THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Online Content Popularity Prediction and Learning in Wireless Edge Caching

OPEN ACCESS

Online Content Popularity Prediction and Learning in Wireless Edge Caching

Navneet Garg, *Member, IEEE*, Mathini Sellathurai, *Senior Member, IEEE*, Vimal Bhatia, *Senior Member, IEEE*, B. N. Bharath, *Member, IEEE*, Tharmalingam Ratnarajah, *Senior Member, IEEE*

Abstract—Caching popular contents in advance is an important technique to achieve low latency and reduce the backhaul costs in future wireless communications. Considering a network with base stations distributed as a Poisson point process, optimal content placement caching probabilities are obtained to maximize the average success probability (ASP) for a known content popularity (CP) profile, which in practice is time-varying and unknown in advance. In this paper, we first propose two online prediction (OP) methods for forecasting CP viz., popularity prediction model (PPM) and Grassmannian prediction model (GPM), where the unconstrained coefficients for linear prediction are obtained by solving constrained non-negative least squares. To reduce the higher computational complexity per online round, two online learning (OL) approaches viz., weighted-follow-the-leader and weighted-follow-the-regularized-leader are proposed, inspired by the OP models. In OP, ASP difference (i.e, the gap between the ASP achieved by prediction and that by known content popularity) is bounded, while in OL, sub-linear MSE regret and linear ASP regret bounds are obtained. With MovieLens dataset, simulations verify that OP methods are better for MSE and ASP difference minimization, while the OL approaches perform well for the minimization of the MSE and ASP regrets.

Index Terms—linear prediction; caching; Poisson point process (PPP); online learning.

I. Introduction

With the continuous development of various intelligent devices such as smart vehicles, smart home appliances, mobile devices, etc, and various sized innovative application services such as news updates, high quality video feeds, software updates, etc., wireless mobile communications has been experiencing an unprecedented traffic surge with a lot of redundant and repeated information, which limits the capacity of the fronthaul and backhaul links [1]. To lower the redundant traffic, caching has emerged as an effective solution for reducing the peak data rates by pre-fetching the most popular contents in the local cache storage of the base stations (BS). In the recent years, caching at the BS is actively feasible due to the reduced cost and size of the memory [2]. In the cache enabled macro-cell networks, heterogeneous networks, D2D networks, etc [2], given a set of a content library and the respective content popularity (CP) profile, content placement and delivery have been investigated in order to optimize the various performance measures like backhaul latency delay [3], server load [4], cache miss rate [5], [6], etc. With the known

N. Garg and M. Sellathurai are with Heriot-Watt University, Edinburgh, UK. V. Bhatia and B. Bettagere are with Indian Institute of Technology Indore and Dharwad. T. Ratnarajah is with The University of Edinburgh, UK. The part of this paper (section V) has been accepted in Asilomar 2019 for poster presentation.

CP profile, reinforcement learning approach [7] is presented for learning the content placement matrix. In [3], femto-caching is modeled as disjoint set cover problem. However, in practice, CP profile is time-varying and not known in advance, therefore, it needs to be estimated from the past observations of the content requests. Deep learning based prediction is employed with huge training data in [8], [9]. In [10], auto regressive (AR) prediction cache is used to predict the number of requests in the time series. Linear prediction approach is investigated for video segments in [11]. Transfer learning methods are used in [12] by leveraging content correlation and information transfer between time periods. To learn CP independently across contents, online policies are presented for cache-awareness in [13], low complexity video caching in [1], [14], user preference learning in [15], etc. These works are employed for a particular system with the fixed number of BSs and users, i.e., the statistical performance of the network as whole is lacking with respect to content delivery in the physical layer.

Parallelly, in the literature [5], [6], [16], geographical caching in the Poisson point process (PPP) network is employed for multi-cell system to maximize cache hit rate with respect to the content placement probabilities (CPPs), which represent availability of the contents at the BSs. Similarly, in [17], the area success probability and area spectral efficiency are maximized for CPPs. In these works, PPP has been a useful tool to assess the performance of a given network. Therefore, it is important to understand the caching performance variations with respect to time [18]. The above existing works with PPP assume the CP profile to be known or unchanged over time. In practical scenarios, the CP changes dynamically in both time and space dimensions owing to randomness of user requests, and needs to be predicted for the efficient caching placements. Therefore, in addition to PPP analysis, we investigate the CP prediction models under dynamic scenarios, and its effect on the caching, which have not been investigated in this context to the best of the authors' knowledge.

