Optimizing MDS Coded Caching in Wireless Networks with Device-to-Device Communication

Citation for the original published paper (version of record):

Pedersen, J., Graell i Amat, A., Andriyanova, I. et al (2019)
Optimizing MDS Coded Caching in Wireless Networks with Device-to-Device Communication
IEEE Transactions on Wireless Communications, 18(1): 286-295
http://dx.doi.org/10.1109/TWC.2018.2879358

Optimizing MDS Coded Caching in Wireless Networks with Device-to-Device Communication

Jesper Pedersen, Alexandre Graell i Amat, *Senior Member, IEEE*,
Iryna Andriyanova, *Member, IEEE*, and Fredrik Brännström, *Member, IEEE*

***Abstract*—We consider the caching of content in the mobile devices in a dense wireless network using maximum distance separable (MDS) codes. We focus on an area, served by a base station (BS), where mobile devices move around according to a random mobility model. Users requesting a particular file download coded packets from caching devices within a communication range, using device-to-device communication. If additional packets are required to decode the file, these are downloaded from the BS. We analyze the device mobility and derive a good approximation of the distribution of caching devices within the communication range of mobile devices at any given time. We then optimize the MDS codes to minimize the network load under a cache size constraint and show that using optimized MDS codes results in significantly lower network load compared to when caching the most popular files. We further show numerically that caching coded packets of each file on all mobile devices, i.e., maximal spreading, is optimal.**

***Index Terms*—Caching, content delivery, device-to-device communication, device mobility, erasure correcting codes.**

I. Introduction

Mobile data traffic is predicted to increase significantly in the coming years [1], which imposes a severe strain on existing wireless networks. One of the most promising methods to offload traffic is storing content closer to the end users, a technique known as *caching* [2]–[6]. Content can be cached at small base stations (BSs) to reduce the burden on the backhaul links [3], [4]. Alternatively, content may be cached directly in the mobile devices, which helps in reducing both downlink traffic from BSs [4], [5] and the backhaul traffic.

A plethora of works on coded caching has appeared in recent years. In the literature, the concept of coded caching refers to both the caching of uncoded content to facilitate index-coded broadcasts [2], [7], and the use of erasure correcting codes to cache the content [6]–[15]. In both cases, the goal is to deliver content efficiently. In [2], index coding is shown to significantly reduce the amount of data that is required to transmit over a shared link. In [7], content is cached directly in the mobile devices. Asymptotic scaling laws of the amount of data necessary to transmit to satisfy worst case file demands using index-coded device-to-device (D2D) broadcasts for fixed network topologies is investigated for the case where the file size, the number of files in the library, and the number of users grows large. An additional layer of erasure correcting codes is suggested to facilitate a decentralized caching scheme.

Erasure correcting codes can significantly improve the performance in wireless networks when a user requesting content can access only a subset of the caches [6], [8]–[15]. In [6] and [8], files are cached in a number of small BSs from which mobile devices download content. It is shown that caching content using maximum distance separable (MDS) codes reduces the download delay and that the performance improves with an increase in the density of small BSs and a decrease of the density of devices in the network [6]. In [8], the use of MDS codes is shown to reduce the amount of data that is required to download from the macro BS. In [9]–[15] caching coded content directly in the mobile devices is considered and devices download requested files using D2D communication.

The caching of complete files in the mobile devices, where the devices move around according a simple random walk model, is considered in [9]. Files are cached randomly according to a Zipf distribution and the Zipf parameter is optimized to maximize the number of times a requested file can be found in the cache of a nearby device. Coded caching in the mobile devices considering device mobility has been studied in [10]–[15]. In [10], [11], [14], devices arrive to and depart from an area according to a Poisson process and coding is shown to reduce the amount of data required to download requested files. In [12], [13], the use of MDS codes is shown to minimize the download delay. All these previous works [9]–[12], [14] assume an area-centric model, where all devices within an area can communicate with each other, regardless of the distance between them. A more realistic model is a user-centric model where a device can communicate only with neighboring devices within a given communication range. To the best of our knowledge, [15] is the only paper that considers a user-centric mobility model and studies the effects of coded caching with device mobility. However, [15] assumes that the devices remain within the communication range for a deterministic time and that a file can be reconstructed from a random number of coded packets independent of the content allocation. Also, [15] considers only small networks in terms of number of users, corresponding to low device densities.

A. Contribution

In this paper, we study the effect of MDS-coded caching of content in the mobile devices to reduce the network load (from the BS and the mobile devices) in highly-dense wireless networks considering device mobility. As in [15], we consider

This work was funded by the Swedish Research Council under grant 2016-04253 and by the National Center for Scientific Research in France under grant CNRS-PICS-2016-DISCO.

J. Pedersen, A. Graell i Amat, and F. Brännström are with the Department of Electrical Engineering, Chalmers University of Technology, SE-41296 Gothenburg, Sweden (e-mail: {jesper.pedersen, alexandre.graell, fredrik.brannstrom}@chalmers.se).

I. Andriyanova is with the ETIS-UMR8051 group, ENSEA/University of Cergy-Pontoise/CNRS, 95015 Cergy, France (e-mail: iryna.andriyanova@ensea.fr).

a user-centric model, and assume that the devices move around an area according to the random waypoint model [16] and request files from a library at random times. Files are encoded using MDS codes of equal code length but potentially different code rate, and coded packets are cached in a number of mobile devices. When a device requests a file from the library, coded packets are downloaded from mobile devices within the communication range using D2D communication, and if additional packets are required to decode the requested file, these are retrieved from the BS. We analyze the mobility model and derive a good approximation of the distribution of the number of devices within range at the time of a request. We then formulate the minimization of the network load as a mixed integer linear program (MILP) that allows us to find the content allocation that minimizes the network load, i.e., minimizes the amount of data that is downloaded from the BS and mobile devices, assuming a global average cache size constraint (across all devices). We also suggest a greedy algorithm to enforce a strict cache size constraint per device. The problem formulation includes a weighting parameter to reflect the cost of utilizing the downlink and D2D communication. For a number of devices up to ~ 1000, we can solve the MILP using a branch-and-bound method that guarantees that the global minimum is attained. For a larger number of devices, i.e., higher device density scenarios, we propose a relaxation of the integer constraint of the MILP into a linear program (LP) which provides a lower bound on the network load. We also give a simple suboptimal algorithm to find an upper bound on the network load. We show numerically that caching packets of a given file on all mobile devices, i.e., maximal spreading [17], is optimal. We further show numerically for maximal spreading that the upper and lower bounds are approximately equal. Hence, the proposed lower and upper bounds provide a very good approximation to the optimal solution. Compared to [15], our formulation allows to analyze highly-dense networks.

II. System Model

We consider a cell with area A square meters served by a BS. We assume a higher level inter-cell interference coordination [18] such that we can consider inter-cell interference to be negligible, which enables us to analyze one cell in isolation. The area is projected onto a sphere of radius ρ meters to remove the area boundaries [19], where $A = 4\pi\rho^2$. M mobile devices are uniformly spread over the area. Users wish to download files from a library of N files that is always accessible to the BS[1]. We assume that all files have equal size F bits, which is without loss of generality since contents can always be divided into chunks with equal size [20]. Similar to most previous works on coded caching, see, e.g., [8], [21], we assume that the file popularity follows the Zipf distribution [22]. Hence, the popularity of file i is given by

$$p_i = \frac{1/i^\sigma}{\sum_{\ell=1}^{N} 1/\ell^\sigma}, \ i = 1, \ldots, N, \ 0 < \sigma \leq 1.5, \qquad (1)$$

where σ is the skewness parameter of the distribution. Note that, although all our results are obtained assuming a Zipf distribution of file popularities, the framework is general in the sense that other distributions, such as the Weibull and Gamma distributions that are suggested alternatives for YouTube videos in [23], can be used.

