\dagger}_{ri} \geq 0$, $c_B + c_{Wr} \geq 0$, $c_n \geq 0$. The third inequality is because: $y^{\dagger}_{ri} \geq m \geq \frac{1}{2}-\mu$, $\forall (r,i) \in \mathcal{A}$ and $y^{\dagger}_{ri} < m \leq \frac{1}{2}+\mu$, $\forall (r,i) \in \mathcal{B}$.

We also note that the m value is picked uniformly at random from an interval of size 2μ. According to step 7 of Algorithm 1, a file i will be placed at a SBS cache n ($x_{ni}=1$) only if there exists $r \in \mathcal{R}$ for which $n \in r$ and $y_{ri}=0$. Variable y_{ri} takes the zero value when m is larger than $y^{\dagger}_{ri}$, which happens with probability at most $\frac{1-y^{\dagger}_{ri}}{2\mu}$. Hence, the probability that x_{ni} takes the value 1 is at most:

$$
\frac{1-\min\limits_{r\in\mathcal{R}:n\in r} y^{\dagger}_{ri}}{2\mu} \overset{(9)}{\leq} \frac{x^{\dagger}_{ni}}{2\mu} \qquad (17)
$$

Summing over all the files yields that the expected amount of data placed in a SBS cache $n \in \mathcal{N}$ is at most:

$$
\sum_{i\in\mathcal{I}} \big(\frac{x^{\dagger}_{ni}}{2\mu}\big) \overset{(12)}{\leq} \frac{1}{2\mu}\cdot S_n \qquad (18)
$$

■

For example, picking the value $\mu = \frac{1}{6}$ will result a solution of cost that is at most three times larger than the optimal violating cache capacities by a factor less than three. Picking a lower value μ yields a tighter performance guarantee, but increases the factor within which the cache capacities are violated. Hence, the parameter value μ can be used to control the trade off between performance and robustness of the solution, where different operators may decide different μ values based on their priorities.

Constructing a feasible solution. We note that, as the cache capacities of the SBSs may be violated by a factor of $\frac{1}{2\mu}$ when applying Algorithm 1, the operator may not be able to store and deliver through the SBSs all the files required to ensure the performance guarantee of our algorithm ($\frac{2}{1-2\mu}$). In this case, an option for the operator is to expand the cache capacities by a factor of $\frac{1}{2\mu}$. Nevertheless, the operator is often unwilling (or, incapable) to perform additional investments. Hence, it is needed to convert the solution of Algorithm 1 into a feasible solution, i.e., a solution that satisfies equation (12).

Algorithm 2: Heuristic algorithm

```
x ← [0, ..., 0] ;
I_n ← 0, ∀n ∈ N;
D ← N × I ;
for t = 1, 2, ..., Σ_{n∈N}(S_n) do
    (n*, i*) ← argmin_{(n,i)∈D} f(x, n, i);
    x_{n*i*} = 1;
    D ← D \ (n*, i*);
    I_{n*} ← I_{n*} + 1;
    if I_{n*} = S_{n*} then
        for i ∈ I such that (n*, i) ∈ D do
            D ← D \ (n*, i)
        end
    end
end
Set y using equation (19);
Output x, y;
```

To obtain such a solution, we first note that for a given caching policy $\boldsymbol{x}$, we can compute the multicast policy $\boldsymbol{y}$ as follows:

$$
y_{ri} = \max\Big\{\max_{n\in r\setminus n_0}\{1-x_{ni}\}, \mathbf{1}_{\{n_0\in r\}}\Big\}. \qquad (19)
$$

Here, the external max term is equal to 1 if at least one of the two internal terms is equal to 1, i.e., if a request for file i is generated in area $n_0 \in r$ or a request for file i is generated in an area $n \in r$ and SBS n has not stored this file (cf. inequalities (8),(9)). Keeping that in mind, we can write the energy cost as a function of the caching policy $\boldsymbol{x}$ only. Then, starting with the $\boldsymbol{x}$ solution outputted by Algorithm 1, we iteratively perform the removal from a file to a SBS cache that yields the minimum energy cost increment. At each iteration, we ensure that the SBSs with remaining amount of cached data, that is lower or equal to their capacities, are excluded from content removal. The procedure ends when there is not any available SBS to remove content.

Please notice that, the above conversion may deteriorate the quality of the solution of Algorithm 1. Unfortunately, we cannot derive a tight theoretical performance bound for the obtained solution due to hardness of the MACP problem (as we described in Theorem 1). However, as we show with an extensive numerical study in the next section, the obtained solution operates very close to the optimal one in realistic settings.

C. Heuristic algorithm

Finally, we present an alternate algorithm firstly proposed in our preliminary work in [1]. In contrast to the previous algorithm, this algorithm finds a solution to the MACP problem in a greedy manner, rather than using a systematic optimization procedure. The proposed iterative algorithm starts with all the caches being empty. At each iteration, it greedily places the file to a cache that improves the objective function the most, terminating if all the caches become full. The procedure is summarized in Algorithm 2.