A. Motivation and Contributions

In this paper, for the PPP network where both the BS and users are distributed as homogeneous PPP and content requests are characterized using a global CP profile, we compute the average success probability (ASP) caching measure as a function of CPs and CPPs. ASP is the probability of successful transmission of the content in the physical layer. From caching perspective, it is a measure for content placement as well as

content delivery. Further, to optimize ASP for a given CP profile, an algorithm is proposed, which reduces the ASP maximization with respect to CP and CPPs to the prediction of CPs only. To optimize ASP for the future time slots, online prediction (OP) and online learning (OL) methods are investigated for the prediction of CP profile for the next time slot. The prediction performance is measured by mean squared error (MSE), while caching is evaluated via ASP. Therefore, the challenge is to investigate the prediction approach to maximize the ASP. It is shown at the end of section III that the joint optimization of ASP and MSE is non-convex and leads to a fixed MSE reduction. Therefore, separate prediction approaches based on MSE are carried out. Towards that, for OP methods, linear popularity prediction model (PPM) and non-linear Grassmannian prediction model (GPM) are proposed, and the respective prediction MSEs and ASP differences are analyzed. The motivation behind using linear prediction is that the parameters controlling the popularity change (such as location, time, etc.) can be modeled using linear predictors [19], and are already present in the past observations. However, a constrained non-negative least squares (CNNLS) optimization is required to solve per online round. Therefore, to reduce the computational complexity, OL methods are investigated which are inspired by PPM and GPM. In OL methods, weighted follow-the-leader (FTL) and weighted follow-the-regularized-leader (FoReL) are presented and the corresponding MSE and ASP regret bounds are analyzed. The difference between OP and OL is that OP yields a linear sum of recent past observations for prediction, whereas OL provides a convex sum of all the past ones. In simulations, considering MovieLens dataset [20], the MSE, ASP and the respective regrets are verified for both OP and OL methods. It shows that OP methods are suitable when MSE and ASP difference are minimized, while for the regret minimization, OL approaches provide better results. The contribution of this paper is summarized as follows:

- For a network with PPP distributed BSs and users, we find the optimum CPPs to maximize the ASP, when the popularity distribution is known. It shows that there are three kinds of contents viz., most popular, mid-popular and least popular. To maximize the ASP, most popular content is placed in each cache, and the least popular ones should be omitted, while the mid-popular content needs strategic placement proportional to square-root of content popularity (SCP). We provide the method to find the indices of the contents of these three kinds.
- For a given CP profile, the ASP maximization with respect to CP and CPPs is reduced to the prediction of CPs only. Therefore, we start with an intuitive PPM approach. However, the novel use of unconstrained coefficients which enables to the predict any positive/negative change in CP, leads to CNNLS with additional sum constraint. This CNNLS is solved by modifying the existing fast-NNLS algorithm, which does not deal with additional constraints except non-negativity. Further, to improve the ASP whose optimum value is proportional to SCP, GPM is proposed to predict SCP.

- Since the OP methods require to solve an optimization problem per online round, to reduce the computational complexity, OL methods (weighted-FTL and weighted-FoReL) are presented in order to minimize the MSE and ASP regrets respectively. These methods are inspired by PPM and GPM respectively.
- In OP, the bounds on the ASP difference are derived which is minimized when CPs (for mid-popular contents) are close to uniform. The analysis for regret bounds show that they achieve sub-linear MSE regret and linear ASP regrets.
- These analysis for both OP and OL methods are verified via simulations for the MovieLens dataset. ASP and ASP-regret are better for GPM and weighted-FoReL respectively, whereas for MSE and MSE-regret, PPM and weighted-FTL respectively provides better performance.

B. Organization

The organization of the paper is as follows: section II describes the system model. In section III, ASP has been maximized. For time-varying popularities, the next section IV explains the online prediction methods, while the following section V presents the online learning approaches. Simulation results are provided in section VI. Section VII concludes the paper.