[1]This assumption is valid if we consider the BS to be connected to the core internet through a high capacity optical fiber backhaul link.

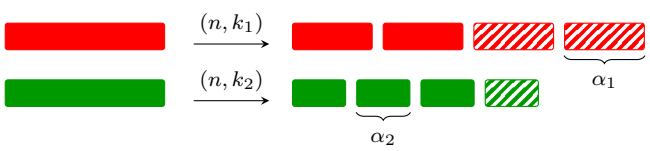

Figure 1. Encoding example for the caching of two files (red and green). In the example, $n = 4$, $k_1 = 2$, and $k_2 = 3$. The solid rectangles (to the right) represent the k_i packets which together with the dashed rectangles represent the n coded packets.

A. Content Allocation

Each file i that is to be cached is partitioned into k_i packets, each of size F/k_i bits, and encoded into n packets, also of size F/k_i bits, using an (n, k_i) MDS code of code length n, dimension $k_i = 1, \ldots, n$, and rate $R_i = k_i/n \leq 1$ [24]. Thus, different files are encoded by MDS codes of the same code length but potentially of different dimension, i.e., different rate. The n coded packets are cached in n mobile devices (possibly different for each file) in the area, chosen uniformly at random. Hence, for each file i, each of the n devices caching the file stores one coded packet of the file, i.e., a fraction $\alpha_i = 1/k_i$ of the file. Thus, as $k_i = 1, \ldots, n$,

$$\alpha_i \in \{0, 1/n, 1/(n-1), \ldots, 1\} \triangleq \mathcal{A}, \ i = 1, \ldots, N, \qquad (2)$$

where $\alpha_i = 0$ implies that file i is not cached. A small illustrative example where two files are to be cached using codes of parameters (n, k_1) and (n, k_2), with $n = 4$, $k_1 = 2$, and $k_2 = 3$, is shown in Fig. 1. We define the vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)$ and refer to it as the *content allocation*. Note that the content allocation is inversely proportional to the code rate as

$$R_i = \frac{1}{n\alpha_i}. \qquad (3)$$

In practice, a strict cache size constraint per device would be desirable. Unfortunately, this leads to a very complicated optimization problem. To simplify the problem and similar to [17], we enforce a global average cache size constraint, denoted by β, where

$$\sum_{i=1}^{N} \alpha_i \leq \beta. \qquad (4)$$

This implies an average cache size constraint per device

$$\beta_{\mathrm{d}} = \beta n/M. \qquad (5)$$

In Section IV, we suggest a suboptimal greedy algorithm that enforces a strict cache size constraint per device and show numerically in Section V that the incurred performance loss is negligible for a small cache size overhead. We remark that for $n = M$, i.e., maximal spreading, the average cache size constraint becomes a strict cache size constraint.

The commonly used *popular* content allocation, where each of the $\lfloor \beta \rfloor$ most popular files is cached in n (possibly different)

mobile devices (i.e., using an $(n, 1)$ repetition code), is given by

$$\boldsymbol{\alpha}_{\text{pop}} = \left(\mathbf{1}_{\lfloor\beta\rfloor}, \mathbf{0}_{N-\lfloor\beta\rfloor}\right), \tag{6}$$

where $\mathbf{1}_{\lfloor\beta\rfloor}$ is a vector with $\lfloor\beta\rfloor$ ones and $\mathbf{0}_{N-\lfloor\beta\rfloor}$ is a vector with $N - \lfloor\beta\rfloor$ zeros.

B. Data Download

Mobile devices request files at random times, with the time between requests exponentially independent, identically distributed (i.i.d.) with rate ω per second, i.e., the request process is a Poisson process [25]. Hence, the expected total request rate in the area is $M\omega$. A device requests file i with probability p_i given by (1). Due to the MDS property, k_i coded packets are sufficient to decode the file [24]. The user requesting content downloads as many coded packets as possible (up to k_i) from caching devices within a communication radius of r meters (measured over the curvature of the sphere), referred to as the communication range. If additional packets are required, these are retrieved from the BS. The equivalent number of files that are downloaded from the BS per second is referred to as the *downlink rate* and the equivalent number of files downloaded from caching devices (per second) is referred to as the *D2D communication rate*. We assume that D2D interference can be considered negligible, which can be achieved by considering the coordination of the radio resources using, e.g., a scheme similar to the one suggested in [26]. Similar assumptions are made in [14] and [15]. We furthermore assume that the communication is error free and incurs zero delay.

C. Device Mobility

The mobile devices move around the area according to the random waypoint model [16], which was compared with realistic data sets for a smaller number of mobile devices in [15]. According to this model, a mobile device pauses for a deterministic time and then picks a target uniformly in the area and a speed uniformly between a minimum and a maximum speed, denoted by $s_{\min}$ and $s_{\max}$, respectively. For simplicity, we assume that the pause time is zero but the analysis and results are easily generalized to account for a nonzero pause time using the results in [27]. The device traverses the great circle towards the target and, once the target has been reached, repeats the process. The targets and speeds are i.i.d. for all devices in the area. As two mobile devices move around the area, they are within the communication range for a random *contact time*, denoted by T_{c} and measured in seconds. The time between two contacts is referred to as the *intercontact time*, denoted by T_{i}, and the time between the beginning of two contacts is referred to as the *interarrival time*, denoted by T_{a}, where

$$T_{\text{a}} = T_{\text{c}} + T_{\text{i}}.$$

The device mobility model is illustrated in Fig. 2.

III. Network Statistics Analysis

In this section, we derive the probability to have a number of caching devices within the communication range at the time of a request, as well as the amount of data that is downloaded to serve user requests. We have the following theorem.

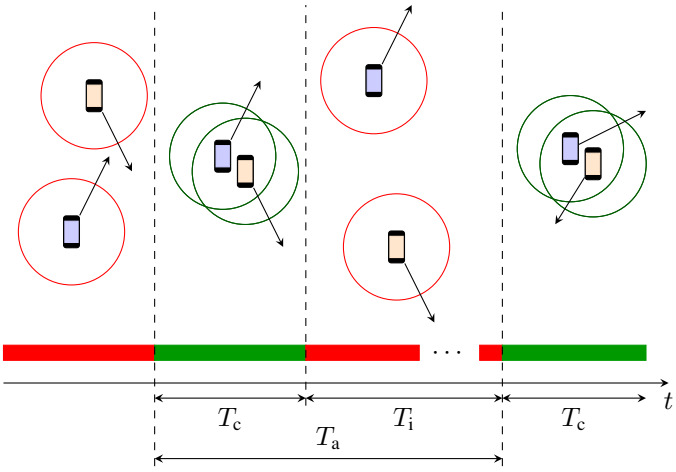

Figure 2. The device mobility model with contact, intercontact, and interarrival times illustrated for two devices. In frames 2 and 4, the devices are within each others communication range and in frames 1 and 3, the devices are not within range.