Specifically, I_n is the number of files already stored at the cache of SBS n at every iteration of the algorithm, and $(\times)$ denotes the cartesian product of two sets. The set $\mathcal{D}$ includes all the pairs (n, i) for which the placement of file i at the cache of SBS n has not been performed yet, and the cache of n has not been filled up yet. Let $f(\boldsymbol{x}, n, i)$ be the energy cost for the caching policy $\boldsymbol{x}$, where we additionally set $x_{ni} = 1$. Recall that, for a given caching policy the multicast policy $\boldsymbol{y}$ can be found using equation (19). This way, $f(.)$ is expressed as a function of $\boldsymbol{x}$ only. At every iteration, Algorithm 2 picks the pair $(n^*, i^*) \in \mathcal{D}$ with the lowest cost value $f(\boldsymbol{x}, n^*, i^*)$ provided that this is lower than in the previous iteration. This corresponds to the placement of the file i^* at the cache of the SBS n^*. If the cache of SBS n^* becomes full, Algorithm 2 excludes all the pairs $(n^*, i)\ \forall i$ from $\mathcal{D}$. This way, no more files will be stored at cache n^*.

Algorithm 2 terminates in $\sum_{n=1}^{N}(S_n)$ iterations. At each iteration it evaluates $f(.)$ after each one of $N \cdot I$ candidate file placements. Despite the lack of any theoretical performance guarantees, Algorithm 2 performs markedly better than existing caching schemes, as we show numerically in the next section. Moreover, Algorithm 2 can be extended to handle scenarios where *multiple* MBSs *share* a backhaul link and may coordinate their downloads over it to avoid unnecessary data retransmissions. Consider for example two MBSs that receive requests for file i from areas $r_1 \in \mathcal{R}_1$ and $r_2 \in \mathcal{R}_2$ respectively. File i can be multicasted to MBSs via the backhaul link when at least one MBS requests it, i.e., when $y_{r_1 i} = 1$ or $y_{r_2 i} = 1$. We denote with $z_{r_1 \cup r_2} \in \{0, 1\}$ the above event. Then, the energy cost for delivering file i is:

$$\widehat{J}_i(\boldsymbol{y}, \boldsymbol{z}) = \tag{20}$$
$$= \sum_{r_1 \in \mathcal{R}_1, r_2 \in \mathcal{R}_2} q_{r_1 \cup r_2 i} \cdot \Big(z_{r_1 \cup r_2 i} \cdot c_B + y_{r_1 i} \cdot c_{W r_1} +$$
$$+ (1 - y_{r_1 i}) \cdot \sum_{n \in r_1} c_n + y_{r_2 i} \cdot c_{W r_2} + (1 - y_{r_2 i}) \cdot \sum_{n \in r_2} c_n \Big),$$

where it is needed that $z_{r_1 \cup r_2 i} \geq y_{r_1 i}$ and $z_{r_1 \cup r_2 i} \geq y_{r_2 i}\ \forall r_1 \in \mathcal{R}_1, r_2 \in \mathcal{R}_2$. Algorithm 2 can be directly extended by considering the above function in place of $J_i(\boldsymbol{y})$.

IV. Performance Evaluation

In this section, we numerically evaluate the energy savings achieved by the proposed multicast-aware caching algorithms over existing caching strategies. The main part of the evaluation is carried out for a sporting event with thousand attendees [15] covered by a MBS and several SBSs. Additional scenarios differing in the population density, number of SBSs and energy costs are evaluated, which lead to an understanding of how the savings vary in different regions and markets. Overall, we find that moving from a conventional caching scheme to one enhanced with multicast-awareness can indeed reduce energy costs, and the benefits are higher when the demand is massive and the user requests for content are delay-tolerant. These benefits are up to 19% when the multicast streams are delivered every 3 minutes, increasing further with the steepness of content access pattern. In the rest of this section, we discuss these results in detail; we begin by describing the algorithms and the simulation setup used in the later evaluations.

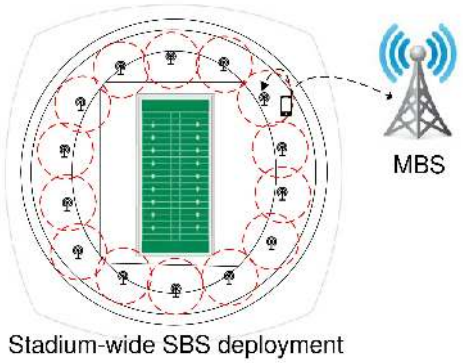


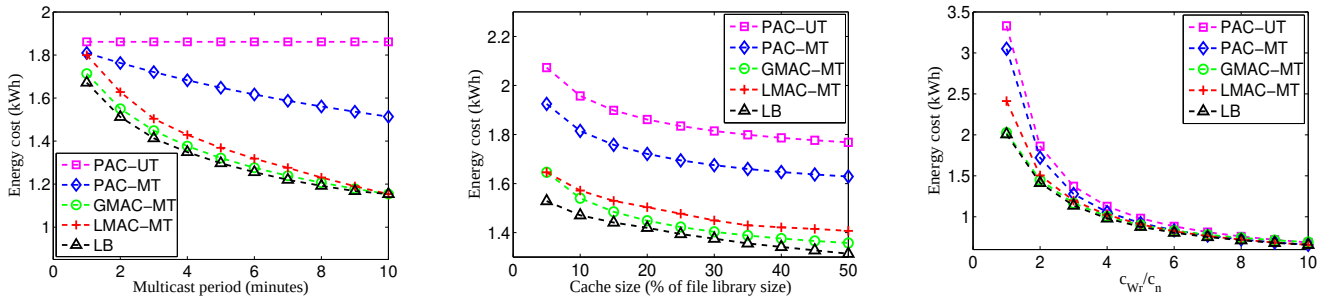
Fig. 4. A stadium-wide deployment of SBSs. The dashed circles represent the coverage areas of the SBSs. A user can be served either by the neighbor SBS or by the collocated MBS.