C. Notations

Scalars, vectors, matrices and sets are represented by lower case (a), lower case bold face ($\mathbf{a}$), upper case bold face ($\mathbf{A}$) and calligraphic $\mathcal{A}$ letters respectively. Transpose and Hermitian transpose product of matrices are denoted by $(\cdot)^T$ and $(\cdot)^\dagger$ respectively. The notations $\|\cdot\|_2$ or $\|\cdot\|$, and $\|\cdot\|_F$ denote the l_2 norm and Frobenious norm. $\mathcal{D}(\mathbf{A}_i)$ denotes a block diagonal matrix with matrices $\mathbf{A}_i$ as its block diagonal components. $[N]$ and $|\mathcal{F}|$ denote the set $\{1, \ldots, N\}$ and the cardinality of the set. $\mathcal{O}(\cdot)$ denotes the big-O notation, i.e., $f(x) = \mathcal{O}(g(x))$ iff $\frac{|f(x)|}{g(x)} \leq M$ for all x where M is positive real.

II. System Model

We consider the edge caching scenario in a cellular network, where a large number of BSs with limited cache size are spatially distributed according to a two-dimensional (2D) homogeneous PPP. For example, a dense area with small cells and moderate user mobility, a park with relatively dense crowd and high user mobility, or a stadium with ultra-dense crowd and relatively very low user mobility are the instances of a typical scenario that can be benefited from the edge caching [15]. Owing to the variations in the density and the mobility of users in different scenarios, the popularity of the contents varies with time.

Let Φ_{BS} denote the positions of base stations which are distributed as PPP with density $\lambda_{bs} > 0$ as shown in Figure 1. These BSs serve the users which are also assumed to be distributed as PPP. From Slivanyak-Mecke theorem, for stationary and homogeneity of PPP, we consider a typical user
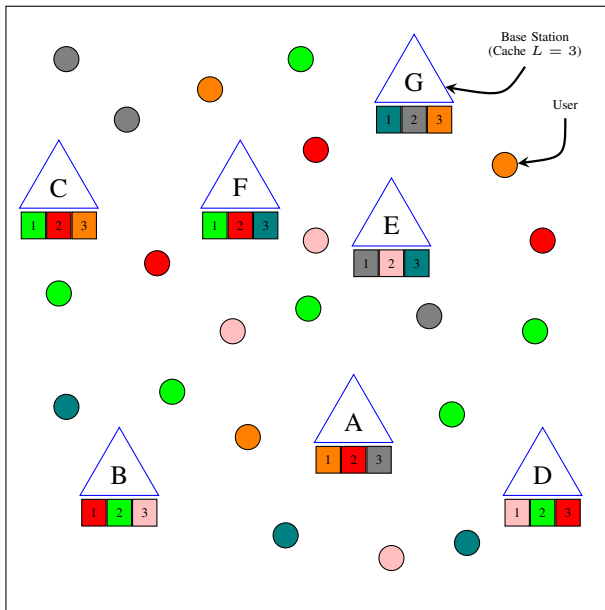
Figure 1. BSs with cache $L = 3$, and users distributed as PPP. Users color indicate their requesting content.