Theorem 1. *Consider the scenario in Section II with an area $4\pi\rho^2$, M mobile devices with communication range r, a minimum and maximum speed of devices $s_{\min}$ and $s_{\max}$, respectively, and a code length n. Under the assumption that $r \ll 2\rho$ and $s_{\min} \approx s_{\max}$, the probability that there are j caching devices within the communication range of any device is*

$$q_j = \frac{\left(\frac{\lambda}{\mu} \cdot \frac{n}{M}\right)^j}{j!} \cdot e^{-\frac{\lambda}{\mu}\cdot\frac{n}{M}}, \quad j \geq 0, \tag{7}$$

where the aggregate arrival rate of devices to within the communication range of any mobile device, denoted by λ, is

$$\lambda = (M-1)\frac{2rs}{4\pi\rho^2} \tag{8}$$

and the departure rate, denoted by μ, is

$$\mu = \frac{2s}{\pi r}. \tag{9}$$

Proof: For the random waypoint model, under the assumption that $s_{\min} \approx s_{\max}$, two devices move at an approximate average relative speed [27]

$$s = \frac{2(s_{\max} + s_{\min})}{\pi}$$

and the expected contact time is [27]

$$\mathbb{E}[T_{\text{c}}] = \frac{\pi r}{2s}, \tag{10}$$

which holds under the assumption that a device does not change direction during the contact time. This assumption is valid when the communication area πr^2 is much smaller than the area A, i.e., when $r \ll 2\rho$. The departure rate is given by

$$\mu = 1/\mathbb{E}[T_{\text{c}}], \tag{11}$$

which inserted in (10) gives (9).

The distribution of T_{i} can be closely approximated by

$$T_{\text{i}} \overset{\text{i.i.d.}}{\sim} \text{Exp}\left(\frac{2rs}{4\pi\rho^2}\right), \tag{12}$$

for $r \ll 2\rho$ [27]. The interarrival time is

$$T_{\mathrm{a}} = T_{\mathrm{c}} + T_{\mathrm{i}} \approx T_{\mathrm{i}}, \tag{13}$$

since $r \ll 2\rho$ implies $T_{\mathrm{c}} \ll T_{\mathrm{i}}$. Using (12) and (13), the aggregate arrival rate is the sum of the interarrival rates of $M-1$ devices in the area and we get (8) [28, Sec. 1.3.1.2].

To simplify the analysis, we assume that the arrival rate is independent of the number of devices within the communication range, which is a reasonable assumption for $r \ll 2\rho$. Under this assumption, the stochastic process describing the number of mobile devices within the communication range of any reference device can be characterized by an $\mathsf{M}/\mathsf{G}/\infty$ queueing model [28, Sec. 6.1.1]. Since the interarrival times are independent and homogeneous, the steady-state distribution of the number of devices within the communication range of any reference device is Poisson with mean [29]

$$\int_0^\infty \lambda[1-\mathbb{P}(T_{\mathrm{c}} \leq t)]\ dt = \lambda \mathbb{E}[T_{\mathrm{c}}] = \lambda/\mu. \tag{14}$$

A fraction n/M of the devices cache a packet of a particular file. Hence, the expected number of caching devices within the communication range is [28, Sec. 1.3.1.2]

$$\frac{\lambda}{\mu} \cdot \frac{n}{M},$$

which gives (7) and the proof is complete. ∎

For a Poisson point process, $M-1$ devices are uniformly spread over an area A. The number of devices within a communication range r follows the Poisson distribution with mean [30]

$$(M-1)\frac{\pi r^2}{A}. \tag{15}$$

Note that, for such a process, independent realizations of the device locations are assumed. For the device mobility model considered in Section II, the device locations are uniformly distributed over the sphere, but the location of a device at any given time is dependent of the location at the previous time instant. Interestingly, using (8) and (9), (15) is equal to λ/μ and the distribution of devices within the communication range of any reference device is the same according to Theorem 1.

We denote by $\mathcal{C}_i$ and $\bar{\mathcal{C}}_i$ the events that the request for file i comes from a device caching or not caching a coded packet of file i, respectively, where

$$\mathbb{P}(\mathcal{C}_i) = 1 - \mathbb{P}(\bar{\mathcal{C}}_i) = \frac{n}{M}, \quad i = 1, \ldots, N. \tag{16}$$

The amount of data that is downloaded from the BS and from the mobile devices is given by the following proposition.

Proposition 1. *Assume a device requests file i and there are j devices within its communication range caching a coded packet of file i. If the device caches a coded packet of file i, the fraction of file i that is downloaded from the BS is*

$$\gamma_{\mathrm{BS}}^{(\mathcal{C})}(i,j) = \begin{cases} 1-\alpha_i(j+1), & \text{if } 0 \leq j < 1/\alpha_i \\ 0, & \text{if } j \geq 1/\alpha_i \end{cases}. \tag{17}$$

Otherwise, the corresponding fraction of file i that is downloaded from the BS is

$$\gamma_{\mathrm{BS}}^{(\bar{\mathcal{C}})}(i,j) = \begin{cases} 1-j\alpha_i, & \text{if } 0 \leq j < 1/\alpha_i \\ 0, & \text{if } j \geq 1/\alpha_i \end{cases}. \tag{18}$$

Moreover, the fraction of file i that is downloaded from j caching devices is

$$\gamma_{\mathrm{D2D}}^{(\mathcal{C})}(i,j) = 1 - \alpha_i - \gamma_{\mathrm{BS}}^{(\mathcal{C})}(i,j), \tag{19}$$

and

$$\gamma_{\mathrm{D2D}}^{(\bar{\mathcal{C}})}(i,j) = 1 - \gamma_{\mathrm{BS}}^{(\bar{\mathcal{C}})}(i,j) \tag{20}$$

depending on whether the device requesting file i caches a packet of file i or not.

Proof: A device requires k_i coded packets to decode a requested file i. If there are $j < k_i = 1/\alpha_i$ devices caching a packet of file i within the communication range, $k_i - j$ packets are retrieved from the BS if the device placing the request does not cache a packet of file i. If the device caches a packet of file i, $k_i - j - 1$ packets are downloaded from the BS. If $j \geq k_i$, no packets are downloaded from the BS. Hence, the fraction of file i that is downloaded from the BS is given by (17) and (18). Following the same argument, one obtains (19) and (20). ∎

The expected downlink rate for a content allocation $\boldsymbol{\alpha}$, denoted by $f(\boldsymbol{\alpha})$, is given by

$$f(\boldsymbol{\alpha}) = M\omega \sum_{i=1}^{N} p_i \sum_{j=0}^{\infty} q_j \left(\gamma_{\mathrm{BS}}^{(\bar{\mathcal{C}})}(i,j)\mathbb{P}(\bar{\mathcal{C}}_i) + \gamma_{\mathrm{BS}}^{(\mathcal{C})}(i,j)\mathbb{P}(\mathcal{C}_i) \right), \tag{21}$$

where, using (16)–(18),

$$\gamma_{\mathrm{BS}}^{(\bar{\mathcal{C}})}(i,j)\mathbb{P}(\bar{\mathcal{C}}_i) + \gamma_{\mathrm{BS}}^{(\mathcal{C})}(i,j)\mathbb{P}(\mathcal{C}_i) = \begin{cases} 1-\alpha_i\left(j+\frac{n}{M}\right), & \text{if } 0 \leq j < 1/\alpha_i \\ 0, & \text{if } j \geq 1/\alpha_i \end{cases}. \tag{22}$$