A. Algorithms and evaluation setup

Throughout, we compare the performance of five schemes:

1) *Popularity-Aware Caching & Unicast Transmissions (PAC-UT)*: The standard mode of operation currently in use in many caching systems. Each SBS stores in its cache the locally most popular files independently from the others. Each user request is served by a separate unicast transmission.
2) *Popularity-Aware Caching & Multicast Transmissions (PAC-MT)*: Similar to PAC-UT, differing in that all the requests for the same file within the same multicast period are served by a single multicast transmission (cf. equation (19)).
3) *Linear-Relaxed Multicast-Aware Caching & Multicast Transmissions (LMAC-MT)*: We apply Algorithm 1 with $\mu = 1/6$ to decide the cache placement. The placement is further processed to yield a feasible solution as described in the end of Subsection III-B. All the user requests for the same file within the same multicast period are served by a single multicast transmission (cf. equation (19)).
4) *Greedy Multicast-Aware Caching & Multicast Transmissions (GMAC-MT)*: Similar to LMAC-MT, differing in that we apply Algorithm 2 to decide the cache placement.
5) *Lower-bound (LB)*: The lower bound to the optimal solution of the MACP problem found by solving the linear relaxed problem (LR(MACP)). Since, this solution is not feasible, it is only used as a benchmark for measuring the efficacy of the proposed algorithms.

We need to emphasize that, in order to solve the linear problem in LMAC-MT and LB schemes, we executed code from the Visual Studio environment using the Mosek Optimization Toolbox [38]. The main part of the code we wrote is publicly available online in [39]. Hence, the presented results can be easily verified for correctness, while we believe this



(a) Impact of multicast period (b) Impact of cache sizes (c) Impact of transmission costs

Fig. 5. Energy cost achieved by PAC-UT, PAC-MT, LMAC-MT, GMAC-MT and LB schemes for various values of: (a) the multicast time period, (b) the cache size of each SBS and (c) the base station transmission costs.

will encourage future experimentation with wireless caching algorithms for the benefit of the research community.

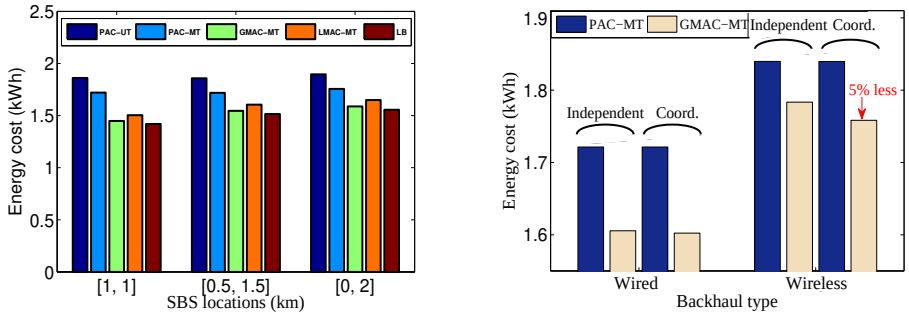
The main part of the evaluation is carried out for a sporting event with macrocell coverage and stadium-wide deployment of $N = 14$ SBSs as in Fig. 4. The system parameters are set using the measured trace of content requests collected during the 2013 Superbowl in February at the New Orleans Superdome [15]. During this event, over fifty thousand users generated around three thousand requests for a set of $I = 1,000$ popular files. Considering that all requests appear during the four-hour period of the game, this results to an average rate of ≈ 12.5 requests per minute. To model the user demand in our evaluation, we uniformly spread the requests in the trace across the coverage regions of the 14 SBSs. We further spread these requests across files using a Zipf popularity distribution with shape parameter z [40]. This results the demand values λ_{ni} for each SBS n and file i. We also set $\lambda_{0i} = 0$, $\forall i \in \mathcal{I}$. For the computation of p_{ni} and q_{ri}, we assume that request generation follows an independent Poisson distribution (equations (1), (3)). Unless otherwise specified, all files are of size 30MB and each SBS is equipped with a cache that can store up to 20% of the entire file library size. Finally, we set $z = 1.2$ (as in [40]) and $d = 3$ minutes, while our evaluation also covers a wide range of z and d values.

Following recent measurement traces in 3G networks, we approximate the power required by MBS for transmitting a file to a user in an area n by $P_n = 825/G_{MBS}$ Watts (cf. Fig. 3 in [22]). Here, G_{MBS} denotes the bandwidth capacity of the MBS. Since, the MBS capacities are typically dimensioned based on the anticipated demand, we set G_{MBS} to be capable of handling all the user requests in our simulation, i.e., $G_{MBS} = 12.5 \cdot d$ (requests per multicast period); therefore it is $P_n = 825/(12.5 \cdot d)$ $\forall n$. Then, using equation (5) we set: $c_{Wr} = \max_{n \in r} P_n$ $\forall r$. We later study the impact of heterogeneous c_{Wr} values, with power consumption increasing with the distance between MBS and user. SBS energy consumption is typically lower than the one for the MBS, due to the closer proximity to the users, with the actual value depending on the type of the SBS and its coverage. As a canonical scenario we set $c_n = c_{Wr}/2$, while our evaluation also covers the cases where: $\frac{c_{Wr}}{c_n} \in \{1, 2, ..., 10\}$ [34]. The power consumption of a wired backhaul link includes the power consumed at the aggregation switches $(1 - \alpha)\frac{Ag_{switch}}{Ag_{max}} P_{max}$ [34]. Here, P_{max} represents the maximum power consumption of the switch, Ag_{switch} is the amount of carried traffic, Ag_{max} is the maximum amount of traffic a switch can handle and $\alpha \in (0, 1)$. We set $P_{max} = 300$ (Watts), $Ag_{max} = G_{MBS}$ and $\alpha = 0.1$ (as in Table II in [34]); therefore it is $c_B = 30/(12.5 \cdot d)$. Finally, we consider a caching cost of $6.25 \cdot 10^{-12}$ Watts per bit (suitable for SSD hardware technology [35]) and set c_S accordingly.