at the origin for evaluating the performance. Without loss of generality, it is assumed that each content has the same size[1] and each BS has the equal storage capacity to store up to L contents from the content library $\mathcal{F} := \{1, 2, \ldots, f, \ldots, N\}$. The content library may change over time and is accessed via backhaul link. Considering the user mobility, the BSs regularly monitor the users during discrete time periods $t = 1, 2, \ldots, T$, where T is the length of finite time horizon. During a discrete time period, the position of the users remain unchanged. Let $n_{t,f}$ denote the number of requests of fth file during t^{th} time period; $n_t = \sum_{f=1}^{N} n_{t,f}$ be the total number of requests; and $p_{t,f} = \frac{n_{t,f}}{n_t}$ represent the popularity of the fth file. Due to large number of requests in a time slot, $p_{t,f}$ is assumed to be the true content popularity unknown in advance and is obtained by exchanging information among BSs at the end of time slot t. For convenience, we use $\mathbf{p}_t = [p_{t,1}, \ldots, p_{t,N}]^T$ to the content popularity profile vector such that $\mathbf{p}_t^T \mathbf{1} = 1$ and $\mathbf{p}_t \geq \mathbf{0}$. The cache memory at the BS for t^{th} time period is denoted by $\mathcal{L}_t$, which is a subset of $\mathcal{F}$, such that $|\mathcal{L}_t| \leq L$. For simplicity, we adopt probabilistic (random) placement method, where each content f at time t is stored in the BS is given with the probability $q_{t,f} = \Pr[f \in \mathcal{L}_t], \forall f \in \mathcal{F}$ [5]. The probability that in t^{th} time period, a typical user finds the desired content in the cache depends on the distribution of contents in the random set $\mathcal{L}_t$ through the one-set coverage probabilities $q_{t,f}$, collected in a vector form as $\mathbf{q}_t = [q_{t,1}, \ldots, q_{t,N}]^T$. These probabilities satisfy the cache constraint $\sum_{f=1}^{N} q_{t,f} \leq L, \forall t$.

We consider the association of a BS to a user based on both the channel state information (CSI) and the cached files in the BSs. Specifically, when a user requests the fth file, it associates with the BS that has the required file and the

strongest received power. The chances that the required file at time t is available in the BS's cache, is given by $q_{t,f}$. If a file is not available in any of the caches at the BSs, it is considered as a failure event and the required file must be fetched via backhaul link. Let $\Phi_{BS}(f)$ denote the thinned PPP of the BSs whose cache has the fth file. The associated k^{th} BS transmits the fth file to the typical user with the power P over the Rayleigh fading channel, denoted by h_i. At the typical user, the received signal is given as

$$y_{f,k}[t] = \sum_{i \in \Phi_{BS}} h_i[t] P^{1/2} r_i^{-\alpha/2} x_{l,i}[t] + w[t] \qquad (1)$$

$$= h_k[t] P^{1/2} r_k^{-\alpha/2} x_{f,k}[t] + w[t] \qquad (2)$$

$$+ \sum_{i \in \Phi_{BS}(f) \setminus \{k\}} h_i[t] P^{1/2} r_i^{-\alpha/2} x_{f,i}[t]$$

$$+ \sum_{i \in \Phi_{BS}^c(f)} h_i[t] P^{1/2} r_i^{-\alpha/2} x_{l,i}[t] \qquad (3)$$

where $x_{f,k}[t]$ is the transmitted symbol for f^{th} file from the kth BS, $w[t] \sim \mathcal{CN}(0, \sigma^2)$ is the additive Gaussian noise, and α is the path loss exponent. The first term in the above equation corresponds to the desired signal, the second term pertains to the interfering transmission from the other BSs ($\Phi_{BS}(f) \setminus \{k\}$) having the fth file transmitting to other users, and the next term is for the interfering signals from BSs ($\Phi_{BS}^c(f) = \Phi_{BS} \setminus \Phi_{BS}(f)$), who do not have f^{th} file in the cache. For the above received signal model, the downlink signal to interference plus noise ratio (SINR) at the typical user can be given as

$$\Gamma_{f,k}[t] = \frac{|h_k[t]|^2 P r_k^{-\alpha}}{I_{fk}[t] + I_f^c[t] + \sigma^2}, \qquad (4)$$

where $I_{fk}[t] = \sum_{i \in \Phi_{BS}(f) \setminus \{k\}} |h_i[t]|^2 P r_i^{-\alpha}$ and $I_f^c[t] = \sum_{i \in \Phi_{BS}^c(f)} |h_i[t]|^2 P r_i^{-\alpha}$ represents the received interference power. Due to the concurrent transmissions in the PPP network where the interference terms are dominant, it is essential to ensure the successful reception of the fth file. Therefore, from the user's perspective, to maintain a quality of service and measure the caching performance, we consider average success probability, which is defined as the probability that the achievable rate of a typical user exceeds the rate requirements R_0. The ASP can be written as

$$P(\mathbf{p}_t, \mathbf{q}_t) = \sum_{f \in \mathcal{F}} p_{t,f} \mathbb{E}_{k \in \Phi_{BS}} \Pr \{W \log_2 (1 + \Gamma_{f,k}[t]) \geq R_0\}, \qquad (5)$$

where W is the transmission bandwidth.