The expected D2D communication rate, denoted by $g(\boldsymbol{\alpha})$, is given by

$$g(\boldsymbol{\alpha}) = M\omega \sum_{i=1}^{N} p_i \sum_{j=0}^{\infty} q_j \left(\gamma_{\mathrm{D2D}}^{(\bar{\mathcal{C}})}(i,j)\mathbb{P}(\bar{\mathcal{C}}_i) + \gamma_{\mathrm{D2D}}^{(\mathcal{C})}(i,j)\mathbb{P}(\mathcal{C}_i) \right), \tag{23}$$

where, using (16)–(20),

$$\gamma_{\mathrm{D2D}}^{(\bar{\mathcal{C}})}(i,j)\mathbb{P}(\bar{\mathcal{C}}_i) + \gamma_{\mathrm{D2D}}^{(\mathcal{C})}(i,j)\mathbb{P}(\mathcal{C}_i) = \begin{cases} j\alpha_i, & \text{if } 0 \leq j < 1/\alpha_i \\ 1-\alpha_i\frac{n}{M}, & \text{if } j \geq 1/\alpha_i \end{cases}. \tag{24}$$

IV. Minimizing the Weighted Communication Rate

We consider the optimization of the content allocation $\boldsymbol{\alpha}$. Similar to the work in [6], where the average delay, formulated as a linear scalarization of the macro BS and small BS download delays, we minimize the *weighted communication rate*

$$h(\boldsymbol{\alpha}) \triangleq \theta f(\boldsymbol{\alpha}) + (1-\theta) g(\boldsymbol{\alpha}) \tag{25}$$

for some given θ, $0.5 \leq \theta \leq 1$. Note that the communication rate is directly related to the download delay considered in [6]. Minimizing the expected downlink rate corresponds to $\theta = 1$. However, it might be desirable to also limit the D2D communication rate for various reasons, such as device battery constraints and interference between devices. Therefore, we

$$h(\boldsymbol{\alpha}) = M\omega \sum_{i=1}^{N} p_i \sum_{j=0}^{\infty} q_j \max\left\{\alpha_i \left((1-2\theta)j - \theta\frac{n}{M}\right) + \theta, (1-\theta)\left(1-\alpha_i\frac{n}{M}\right)\right\} \tag{26}$$

consider $0.5 \leq \theta \leq 1$, where $\theta \geq 0.5$ stems from the fact that the bottleneck is in the downlink. The weighted communication rate (25) can be rewritten as shown in (26) at the top of the page using (21)–(24). The minimization of the weighted communication rate (25) in terms of the content allocation can then be formulated as the following optimization problem

$$\underset{\alpha_i \in \mathcal{A}}{\text{minimize}}\ h(\boldsymbol{\alpha}) \tag{27a}$$

$$\text{subject to}\ \sum_{i=1}^{N} \alpha_i \leq \beta. \tag{27b}$$

We denote by $\boldsymbol{\alpha}^*$ the *optimal* content allocation resulting from (27). In the following theorem, we rewrite the optimization problem (27) in an equivalent form that is tractable.

Theorem 2. *Problem (27a)–(27b) is equivalent to the MILP*

$$\underset{\substack{\alpha_i \in \mathbb{R} \\ t_{ij} \in \mathbb{R} \\ b_{i\ell} \in \{0,1\}}}{\text{minimize}}\ \sum_{i=1}^{N}\sum_{j=0}^{\infty} t_{ij} \tag{28a}$$

$$\text{subject to}\ \sum_{i=1}^{N} \alpha_i \leq \beta \tag{28b}$$

$$t_{ij} + p_i q_j \left((2\theta-1)j + \theta\frac{n}{M}\right)\alpha_i \geq \theta p_i q_j \tag{28c}$$

$$t_{ij} + (1-\theta)\frac{n}{M} p_i q_j \alpha_i \geq (1-\theta) p_i q_j \tag{28d}$$

$$\alpha_i - \sum_{\ell=1}^{n} \frac{b_{i\ell}}{n-\ell+1} = 0 \tag{28e}$$

$$\sum_{\ell=0}^{n} b_{i\ell} = 1 \tag{28f}$$

Proof: The objective function (26) is a sum of piecewise linear functions of α_i. This allows us to rewrite the optimization problem in a way that is tractable, using the epigraph formulation [31]. Using (26) and introducing a new optimization variable $t_{ij} \in \mathbb{R}$, we minimize the objective function (28a) over the optimization variables $\alpha_i \in \mathcal{A}$ and $t_{ij} \in \mathbb{R}$. The constraints (28c) and (28d), which arise from the first and second term in the max function in (26), are added to the optimization problem. Note that we drop the factor $M\omega$ in (28a) since it is irrelevant to the solution of the optimization problem. Variables $\{\alpha_i\}$ can only take on the discrete values given by (2). To handle this, we introduce the binary auxiliary optimization variable $b_{i\ell} \in \{0,1\}$ and let

$$\alpha_i = b_{i0}\cdot 0 + b_{i1}\frac{1}{n} + b_{i2}\frac{1}{n-1} + \ldots + b_{in},\ \forall i, \tag{29}$$

which constitutes constraint (28e) [32, Sec. 3.2]. To guarantee that, for each i, only one $b_{i\ell} \neq 0$, the constraint (28f) is added to the problem. We can now optimize over the variable $\alpha_i \in \mathbb{R}$ and formulate the MILP in (28) that is equivalent to (27). ■

So far, an average cache size constraint has been assumed in order to simplify the optimization problems (27) and (28). We suggest the following greedy approach to enforce a strict cache size constraint per device. For a cache size overhead, denoted by $\delta \geq 0$, Algorithm 1 enforces strictly the cache size constraint $(1+\delta)\beta_{\text{d}}$ per device. We refer to the output of the algorithm as the *strict* content allocation and denote it by $\boldsymbol{\alpha}'$.