B. Evaluation results

We compare the energy cost achieved by the above schemes as a function of the duration of multicast period, the cache sizes and the base station transmission costs. Following that, we repeat the experiments for two macro-cells sharing a backhaul link. Finally, we investigate how the population density, the steepness of demand and the number of SBSs impact the results.

Impact of the duration of the multicast period: Intuitively, multicast will be effective when there is significant concurrency in accessing content across users, i.e., many requests for the same file frequently appear within a multicast period. Although, this may occasionally be the case for typical urban macrocells with a few hundred or thousand users, our analysis reveals that it may be particularly relevant during crowded events with tens of thousand people collocated in the same area. For the specific sporting event that we consider in the evaluation, Fig. 5(a) shows the energy cost achieved by the discussed schemes when the duration of the multicast period d is varied within 1 to 10 minutes. We observe that the performance gap between each one of the schemes that enable mulitcast (PAC-MT, LMAC-MT, GMAC-MT and LB) and the PAC-UT increases with d. This was expected, since increasing d increases the probability of receiving multiple requests for a file within a period. Importantly, the proposed multicast-aware caching schemes (LMAC-MT, GMAC-MT) consistently outperform PAC-MT, with the gains increasing with d (up to 31%). Even for a relatively small value of d, the proposed schemes achieve significant gains over PAC-MT. For example, the gains are 19% for $d = 3$. This is important since users are unlikely to tolerate large delays in receiving content. Interestingly, the proposed schemes operate very close to LB and hence the optimal solution (less than 7% gap).



(a) Impact of cost heterogeneity

(b) Impact of coordination between MBSs

Fig. 6. The impact of (a) cost heterogeneity and (b) coordination between MBSs on algorithms' performance.

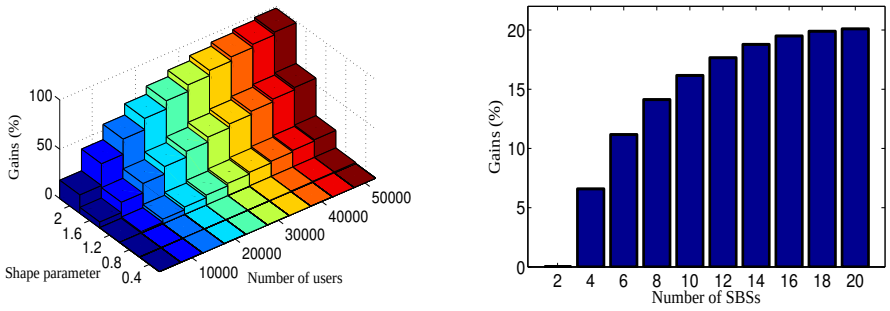
Impact of cache sizes: We analyze the impact of the cache sizes on the algorithms' performance in Fig. 5(b). Here, the cache size of each SBS is varied from 5% to 50% of the entire file library size. As expected, increasing cache sizes reduces energy costs for all schemes as more requests are satisfied locally (without the participation of the MBS). PAC-UT results in the largest energy cost compared to the rest schemes (up to 35% difference), since the latter schemes serve many aggregated requests via a single multicast instead of many unicast transmissions. The proposed multicat-aware caching schemes (LMAC-MT and GMAC-MT) consistently outperform the popularity-aware caching scheme PAC-MT, with the gains increasing with cache sizes (up to 20%). More importantly, LMAC-MT and GMAC-MT operate very close to LB -and hence the optimal solution- for all the cache sizes (less than 7% worse).

Impact of base station transmission costs: We explore the impact of the base station transmission cost parameters on the algorithms' performance in Fig. 5(c). Particularly, we keep c_{Wr} constant and alter the c_n values within $\{c_{Wr}/1, c_{Wr}/2, ..., c_{Wr}/10\}$. We observe that as the ratio c_{Wr}/c_n increases, the energy cost achieved by all the schemes decreases since the cost incurred by the service from the SBSs becomes lower. The popularity-aware caching schemes (PAC-UT and PAC-MT) are the most sensitive to this alteration. Again, LMAC-MT and GMAC-MT outperform the popularity-aware schemes, especially for low values of c_{Wr}/c_n. For $c_{Wr} = c_n$, the gains are 51% and 27% when compared to the PAC-UT and PAC-MT scheme respectively. However, when c_n values become relatively low compared to c_{Wr}, the performance of the PAC-MT scheme comes very close to the multicast-aware schemes. This is because, the file popularity distribution is the same across all the SBSs (homogeneous demand) in our experiment, and hence simply replicating the (same) most popular files at all the caches drastically reduces the number of multicast-transmissions employed by the MBS. We explored the impact of heterogeneous demand across the SBSs in our prior work [1] using synthetic data, where we showed that GMAC-MT exhibits substantial gains over PAC-MT and PAC-UT for arbitrarily low c_n values.