In the above formulation, the popularity profile ($\mathbf{p}_t$) denotes the global popularity across all the BSs, and the caching probabilities ($\mathbf{q}_t$) represent the probabilistic status of the caches at the BSs in the network. In practice, the cache placement at time $t+1$, which depends on the caching probabilities ($\mathbf{q}_{t+1}$), is decided by the present content popularity ($\mathbf{p}_t$). However, the success probability in time $t + 1$ apparently depends on the popularity in the time period $t+1$. Therefore, it is essential to accurately predict the future content popularity in order to maximize the success probability.

___

[1]Contents of different sizes can always be split into data segments of same size [15], [21].

As BSs monitors the mobility of the users and tracks the popularities, we assume that the future popularity at time $t+1$ is related to the past ones by the following relation

$$\mathbf{p}_{t+1} = \psi_t(\mathbf{p}_1, \ldots, \mathbf{p}_t) \tag{6}$$

where ψ_t is an unknown function. The corresponding optimum cache placement probabilities can be computed as some function of present and the past content popularities as

$$\mathbf{q}_{t+1}^* = \varphi_t(\mathbf{p}_{t+1}; \mathbf{p}_1, \ldots, \mathbf{p}_t). \tag{7}$$

Our objective in this paper is to maximize the ASP at time $t+1$ with respect to the content popularity profile and the caching probabilities

$$(P0) \max_{\psi_t, \varphi_t} P(\mathbf{p}_{t+1}, \mathbf{q}_{t+1}) \tag{8}$$

subject to (6), (7),

where at time $t+1$, the above ideal ASP can be determined using the popularity profile $\mathbf{p}_{t+1}$ and the respective caching $\mathbf{q}_{t+1}$, if both are known perfectly in advance, which is not possible in practice. Let $\hat{\mathbf{p}}_{t+1} = \hat{\psi}_t(\mathbf{p}_1, \ldots, \mathbf{p}_t)$ and $\hat{\mathbf{q}}_{t+1} = \varphi_t(\hat{\mathbf{p}}_{t+1}; \mathbf{p}_1, \ldots, \mathbf{p}_t)$ be the estimated future popularity and the corresponding placement probabilities. Therefore, we choose to maximize the estimated ASP as

$$(P1) \max_{\hat{\psi}_t, \varphi_t} P(\hat{\mathbf{p}}_{t+1}, \hat{\mathbf{q}}_{t+1}). \tag{9}$$

In the time t, where the true content popularity is $\mathbf{p}_t$, the achievable ASP can be given as $P(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1}(\hat{\mathbf{p}}_{t+1}))$. To measure the discrepancy in the prediction and the optimization, we define the MSE, the observed and expected ASP differences respectively as follows

$$\mathcal{E}_t(\hat{\mathbf{p}}_t) = \|\mathbf{p}_t - \hat{\mathbf{p}}_t\|_2, \tag{10}$$
$$\delta_t(\hat{\mathbf{p}}_t) = P(\mathbf{p}_t, \mathbf{q}_t) - P(\mathbf{p}_t, \hat{\mathbf{q}}_t(\hat{\mathbf{p}}_t)) \geq 0, \tag{11}$$
$$\Delta_t(\hat{\mathbf{p}}_t) = P(\mathbf{p}_t, \mathbf{q}_t) - P(\hat{\mathbf{p}}_t, \hat{\mathbf{q}}_t) \geq 0. \tag{12}$$

In above, the observed ASP difference is the ASP difference which is measured with respect to true popularity; while the expected one is related to predicted ASP. The utility of the former one is to analyze the caching theoretically, while the latter is useful for the prediction as well. Since the placement probabilities are function of the content popularities, in the following, we first seek the optimum caching probabilities given the content profile. Subsequently, we focus on the prediction employing two classes of methods viz., OP and OL.