For $N \leq 100$ and $n = M \leq 1000$ (approximately), we are able to solve the optimization problem in (28) using a branch-and-bound method with a guarantee to attain the best bound, i.e., the optimality gap goes to zero. To be able to solve for a larger code length n, i.e., potentially a larger number of devices M, we consider a relaxation of the optimization problem in (28) where the integer constraints (28e) and (28f) of the MILP (28) are replaced by the constraint $0 \leq \alpha_i \leq 1$, resulting in the LP

$$\begin{aligned}
\underset{\substack{\alpha_i \in \mathbb{R} \\ t_{ij} \in \mathbb{R}}}{\text{minimize}}\ & \sum_{i=1}^{N}\sum_{j=0}^{\infty} t_{ij} \\
\text{subject to}\ & \sum_{i=1}^{N} \alpha_i \leq \beta \\
& t_{ij} + p_i q_j \left((2\theta-1)j + \theta\frac{n}{M}\right)\alpha_i \geq \theta p_i q_j \\
& t_{ij} + (1-\theta)\frac{n}{M} p_i q_j \alpha_i \geq (1-\theta) p_i q_j \\
& 0 \leq \alpha_i \leq 1.
\end{aligned} \tag{30}$$

We denote by $\check{\boldsymbol{\alpha}}^*$ the allocation resulting from (30) and refer to it as the *integer-relaxed optimal* content allocation. Note that the weighted communication rate (25) resulting from the integer-relaxed optimal allocation is at least as good as the weighted communication rate using the optimal content allocation provided by the MILP solution (28) since the allocation obtained from (28) is also a feasible solution to (30). Hence, the weighted communication rate using the allocation obtained from (30) is a lower bound on the weighted communication rate using the allocation obtained from (28), i.e.,

$$h(\check{\boldsymbol{\alpha}}^*) \leq h(\boldsymbol{\alpha}^*). \tag{31}$$

In Section V, we observe numerically that $h(\boldsymbol{\alpha}^*) \approx h(\check{\boldsymbol{\alpha}}^*)$, i.e., the weighted communication rate resulting from the integer-relaxed content allocation $\check{\boldsymbol{\alpha}}^*$ represents well the weighted communication rate corresponding to the optimal content allocation $\boldsymbol{\alpha}^*$.

Note that a practical coding scheme must have valid values of α_i, i.e., $\alpha_i \in \mathcal{A}$. In other words, valid values of α_i are such that $k_i = 1/\alpha_i$ is integer in $[1, n]$ and therefore a code (n, k_i) can be realized. This is not guaranteed by the solution of the LP (30). To remedy this problem, we suggest a simple algorithm to ensure valid values of α_i from the values $\check{\alpha}_i^*$ resulting from (30), without violating the cache size constraint (4). The algorithm is given in Algorithm 2. We

Algorithm 1 Greedy strict cache size constraint

Input: $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_N)$, N, M, n, β_{d}, and δ
Output: $\boldsymbol{\alpha}' = (\alpha'_1, \ldots, \alpha'_N)$

$\boldsymbol{\alpha}' \leftarrow \boldsymbol{\alpha}$
$c_{ij} \leftarrow 0,\ i = 1, \ldots, N,\ j = 1, \ldots, M$
$i \leftarrow 1$
while $i \leq N$ **do**
  $\mathcal{M} \leftarrow \{1, \ldots, M\}$, $\ell \leftarrow n$
  while $\ell > 0$ **do**
    $\mathcal{L} \leftarrow \ell$ values of $\mathcal{M}$, chosen uniformly at random without replacement
    for $j \in \{j : j \in \mathcal{L}, \alpha'_i + \sum_{m=1}^{i-1} \alpha'_m c_{mj} \leq (1+\delta)\beta_{\mathrm{d}}\}$ **do**
      $c_{ij} \leftarrow 1$
    end for
    $\mathcal{M} \leftarrow \mathcal{M} \backslash \mathcal{L}$
    $\ell \leftarrow n - \sum_{j=1}^{M} c_{ij}$
    if $\ell > |\mathcal{M}|$ **then**
      break
    end if
  end while
  if $\sum_{j=1}^{M} c_{ij} < n$ **then**
    $c_{ij} \leftarrow 0,\ j = 1, \ldots, M$
    $k_i \leftarrow \frac{1}{\alpha'_i} + 1$
    if $k_i > n$ **then**
      $\alpha'_i \leftarrow 0$
    else
      $\alpha'_i \leftarrow 1/k_i$
    end if
  else
    $i \leftarrow i + 1$
  end if
end while

Algorithm 2 Round-to-integer content allocation

Input: $\check{\boldsymbol{\alpha}}^* = (\check{\alpha}_1, \ldots, \check{\alpha}_N)$, n, and N
Output: $\hat{\boldsymbol{\alpha}}^* = (\hat{\alpha}_1, \ldots, \hat{\alpha}_N)$

for $i = 1, \ldots, N$ **do**
  $\hat{\alpha}_i \leftarrow \frac{1}{\lceil 1/\check{\alpha}_i \rceil}$
  if $\hat{\alpha}_i < 1/n$ **then**
    $\hat{\alpha}_i \leftarrow 0$
  end if
end for

denote the resulting content allocation by $\hat{\boldsymbol{\alpha}}^*$ and refer to it as the *round-to-integer* content allocation. By using Algorithm 2, we are guaranteed valid values of α_i. Note that the weighted communication rate arising from the content allocation provided by Algorithm 2 is higher than or equal to the weigthed communication rate with the optimal content allocation $\boldsymbol{\alpha}^*$ obtained solving (28). Therefore, the weighted communication rate using the allocation provided by Algorithm 2 is an upper bound to the weighted communication rate using the optimal content allocation, i.e.,

$$h(\hat{\boldsymbol{\alpha}}^*) \geq h(\boldsymbol{\alpha}^*).$$

Using (31), we have

$$h(\check{\boldsymbol{\alpha}}^*) \leq h(\boldsymbol{\alpha}^*) \leq h(\hat{\boldsymbol{\alpha}}^*). \tag{32}$$

As shown in Section V, our numerical results show that, for the important case of maximal spreading, the gap between the upper and lower bounds is very small.

For the specific case of equally expensive downlink and D2D communication, i.e., $\theta = 0.5$, it is optimal to use the popular content allocation (6). The result is given by the following proposition.

Proposition 2. *For $\theta = 0.5$ and $\beta \in \mathbb{N}$, popular content allocation* (6) *is optimal.*

Proof: For $\theta = 0.5$, (26) reduces to

$$\begin{aligned} h(\boldsymbol{\alpha}) &= \frac{M\omega}{2} \sum_{i=1}^{N} p_i \sum_{j=0}^{\infty} q_j \left(1 - \alpha_i \frac{n}{M}\right) \\ &= \frac{M\omega}{2} \left(1 - \frac{n}{M} \sum_{i=1}^{N} \alpha_i p_i\right) \end{aligned}$$

and the minimization problem (27) converts to the maximization problem

$$\underset{\alpha_i \in \mathcal{A}}{\text{maximize}} \ \sum_{i=1}^{N} \alpha_i p_i \tag{33a}$$

$$\text{subject to} \ \sum_{i=1}^{N} \alpha_i \leq \beta. \tag{33b}$$

Since $p_1 \geq p_2 \geq \ldots \geq p_N$ according to (1), it is trivial to see that the sum (33a) is maximized for $\alpha_i = 1,\ i = 1, \ldots, \beta$, i.e., it is optimal to use the popular content allocation. ■

V. Numerical results

In Figs. 3–8, we evaluate the downlink rate (21) and the weighted communication rate (25) for the optimal content allocation $\boldsymbol{\alpha}^*$ obtained by solving the MILP (28), with lower and upper bounds given by the integer-relaxed optimal content allocation $\check{\boldsymbol{\alpha}}^*$ provided by the solution of the LP (30), and the round-to-integer content allocation $\hat{\boldsymbol{\alpha}}^*$ provided by Algorithm 2, respectively. Specifically, we investigate the reduction in the weighted communication rate of using the optimal content allocation over the popular content allocation, for which we consider a code length n such that $\beta \in \mathbb{N}$, i.e., the constraint (4) is attained with equality, which means that the content allocation uses all the available cache space. For the results, we consider a file library with $N = 100$ files, an area on a sphere of radius $\rho = 30$ meters, which corresponds to an area of roughly 11000 square meters, and a communication range $r = 10$ meters. These values of ρ and r are enough to satisfy the condition $r \ll 2\rho$ and provide a good approximation of the distribution of devices within the communication range in Theorem 1. This is verified by computing the Kullback-Leibler divergence between the empirical distribution of the number of caching devices within the communication range of any reference device, obtained through simulations, and the theoretical distribution provided by Theorem 1. To reflect a typical walking speed, we assume