The numerical results presented so far assume homogeneous power consumption of the MBS across SBS areas, i.e., the P_n values are the same. Nevertheless, the power consumption typically varies depending on the distance between MBS and receiver and the conditions of the channel. To capture the above, we consider P_n to increase proportionally to the square of the distance between MBS and SBS n. Then, we randomly deploy the SBSs such that their distances from MBS range within $[1,\ 1]$, $[0.5,\ 1.5]$ and $[0,\ 2]$ km (Fig. 6(a)). In the first case, all the SBSs are deployed on a perfect circle of radius 1km around MBS and the power consumption of the MBS is homogeneous as before ($P_n = 825/(12.5 \cdot d)\ \forall n$). In the rest scenarios P_n is heterogeneous. We observe that the energy cost slightly increases for all the schemes as P_n becomes more heterogeneous. Interestingly, the proposed schemes consistently outperform the rest.

Extension to multiple MBSs: We now evaluate the scenario of two MBSs sharing a backhaul link towards the core network. The MBSs may operate independently one another or coordinate their data downloading through the backhaul link. Therefore, a natural question that arises is what benefits such coordination may yield. Fig. 6(b) aims to shed light on this question by considering the cases that the backhaul link is wired or wireless. For the latter, we set the c_B cost to be ten times higher than the wired case. We find that coordination can indeed reduce energy cost, but the gains are low ($\leq 1\%$ and $\leq 5\%$ for the wired and wireless case respectively). This can be explained noting that most of the energy is consumed at the links between MBS and users rather than the backhaul.

Impact of demand patterns and number of SBSs: The demand patterns used in Fig. 5 and 6 may seem contrived, but in fact, they are very much in line with recent traffic measurements reported during crowded events [15]. To obtain a holistic view of the benefits of enhancing the caching scheme with multicast-awareness, we repeat the experiments for different values of population density and steepness of demand. Specifically, we consider ten scenarios with five to fifty thousand users generating requests for files. The intensity of demand for the case of fifty thousand people matches the one used for the sporting game in Fig. 5 and 6. For the rest choices, the demand intensity is scaled down proportionally to the number of users. For each scenario, five different choices of the Zipf shape parameter z are evaluated. Here, $z = 0.4$ indicates an almost uniform content popularity



(a) Impact of user demand

(b) Impact of number of SBSs

Fig. 7. Gains of multicast-aware caching (GMAC-MT) over conventional caching (PAC-MT) as a function of (a) the intensity and steepness of demand for content and (b) the number of SBSs.

distribution, whereas $z = 2$ stands for a high-steep distribution. The 3-D barplot in Fig. 7(a) shows that the energy gains of a multicast-aware caching scheme (GMAC-MT) over a conventional caching scheme (PAC-MT) increase as either the intensity or the steepness of demand increases. In the best scenario, with fifty thousand users and $z = 2$, the gains are more than 90%.

Finally, we explore how the number of SBSs N impacts the results. Fig. 7(b) shows that the gains of GMAC-MT over PAC-MT increase as N increases. For example, the gains grow from 6.6% when $N = 4$ to 17.7% when $N = 12$, and further increase to 20.1% for $N = 20$. This is because, increasing N makes it more likely that concurrent requests for the same file occur at different SBSs, which implies a higher number of MBS multicast transmissions. This in turn calls for a careful cache-design that intelligently balances the number of requests served via MBS and SBS multicast.

V. Related work

The idea of leveraging storage for improving network performance is gaining increasing interest with applications in content distribution [25], [26], IPTV [27], social [28] and heterogeneous cellular networks [17]-[20], [41], [42]. Caching popular files at the SBSs has been studied from an optimization [17]-[20] and a game theoretic point of view [41], [42] with the results spanning a wide range of techniques, such as discrete/convex optimization, content-centric networking algorithms, facility location algorithms, coalition formation and matching games. The SBS caching problem was reconsidered in [43]-[44] for the special case that mobile users request videos at different qualities. Here, each video is encoded into multiple segments (called versions and layers), and caching decisions are taken per segment, rather than per video. The impact of caching on the energy consumption and backhaul usage for renewable energy powered small cell networks with limited battery capacity and backhaul bandwidth was investigated in [45]. Additional SBS caching schemes targeting to the minimization of user equipment energy consumption have been derived in [46],[47]. A mixed-timescale optimization of MIMO precoding and cache control was proposed in [48] for the case that SBSs cooperate when transmitting data to users.

All the above works assume that the users' demand profiles are perfectly known and optimize caching decisions based on content demand solely, an assumption that was firstly relaxed in [29], [30]. In our recent work in [49], we proposed the caching policy design with concerns on both the user mobility statistics and the content demand. More recently, Yue et al. [50] considered the case that the SBSs are privately owned and proposed an auction-based caching mechanism. *However, this is the first work, building on our initial study [1], that performs SBS caching with concerns on multicast-transmissions.*

Despite the plethora of work related to multicast, previous efforts have mainly focused on homogeneous networks [51]. Among the few works for multicast in heterogeneous cellular networks, protocols that enable cooperation between the macro-cell and femto-cell base stations to support multicast services were presented in [52], [53]. A mechanism to provide seamless handover between different networks and ubiquitous support for multicast/broadcast service was proposed in [54]. Another multicast mechanism that adaptively selects the cell and the wireless technology for each mobile host to join the multicast group was presented in [55]. However, none of the above multicast mechanisms considers caching at the SBSs.