III. AVERAGE SUCCESS PROBABILITY (ASP) MAXIMIZATION

In a given time slot, the placement probabilities ($\mathbf{q}_t$) depends on the content popularities ($\mathbf{p}_t$). Therefore, for the given CP profile, we find the optimal placement probabilities to maximize the ASP. Towards this, the following result presents the ASP expression in terms of $\mathbf{p}_t$ and $\mathbf{q}_t$. For clarity of notation, we drop subscript t in this section.

Theorem 1. *Average success probability of a typical user requesting f^{th} file with popularity p_f and caching probability q_f is given as*

$$P(\mathbf{p}, \mathbf{q}) = \sum_{f=1}^{N} p_f g(q_f),$$

where $g(q) = qC \int_0^\infty (dr^2) \exp\left[-(qA + (1-q)B + qC) r^2 -s_0 r^\alpha \left(\frac{\sigma^2}{P}\right) \right]$, $s_0 = 2^{\frac{R_0}{W}} - 1$, $C = \pi\lambda_{bs}$, $A = 2\pi\lambda_{bs} s_0^{\frac{2}{\alpha}} \frac{1}{\alpha} \int_{\frac{1}{s_0}}^\infty \left(\frac{u^{\frac{2}{\alpha}-1}}{1+u} \right) du$, $B = 2\pi\lambda_{bs} s_0^{\frac{2}{\alpha}} \frac{1}{\alpha} \int_0^\infty \left(\frac{u^{\frac{2}{\alpha}-1}}{1+u} \right) du$.

Proof: Proof is given in Appendix-A. ∎

Since heterogeneous networks are usually interference limited, it is reasonable to neglect the noise i.e., $\sigma^2 = 0$. For this case, the corollary below simplifies the ASP.

Corollary 2. *For interference limited case, i.e., $\sigma^2 = 0$ or at high SNR, the ASP is simplified as $P_s(\mathbf{p}, \mathbf{q}) = \sum_f p_f g_0(q_f)$, where*

$$g_0(q) = qC \int_0^\infty \exp\left[-(qA + (1-q)B + qC) r^2 \right] (dr^2)$$
$$= \frac{qC}{qA + (1-q)B + qC}. \tag{13}$$

Now, given the content popularity profile ($\mathbf{p}$), the next step is to compute the placement probabilities to maximize the ASP expression above. The ASP maximization problem can be expressed as

$$(P3) \max_{\mathbf{q}} P_s(\mathbf{p}, \mathbf{q}) \tag{14}$$

$$\text{subject to } \mathbf{q}^T \mathbf{1} \leq L \tag{15}$$

which is a simplified problem of (8) and (9). The above problem is convex, however, to be solvable in CVX tool [22], the above problem can be cast in semi-definite program (SDP) as

$$\min_{\mathbf{q}^T\mathbf{1} \leq L, \mathbf{t}} -\mathbf{t}^T \mathbf{p} \tag{16}$$

$$s.t. \begin{bmatrix} q_f(A - B + C) + B & 1 \\ 1 & \left(\frac{A-B+C}{BC} t_f - \frac{1}{B}\right) \end{bmatrix} \preceq \mathbf{0}, \forall f$$

where $\frac{q_f C}{q_f A + (1-q_f)B + q_f C} \geq t_f$ has been simplified using Schur's lemma [22]. Analytically, the expression of the solution is presented in the following theorem.

Theorem 3. *The solution of the maximization problem in (14) is given as*

$$q_f^* = \arg \max_{\mathbf{q}^T\mathbf{1} \leq L} P_s(\mathbf{p}, \mathbf{q}) \tag{17}$$

$$= \begin{cases} 1, & f \in \mathcal{R} \\ \left(\frac{B}{A+C-B}\right)\left[\frac{\eta\bar{p}_f}{\mathbf{1}^T\bar{\mathbf{p}}_\mathcal{P}} - 1\right], & f \in \mathcal{P} \\ 0, & f \in \mathcal{Z} \end{cases} \tag{18}$$

where $\bar{\mathbf{p}} = \sqrt{\mathbf{p}}$, $\mathcal{Z} = \{i|q_i = 0\}$, $\mathcal{R} = \{i|q_i = 1\}$, and $\mathcal{P} = \mathcal{F} \setminus \{\mathcal{Z} \cup \mathcal{R}\} = \{i|0 < q_i < 1\}$. The corresponding ASP is obtained as