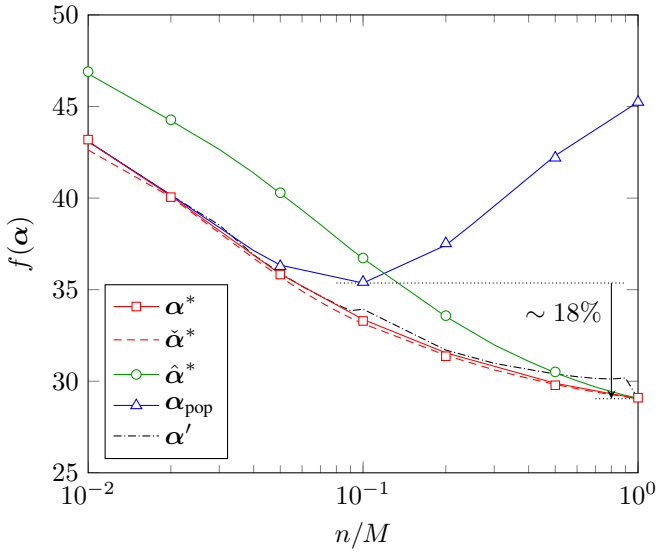

Figure 3. The downlink rate $f(\boldsymbol{\alpha})$ versus n/M using different content allocations for $M = 500$, $\sigma = 0.7$, $\theta = 1$, and $\beta_{\text{d}} = 1$. All markers correspond to simulated downlink rate.

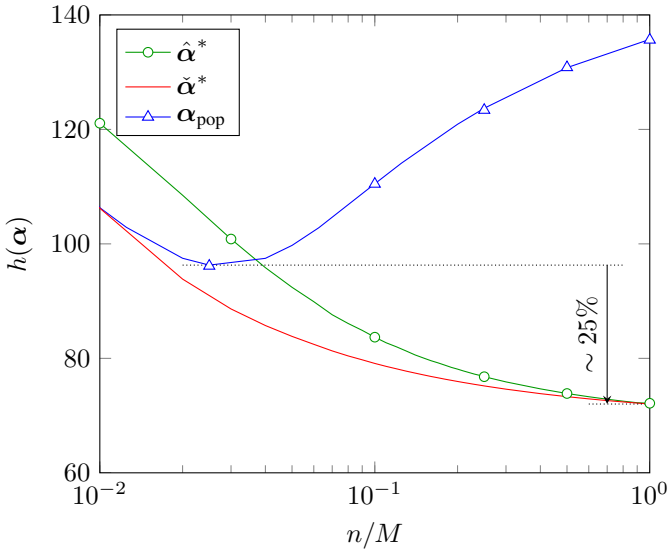

Figure 4. The weighted communication rate $h(\boldsymbol{\alpha})$ versus n/M using different content allocations for $M = 2000$, $\sigma = 0.7$, $\theta = 0.75$, and $\beta_{\text{d}} = 1$. All markers correspond to simulated rate.

a minimum and a maximum device speed of $s_{\min} = 0.3$ and $s_{\max} = 2.5$ m/s, respectively, and a request rate $\omega = 0.1$ s^{-1}. Note that the optimal content allocation does not depend on ω. In Figs. 3–7, we set $\sigma = 0.7$ motivated by the frequency of document accesses [22] and the popularity of YouTube videos under the assumption that the video popularity follows the Zipf distribution [23]. In Figs. 3–8, the markers correspond to simulation results and it can be seen that the approximations $r \ll 2\rho$ and $s_{\min} \approx s_{\max}$ made in Theorem 1 are reasonable and that the theoretical values of the weighted communication rate accurately predict the simulated data.

Fig. 3 shows the downlink rate in (21) versus the code length n, normalized by the total number of mobile devices in the area M, for $M = 500$. Note that, for

$$\beta = \frac{\beta_{\text{d}} M}{n} > N,$$

a device has the capacity to cache more than the entire library, which is inefficient since there are no more files to cache. Hence, we consider only $n/M \geq \beta_{\text{d}}/N = 10^{-2}$. The optimal content allocation $\boldsymbol{\alpha}^*$ is obtained by solving (28) using a branch-and-bound method with a guarantee to have attained the best bound, i.e., the optimality gap goes to zero. Also included in Fig. 3 is the downlink rate when using the strict content allocation resulting from Algorithm 1 (dashdotted black curve) with the optimal content allocation as input for an overhead $\delta = 0$. We observe that there is only a small difference in the downlink rate when using the optimal content allocation and the integer-relaxed content allocation for all values n/M. Note that $n/M = 1$ corresponds to maximal spreading [17], i.e., storing a coded packet of a given file on as many devices as possible. We observe that the downlink rate decreases as n increases for the integer-relaxed optimal content allocation, i.e., maximal spreading [17] appears to be optimal. In Fig. 3, for $n = M$,

$$h(\check{\boldsymbol{\alpha}}^*) \approx h(\boldsymbol{\alpha}^*) \approx h(\hat{\boldsymbol{\alpha}}^*),$$

which implies that Algorithm 2 has not modified much the integer-relaxed optimal content allocation found by solving the LP (30), i.e., the integer-relaxed content allocation was already providing close-to-optimal and valid α_i. For the popular content allocation, there is a tradeoff between a small n/M (large β), i.e., a smaller fraction of the mobile devices cache more files from the library, and a large n/M. Interestingly, using the strict content allocation with $\delta = 0$ does not incur a big loss as compared to when using the optimal content allocation. In fact, for $\delta = 0.1$ (not shown in the figure), it is impossible to distinguish the downlink rate curves when using the optimal and strict content allocations. We observe from the figure that using the optimal content allocation incurs a reduction of roughly 18% as compared to when using the popular content allocation.

Fig. 4 shows the weighted communication rate versus the code length n, normalized by the number of devices M, for $M = 2000$ and $\theta = 0.75$. Note that for such a large M, and consequently large n when $n = M$, solving the MILP (28) is not feasible, and instead the LP (30) is solved. We observe that $h(\check{\boldsymbol{\alpha}}^*) \approx h(\hat{\boldsymbol{\alpha}}^*)$ for $n = M$, i.e., both are good approximations of $h(\boldsymbol{\alpha}^*)$, using (32), and that $h(\check{\boldsymbol{\alpha}}^*)$ decreases with n. The corresponding reduction resulting from using the optimal content allocation instead of the popular content allocation is around 25%.