The optimal multicast scheduling policy for a *given* cache placement at a base station has been explored in [33]. Joint caching and broadcast scheduling policies for information delivery in conventional cellular networks (i.e., without SBSs) were presented in [56], [57]. In these systems, users are equipped with caches in order to store in advance broadcasted content and retrieve later when they need it. More recently, Maddah-Ali et al. [58] developed a joint caching and multicast scheduling scheme aimed at reducing the *peak traffic rate* for serving a set of users, each one requesting a single file. In their subsequent work [59], the authors extended the scheme to minimize the *average traffic rate*, assuming that the file popularity distribution is uniform across all users. In contrast to these works, we consider cache-capable SBSs and design multicast-aware caching policies that minimize the *average cost* incurred for serving users with heterogeneous requests. Finally, we emphasize that, compared to our initial study [1] that focused on the benefits of a heuristic multicast-aware caching algorithm over traditional schemes using synthetic

data, in this paper we additionally derive an algorithm with theoretical performance guarantees and provide a careful trace-driven numerical analysis.

VI. Conclusions

In this paper, we proposed a caching paradigm able to reduce the energy costs for serving the massive mobile data demand in 5G wireless networks. In contrast to the traditional caching schemes that simply bring popular content close to users, our caching strategy is carefully designed so as to additionally exploit multicast. This is of high importance nowadays, since multicast attracts attention as a technique for efficient content delivery in the evolving cellular networks. To overcome the NP-Hardness nature of the revisited caching problem, we introduced an algorithm with performance guarantees and also a simple heuristic algorithm, and evaluated their efficacy through a careful trace-driven numerical analysis. The results demonstrated that combining caching and multicast can indeed reduce energy costs when the demand for delay-tolerant content is massive. The gains over conventional caching schemes are 19% when users tolerate delay of three minutes, increasing further with the steepness of content access pattern. Overall, our work can be seen as an attempt to combine caching and multicast in a systematic way as a means of improving energy efficiency in 5G wireless networks.

References

[1] K. Poularakis, G. Iosifidis, V. Sourlas, L. Tassiulas, "Multicast-aware Caching for Small-Cell Networks", IEEE Wireless Communications and Networking Conference (WCNC), pp. 2300-2305, April 2014.

[2] Ericsson, "Mobility Report: On the Pulse of Networked Society", June 2015, http://www.ericsson.com/mobility-report.

[3] J. G. Andrews, "Seven ways that hetnets are a cellular paradigm shift", IEEE Communications Magazine, vol. 51, no. 3, pp. 136-144, March 2013.

[4] Y. Xu, J. Wang, Q. Wu, Z. Du, L. Shen, A. Anpalagan, "A game theoretic perspective on self-organizing optimization for cognitive small cells",IEEE Communications Magazine, vol. 53, no. 7, pp. 100-108, 2015.

[5] J. Erman, A. Gerber, M.T. Hajiaghayi, "To Cache or Not to Cache-The 3G Case", IEEE Internet Computing, vol. 15, no. 2, pp. 27-34, March 2011.

[6] B.A. Ramanan, L.M. Drabeck, M. Haner, N. Nithi, T.E. Klein, C. Sawkar, "Cacheability Analysis of HTTP traffic in an Operational LTE Network", Wireless Telecommunications Symposium, pp. 1-8, April 2013.

[7] Mobile Europe, "Altobridge debuts intel-based network edge small cells caching solution", June 2013.

[8] Light Reading, "NSN Adds ChinaCache Smarts to Liquid Applications", March 2014.

[9] Saguna, "Saguna Open-RAN", 2015, http://www.saguna.net/products/saguna-cods-open-ran.

[10] OFweek, 'China Telecom successfully deployed LTE eMBMS", June 2014.

[11] Alcatel-Lucent, "eMBMS for More Efficient Use of Spectrum", November 2011.

[12] 3GPP releases, http://www.3gpp.org/specifications/releases/71-release-9.

[13] Ericsson, LTE Broadcast, February 2013, http://www.ericsson.com/res/thecompany/docs/press/backgrounders/lte-broadcast-press-backgrounder.pdf

[14] Qualcomm, LTE Broadcast, https://www.qualcomm.com/invention/technologies/lte/broadcast

[15] J. Erman, K.K. Ramakrishnan, "Understanding the super-sized traffic of the super bowl", ACM IMC, pp. 353-360, November 2013.

[16] M.Z. Shafiq, L. Ji, A.X. Liu, J. Pang, S. Venkataraman, J. Wang, "A First Look at Cellular Network Performance during Crowded Events", ACM SIGMETRICS, pp. 17-28, June 2013.

[17] N. Golrezaei, K. Shanmugam, A. Dimakis, A. Molisch and G. Caire, "FemtoCaching: Wireless Video Content Delivery through Distributed Caching Helpers", IEEE Conference on Computer Communications (Infocom), pp. 1107-1115, March 2012.

[18] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. 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.

[19] K. Poularakis, G. Iosifidis, L. Tassiulas, "Approximation Algorithms for Mobile Data Caching in Small Cell Networks", IEEE Transactions on Communications, vol 62, no. 10, pp. 3665-3677, October 2014.

[20] M. Dehghan, A. Seetharam, B. Jiang, T. He, T. Salonidis, J. Kurose, D. Towsley and R. Sitaraman, "On the Complexity of Optimal Routing and Content Caching in Heterogeneous Networks", IEEE Conference on Computer Communications (Infocom), April 2015.

[21] V. Tokekar, A. K. Ramani, and S. Tokekar, "Analysis of Batching Policy in View of User Reneging in VoD System", IEEE Indicon, pp. 399-403, December 2005.