$$P_s(\mathbf{p}) = P_s(\mathbf{p}, \mathbf{q}^*) = \left(\frac{C}{A + C - B}\right) \bar{\mathbf{p}}^T \bar{\mathbf{Z}} \bar{\mathbf{p}}$$

$$= \mathbf{p}^T \mathbf{a}\left(\mathcal{R}, \mathcal{P}\right) - \left(\frac{C}{A+C-B}\right)\frac{\left(\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}\right)^2}{\eta} \qquad (19)$$

where $\mathbf{a}_{\mathcal{P}}$ *denotes the sub-vector of* $\mathbf{a}$ *with the entries given by* $\mathcal{P}$, $\bar{\mathbf{Z}} = \mathcal{D}\left(\frac{A+C-B}{A+C}\mathbf{I}_{|\mathcal{R}|}, \mathbf{Z}_{\mathcal{P}}, \mathbf{0}_{|\mathcal{Z}|}\right)$, $\mathbf{Z}_{\mathcal{P}} = \mathbf{I}_{|\mathcal{P}|} - \frac{\mathbf{1}\mathbf{1}^T}{\eta}$, $\eta = |\mathcal{P}| + (L - |\mathcal{R}|)\frac{A-B+C}{B}$, *and* $\mathbf{a}^T = \left[\frac{C}{A+C}\mathbf{1}_{|\mathcal{R}|}^T, \frac{C}{A+C-B}\mathbf{1}_{|\mathcal{P}|}^T, \mathbf{0}_{|\mathcal{Z}|}^T\right]$.

Proof: Proof is given in Appendix-B. ∎

The above expression shows that when the popularity of the fth content is high ($p_f \to 1$) i.e. for most popular contents, its caching probability is one i.e. f^{th} content should be stored at each BS's cache. On the other hand, when the contents are least popular ($p_f \to 0$), it is reasonable not to cache the content at any of the cache storages. For the mid-popular contents $0 < p_f < 1$, the expression of q_f^* depends on the index sets ($\mathcal{Z}$ and $\mathcal{R}$), which can be obtained from KKT conditions in the proof of Theorem 3. The procedure to get these sets are presented in Algorithm 1. In the first part of the algorithm, the set $\mathcal{R}$ is obtained by individually checking the fth content popularity for $q_f^* = 1$ in the decreasing order of popularities, i.e., checking $v_f > 0$, we have $\frac{C}{B}\left[\left(\frac{B}{A+C}\right)^2 p_f - \frac{\left(\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}\right)^2}{\eta^2}\right] > 0, \implies \frac{B}{A+C}\bar{p}_j > \frac{\mathbf{1}^T \bar{\mathbf{p}}_{\mathcal{P}}}{\eta}$. In the later part, similarly the remaining content popularities in ascending order for $q_f^* = 0$ yields the set $\mathcal{Z}$, i.e., for $w_f < 0$, we have $\frac{C}{B}p_f - \frac{C}{B}\frac{\left(\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}\right)^2}{\eta^2} > 0, \implies \bar{p}_f > \frac{\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}}{\eta}$. The last set is obtained as $\mathcal{P} = \mathcal{F} \setminus \{\mathcal{Z} \cup \mathcal{R}\}$.

Now, with the caching probabilities obtained for a PPP network, a random caching strategy is utilized to place the content in individual caches [5]. Thus, with the caching prob-