In the subsequent figures, we only consider maximal spreading, i.e., $n = M$, for the optimal content allocation and exhaustively search for the optimal n when using the popular content allocation. Fig. 5 shows the weighted communication rate in (25) versus the density of mobile devices in the area M/A using the various content allocations. For comparison purposes, the weighted communication rate when there is no caching, $\boldsymbol{\alpha}_{\text{nc}} = \mathbf{0}_N$, is also included in the figure. Using (26), it is trivial to obtain

$$h(\boldsymbol{\alpha}_{\text{nc}}) = M\omega\theta.$$

We first note that the round-to-integer content allocation

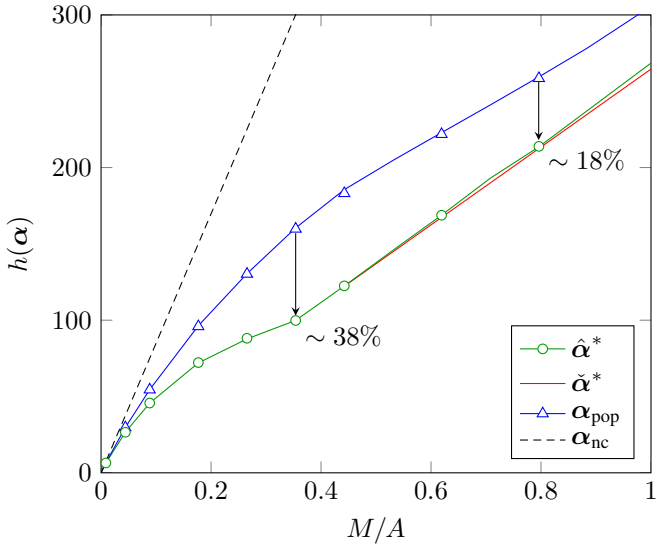

Figure 5. The weighted communication rate $h(\boldsymbol{\alpha})$ versus the device density M/A using different content allocations for $\sigma = 0.7$, $\theta = 0.75$, and $\beta_{\text{d}} = 1$. All markers correspond to simulated rate.

achieves a weighted communication rate very close to that of the integer-relaxed optimal content allocation lower bound. The infliction point observed for the optimal allocation ($M/A \approx 0.35$) corresponds roughly to the value of M for which the expected aggregate cache capacity of caching devices within range of each device exceeds the number of files in the library. Note that using (7) with $n = M$ gives that the expected number of devices within range is λ/μ which is given by (8) and (9). Since each device has the capacity to cache β_{d} files, the expected aggregate cache capacity of caching devices within range is $\beta_{\text{d}}\lambda/\mu$ and

$$\beta_{\text{d}}\frac{\lambda}{\mu} = \beta_{\text{d}}\frac{\pi}{\pi}\frac{\lambda}{\mu} = \beta_{\text{d}}(M-1)\frac{\pi r^2}{A} > N$$
$$\implies \quad \frac{M}{A} > \frac{N}{\beta_{\text{d}}}\cdot\frac{1}{\pi r^2} + \frac{1}{A} \approx \frac{N}{\beta_{\text{d}}}\cdot\frac{1}{\pi r^2} \approx 0.32,$$

where the first approximation holds for a large area A. We see that, using the optimal content allocation, we can effectively leverage the available cache size and reduce the weighted communication rate by around 38% compared to when using the popular content allocation. For a larger density of devices ($M/A \approx 0.8$), the difference in weighted communication rate is reduced since the expected aggregate cache capacity within range of any reference device is very large. In this case, a device requesting a particular file is likely to find coded packets cached by devices within the communication range also when using the popular content allocation. Despite this fact, the reduction in the weighted communication rate of using the optimal content allocation instead of the popular content allocation is still around 18%.

In Fig. 6, the weighted communication rate is plotted versus the weighting parameter θ. As explained by Proposition 2, the popular content allocation is optimal for $\theta = 0.5$. The gain over no caching, i.e., all files are downloaded from the BS, observed for $\theta = 0.5$ is due to devices self-servicing, i.e., finding requested content in the own cache. For the popular content allocation and $\theta < 0.54$, maximal spreading, i.e., a smaller number of files cached by all devices, entails a

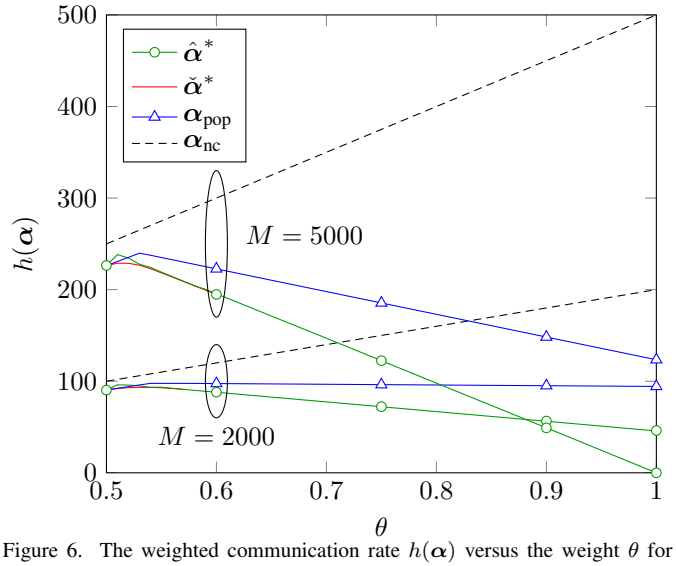

Figure 6. The weighted communication rate $h(\boldsymbol{\alpha})$ versus the weight θ for $\sigma = 0.7$ and $\beta_{\text{d}} = 1$. All markers correspond to simulated rate.

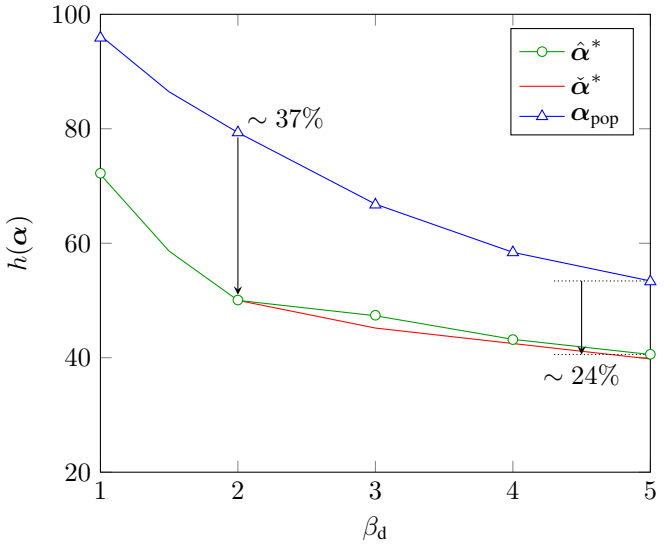

Figure 7. The weighted communication rate $h(\boldsymbol{\alpha})$ versus the average cache size constraint per device β_{d} using different content allocations for $M = 2000$, $\sigma = 0.7$, $\theta = 0.75$. All markers correspond to simulated rate.

lower weighted communication rate. For larger θ, a reduced spreading is desirable. For $M = 2000$ and $0.54 \leq \theta < 0.58$, $n/M = 0.04$ is found to be optimal. For $\theta \geq 0.58$, an exhaustive search reveals that $n/M = 0.025$ is optimal. We also see that the reduction in the weighted communication rate entailed by the optimal content allocation instead of the popular content allocation increases with θ.