[22] C. Peng, S. Lee, S. Lu, H. Luo, H. Li, "Traffic-Driven Power Saving in Operational 3G Cellular Networks", ACM International Conference on Mobile Computing and Networking (Mobicom), pp. 121-132, September 2011.

[23] M. Garey, D. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness", W. Freeman & Comp., San Francisco, 1979.

[24] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi, "A survey on 3GPP heterogeneous networks", IEEE Transactions on Wireless Communications, vol. 18, no. 3, pp. 10-21, June 2011.

[25] S. Borst, V. Gupta, and A. Walid, "Distributed Caching Algorithms for Content Distribution Network", IEEE Conference on Computer Communications (Infocom), pp. 1-9, March 2010.

[26] K. Poularakis and L. Tassiulas, "Optimal Cooperative Content Placement Algorithms in Hierarchical Cache Topologies", Conference on Information Sciences and Systems (CISS), pp. 1-6, March 2012.

[27] J. Dai, Z. Hu, B. Li, J. Liu, and B. Li, "Collaborative Hierarchical Caching with Dynamic Request Routing for Massive Content Distribution", IEEE Conference on Computer Communications (Infocom), pp. 2444-2452, March 2012.

[28] M. Taghizadeh, K. Micinski, C. Ofria, E. Torng, S. Biswas, "Distributed Cooperative Caching in Social Wireless Networks", IEEE Transactions on Mobile Computing, vol. 12, no. 6, pp. 1037-1053, June 2013.

[29] E. Bastug, J. L. Guenego, and M. Debbah, "Proactive Small Cell Networks", International Conference on Telecommunications (ICT), pp. 1-5, May 2013.

[30] P. Blasco and D. Gunduz, "Learning-Based Optimization of Cache Content in a Small Cell Base Station", IEEE International Conference on Communications, pp. 1897-1903, June 2014.

[31] K. Dufkova, M. Popovic, and R. Khalili, J. Boudec, M. Bjelica, and L. Kencl, "Energy Consumption Comparison Between Macro-Micro and Public Femto Deployment in a Plausible LTE Network", International Conference on Energy-Efficient Computing and Networking (e-Energy '11), pp. 67-76, May 2011.

[32] G. Koutitas, G. Iosifidis, B. Lannoo, M. Tahon, S. Verbrugge, P. Ziridis, L. Budzisz, M. Meo, M.A. Marsan, L. Tassiulas,, "Greening the Airwaves with Collaborating Mobile Network Operators", IEEE Transactions on Wireless Communications, September 2015.

[33] B. Zhou, Y. Cui and M. Tao, "Optimal Dynamic Multicast Scheduling for Cache-Enabled Content-Centric Wireless Networks", IEEE International Symposium on Information Theory (ISIT), pp. 1412-1416, June 2015.

[34] S.Tombaz, P. Monti, K.Wang, A. Vastberg, M. Forzati, J. Zander, "Impact of Backhauling Power Consumption on the Deployment of Heterogeneous Mobile Networks", IEEE Global Communications Conference (Globecom), pp. 1-5, December 2011.

[35] N. Choi, K. Guan, D. C. Kilper, and G. Atkinson, "In-network caching effect on optimal energy consumption in content-centric networking", IEEE International Conference on Communications, pp. 2889-2894, June 2012.

[36] A. Hayrapetyan, D. Kempe, M. Pál, Z. Svitkina, "Unbalanced graph cuts", European Symposium on Algorithms (ESA), pp. 191-202, October 2005.

[37] D. Bertsimas and J. N. Tsitsiklis, "Introduction to Linear Optimization", Belmont, MA: Athena Science, 1997.

[38] Mosek Optimization Software, [online] http://www.mosek.com

[39] K. Poularakis, G. Iosifidis, V. Sourlas and L. Tassiulas, Publicly available code, https://www.dropbox.com/s/6u3xmqmi5bmb96t/twccode.rar?dl=0