Fig. 7 shows the weighted communication rate versus the average cache size constraint per device β_{d}. Recall that, for the optimal content allocation with maximal spreading, the cache constraint is strict. We see that using the optimal content allocation instead of the popular content allocation entails a significant reduction in the weighted communication rate for some β_{d}. As $\beta_{\text{d}} \to N$ the reduction vanishes, which is intuitive as each device can cache the entire file library and selecting a content allocation is no longer relevant.

Finally, in Fig. 8, the weighted communication rate is plotted versus the Zipf parameter σ. Note that, for $\sigma \to 0$, the

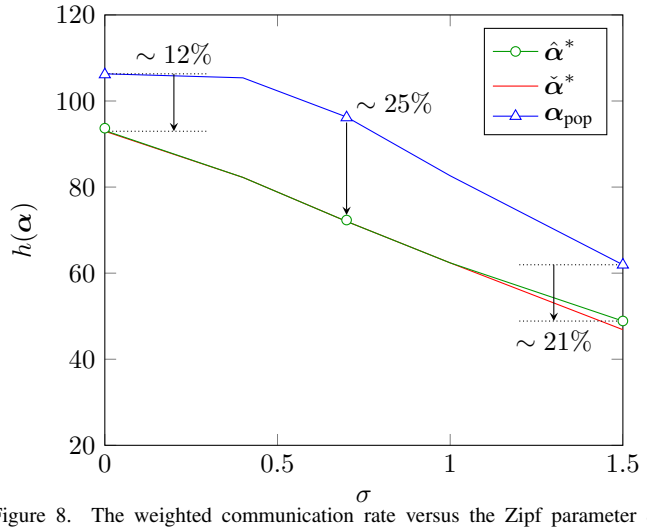

Figure 8. The weighted communication rate versus the Zipf parameter σ using different content allocations for $M = 2000$, $\theta = 0.75$, and $\beta_{\text{d}} = 1$. Markers correspond to simulated rate.

file popularity distribution approaches the uniform distribution. For $\sigma = 0$, as expected, uniform content allocation, i.e., $\alpha_i = \alpha \; \forall i$, is optimal. In other words, all files are cached using the same (n, k) MDS code. Using the round-to-integer content allocation instead of the popular content allocation leads to a reduction of the weighted communication rate of around 12%, which is due to the more efficient use of the cache space when using the former allocation, i.e., the probability of redundant content being cached in devices within the communication range is negligible. For larger σ the distribution is more skewed towards the most popular files and the weighted communication rate decreases. The reason is that less files have a notable popularity, less files are frequently requested, and the fixed cache size constraint allows these few popular files to be cached. In this case the optimal content allocation deviates from the uniform one. For $\sigma = 1.5$, the weighted communication rate is decreased by approximately 21% when the round-to-integer content allocation is used instead of the popular content allocation. This is because the round-to-integer content allocation uses some of the available cache space to cache coded packets from the tail of less frequently requested files that cumulatively adds up to a non-negligible fraction of the requests.

VI. Conclusion

We optimized the caching of content in mobile devices using maximum distance separable codes. We derived a good approximation of the distribution of the number of caching devices within range of a device as the devices move around according to the random waypoint model. We formulated a mixed integer linear program to minimize the weighted sum of the downlink rate and the device-to-device communication rate under a global average cache size constraint. We showed that optimized MDS coded caching yields a significantly lower weighted communication rate compared to when caching (uncoded) the most popular files, especially when the device density is high. Furthermore, we showed numerically that caching coded packets of a particular file on all devices, i.e., maximal spreading, is optimal.

References

[1] "Ericsson mobility report," White Paper, Ericsson, Jun. 2017.

[2] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[3] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[4] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: design aspects, challenges, and future directions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 22–28, Sep. 2016.

[5] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, Feb. 2014.

[6] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.

[7] M. Ji, G. Caire, and A. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849–869, Feb. 2016.

[8] V. Bioglio, F. Gabry, and I. Land, "Optimizing MDS codes for caching at the edge," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, San Diego, CA, 2015.

[9] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3665–3676, Jul. 2014.

[10] J. Pedersen, A. Graell i Amat, I. Andriyanova, and F. Brännström, "Repair scheduling in wireless distributed storage with D2D communication," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Jeju, Korea, 2015, pp. 69–73.

[11] ——, "Distributed storage in mobile wireless networks with device-to-device communication," *IEEE Trans. Commun.*, vol. 64, no. 11, pp. 4862–4878, Nov. 2016.

[12] A. Piemontese and A. Graell i Amat, "MDS-coded distributed storage for low delay wireless content delivery," in *Proc. 2016 9th Int. Symp. Turbo Codes & Iterative Inform. Process. (ISTC)*, Brest, France, 2016, pp. 320–324.

[13] ——, "MDS-coded distributed caching for low delay wireless content delivery," *IEEE Trans. Commun.*, 2018.

[14] J. Pääkkönen, C. Hollanti, and O. Tirkkonen, "Device-to-device data storage for mobile cellular systems," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Atlanta, GA, 2013.

[15] R. Wang, J. Zhang, S. H. Song, and K. B. Letaief, "Mobility-aware caching in D2D networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 5001–5015, Aug. 2017.

[16] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, ser. The Kluwer International Series in Engineering and Computer Science. Boston, MA: Springer, 1996, vol. 353, pp. 153–181.

[17] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocations," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4733–4752, Jul. 2012.

[18] C. Kosta, B. Hunt, A. U. Quddus, and R. Tafazolli, "On interference avoidance through inter-cell interference coordination (ICIC) based on ofdma mobile systems," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 973–995, Third Quarter 2013.

[19] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[20] K. Li, C. Yang, Z. Chen, and M. Tao, "Optimization and analysis of probabilistic caching in N-tier heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 2, pp. 1283–1297, Feb. 2018.

[21] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, Jan. 2016.

[22] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proc. IEEE 18th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, New York, NY, 1999, pp. 126–134.

[23] X. Cheng, C. Dale, and J. Liu, "Statistics and social network of youtube videos," in *16th Int. Workshop Quality Service*, Enschede, The Netherlands, 2008.

[24] W. E. Ryan and S. Lin, *Channel codes: Classical and modern*. Cambridge University Press, 2009.

[25] S. L. Miller and D. Childers, *Probability and random processes*. Elsevier, 2004.

[26] W. Sun, E. G. Ström, F. Brännström, K. C. Sou, and Y. Sui, "Radio resource management for D2D-based V2V communication," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6636–6650, Aug. 2016.

[27] M. Abdulla and R. Simon, "Characteristics of common mobility models for opportunistic networks," in *Proc. 2nd ACM Workshop Performance Monitoring Measurement Heterogeneous Wireless Wired Networks*, Chania, Greece, 2007, pp. 105–109.

[28] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. Wiley-Interscience, 2006.

[29] G. F. Newell, "The M/G/∞ queue," *J. SIAM Appl. Math.*, vol. 14, no. 1, pp. 86–88, Jan. 1966.

[30] M. Haenggi, *Stochastic Geometry for Wireless Networks*. Cambridge University Press, 2013.

[31] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2009.

[32] G. Appa, L. Pitsoulis, and H. P. Williams, *Handbook of Modelling for Discrete Optimization*. Springer, 2006.