[40] Y. Sun, S. K. Fayaz, Y. Guo, V. Sekar, Y. Jin, M. A. Kaafar, and S. Uhlig, "Trace-driven analysis of icn caching algorithms on video-on-demand workloads", ACM CoNEXT, pp. 363-376, December 2014.
[41] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "In-Network Caching and Content Placement in Cooperative Small Cell Networks", International Conference on 5G for Ubiquitous Connectivity (5GU), pp. 128-133, November 2014.
[42] K. Hamidouche, W. Saad and M. Debbah, "Many-to-Many Matching Games for Proactive Social-Caching in Wireless Small Cell Networks", International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (Wiopt), pp. 569-574, May 2014.
[43] K. Poularakis, G. Iosifidis, A. Argyriou, L. Tassiulas, "Video Delivery over Heterogeneous Cellular Networks: Optimizing Cost and Performance", IEEE Conference on Computer Communications (Infocom), pp. 1078-1086, April 2014.
[44] P. Ostovari, A. Khreishah and J. Wu, "Cache Content Placement Using Triangular Network Coding", IEEE Wireless Communications and Networking Conference (WCNC), pp. 1375-1380, April 2013.
[45] A. Kumar and W. Saad, "On the Tradeoff between Energy Harvesting and Caching in Wireless Networks", IEEE International Conference on Communication Workshop (ICCW), pp. 1976-1981, June 2015.
[46] M. Erol-Kantarci, "Content Caching in Small Cells with Optimized Uplink and Caching Power", IEEE Wireless Communications and Networking Conference (WCNC), pp. 2173-2178, March 2015.
[47] M. Erol-Kantarci, "Uplink Power Optimized In-Network Content Caching for HetNets", International Conference on Computing, Networking and Communications (ICNC) - Workshop on Computing, Networking and Communications (CNC), pp. 270-274, February 2015.
[48] A. Liu and V. K. N. Lau, "Mixed-timescale precoding and cache control in cached MIMO interference network", IEEE Transactions on Signal Processing, vol. 61, no. 24, pp. 6320-6332, December 2013.
[49] K. Poularakis and L. Tassiulas, "Exploiting User Mobility for Wireless Content Delivery", IEEE International Symposium on Information Theory (ISIT), pp. 1017-1021, July 2013.
[50] J. Yue, B. Yang, C. Chen, X. Guan, W Zhang, "Femtocaching in video content delivery: Assignment of video clips to serve dynamic mobile users", Computer Communications, vol. 51, pp. 60-69, September 2014.
[51] D-E. Meddour, A. Abdallah, T. Ahmed, R. Boutab, "A cross layer architecture for multicast and unicast video transmission in mobile broadband networks", Journal of Network and Computer Applications, vol. 35, no. 5, pp. 1377-91, September 2012.
[52] M. Peng, Y. Liu, D. Wei, W. Wang, HH. Chen, "Hierarchical cooperative relay based heterogeneous networks", IEEE Transactions on Wireless Communications, vol. 18, no. 3, pp. 48-56, June 2011.
[53] X. Xie, B. Rong, T. Zhang, W. Lei, "Improving physical layer multicast by cooperative communications in heterogeneous networks", IEEE Transactions on Wireless Communications, vol. 18, no. 3, pp. 58-63, June 2011.
[54] K. Ying, H. Yu, X. Wang, H. Luo, "Multicast/broadcast service over heterogeneous networks", IEEE Global Communications Conference (Globecom), pp. 1-5, December 2011.
[55] DN. Yang, MS. Chen, "Efficient resource allocation for wireless multicast", IEEE Transactions on Mobile Computing, vol. 7, no. 4, pp. 387-400, April 2008.
[56] C. Su and L. Tassiulas, "Joint broadcast scheduling and user's cache management for efficient information delivery", Wireless Networks, vol. 6, no. 4, pp 279-288, July 2000.
[57] J. Tadrous, A. Eryilmaz, H. El Gamal, "Proactive Resource Allocation: Harnessing the Diversity and Multicast Gains", IEEE Transactions on Information Theory, vol.59, no.8, pp. 4833-4854, August 2013.
[58] MA. Maddah-Ali and U. Niesen, "Fundamental Limits of Caching", IEEE Transactions on Information Theory, vol. 60, no. 5, pp. 2856-2867, May 2014.
[59] MA. Maddah-Ali and U. Niesen, "Coded caching with nonuniform demands", IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 221-226, April 2014.

Konstantinos Poularakis obtained the Diploma, and the M.S. and Ph.D. degrees in Electrical Engineering from the University of Thessaly, Greece, in 2011, 2013 and 2015 respectively. Currently, he is a Post-doc researcher at the same university. He has been honored with several awards during his studies, from sources including the Greek State Scholarships foundation (IKY) and the Center for Research and Technology Hellas (CERTH). He also received a Ph.D. scholarship from the "Alexander S. Onassis Public Benefit Foundation". His research interests lie in the broad area of network optimization and network economics.

George Iosifidis obtained the Diploma in Electronics and Telecommunications Engineering from the Greek Air Force Academy, in 2000, and the M.S. and Ph.D. degrees in Electrical Engineering from the University of Thessaly, Greece, in 2007 and 2012, respectively. He is currently a Post-doc Associate at Yale University, Institute for Network Science, USA. His research interests lie in the broad area of network optimization and network economics.

Vasilis Sourlas received his Diploma degree from the Computer Engineering and Informatics Department, University of Patras, Greece, in 2004 and the M.Sc. degree in Computer Science and Engineering from the same department in 2006. In 2013 he received his PhD from the Department of Electrical and Computer Engineering, University of Thessaly (Volos), Greece. In Jan. 2015 he joined the Electronic and Electrical Engineering Department, UCL, London to pursue his two years Marie Curie IEF fellowship. His main interests are in the area of Information-Centric Networks and Future Internet.

Leandros Tassiulas (S89-M91-SM06-F07), the John C. Malone Professor of Electrical Engineering at Yale University, obtained the Diploma in Electrical Engineering from the Aristotelian University of Thessaloniki, Greece in 1987, and the M.S. and Ph.D. degrees in Electrical Engineering from the University of Maryland, College Park, in 1989 and 1991, respectively. He has held positions as Assistant Professor at Polytechnic University New York (19911995), Assistant and Associate Professor University of Maryland College Park (19952001), and Professor at University of Ioannina (19992001) and University of Thessaly (2002-2015), Greece. His research interests are in the field of computer and communication networks with emphasis on fundamental mathematical models, architectures and protocols of wireless systems, sensor networks, high-speed internet and satellite communications. Dr. Tassiulas is a Fellow of IEEE. He received a National Science Foundation (NSF) Research Initiation Award in 1992, an NSF CAREER Award in 1995 an Office of Naval Research, Young Investigator Award in 1997 and a Bodosaki Foundation award in 1999. He also received the INFOCOM 1994 best paper award, the INFOCOM 2007 achievement award, and the IEEE 2016 Koji Kobayashi Computers and Communication Award.