---

Algorithm 1 Getting Index sets for ASP maximization

Input: Content popularity $\mathbf{p}$ in the descending order
Output: the sets of indices $\mathcal{R}$, $\mathcal{Z}$ and $\mathcal{P}$
 1: Initialize $\mathcal{R} = \emptyset$, $\mathcal{P} = \mathcal{F} \setminus \mathcal{R}$, $j = 0$.
 2: **while** $\frac{B}{A+C}\bar{p}_j > \frac{\mathbf{1}^T \bar{\mathbf{p}}_{\mathcal{P}}}{\eta}$ and $j \le N$ **do**
 3: $\quad \mathcal{R} \leftarrow \mathcal{R} \cup \{j\}$, $\mathcal{P} \leftarrow \mathcal{P} \setminus \{j\}$, and $j = j + 1$
 4: **end while**
 5: Initialize $\mathcal{Z} = \emptyset$, $\mathcal{P} = \mathcal{F} \setminus \{\mathcal{R} \cup \mathcal{Z}\}$, $j = N$.
 6: **while** $\bar{p}_j > \frac{\mathbf{1}^T \bar{\mathbf{p}}_{\mathcal{P}}}{\eta}$ and $j \ge 1$ **do**
 7: $\quad \mathcal{Z} \leftarrow \mathcal{Z} \cup \{j\}$, $\mathcal{P} \leftarrow \mathcal{P} \setminus \{j\}$, and $j = j - 1$
 8: **end while**

---

abilities known as a function of content popularities, the ideal, estimated and achievable ASPs can be given as $P_s(\mathbf{p}_{t+1}) = P_s(\mathbf{p}_{t+1}, \mathbf{q}_{t+1}(\mathbf{p}_{t+1}))$, $P_s(\hat{\mathbf{p}}_{t+1}) = P_s(\hat{\mathbf{p}}_{t+1}, \hat{\mathbf{q}}_{t+1}(\hat{\mathbf{p}}_{t+1}))$, and $P_s(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1}(\hat{\mathbf{p}}_{t+1}))$ respectively. The ASP maximization problems ($P0$) simplifies to $P_s(\mathbf{p}_{t+1})$ as $\mathbf{p}_{t+1}$ is known in ($P0$). Similarly, ($P1$) can be recast as

$$\max_{\hat{\psi}_t} P_s(\hat{\mathbf{p}}_{t+1}), \qquad (20)$$

which suggests to maximize the future ASP based on the data up to the present content popularities. Therefore, it is important to accurately predict the future CP in order to cache contents in advance. Based on the past content popularities, we employ

two classes of methods to obtain the accurate prediction to further optimize the ASP. For each of the classes, we propose two models based on PPM and GPM. The motivation to use these is as follows. To maximize the ASP, the prediction should be as close to the ground truth, i.e., the mean squared error of the popularities should be minimized which leads to PPM. Further, from the Theorem 3, it can be observed that the square-root of content popularities maximize the ASP. This observation leads to GPM, which is presented in details in the next section.

Remark (Reliability Assumption for analysis): Let $\mathcal{P}_{t+1}$ and $\hat{\mathcal{P}}_{t+1}$ denote the sets of indices for CPPs, obtained by maximizing the ASP for the ideal CP $\mathbf{p}_{t+1}$ and the estimated CP $\hat{\mathbf{p}}_{t+1}$. Under the reliability assumption for analysis, we assume that ASP maximization under the estimated CP leads to the same set of indices as with the ideal ASP. This is a reasonable assumption as with the sufficient past CP observations and the respective CPPs, the index sets can be precisely estimated i.e., to cache f^{th} file in $t + 1$ on at least one of the BSs or not. Therefore, we set $\mathcal{P}_{t+1} = \hat{\mathcal{P}}_{t+1}$ and $\mathcal{R}_{t+1} = \hat{\mathcal{R}}_{t+1}$.

Remark (Feasibility of the ASP and MSE based joint optimization): The combined optimization problem for maximizing ASP in edge-caching can be cast in the most favorable form as

$$\max_{c_k} P_s\left(\sum_{k=1}^{d} c_k \mathbf{p}_{t+1-k}\right)$$

$$s.t., \left\|\mathbf{p}_i - \sum_{k=1}^{d} c_k \mathbf{p}_{i-k}\right\|^2 \le \epsilon,$$

$$\forall i = t - \tau + d + 1, \ldots, t$$

where the objective function is not convex and cannot be reduced to a linear matrix inequality (LMI). Moreover, since $\mathbf{p}_i \le \mathbf{1}$, the MSE constraint reduces to the correlation constraint $\sum_{k=1}^{d} c_k \mathbf{p}_{i-k}^T \mathbf{p}_i \ge \epsilon' = 1 - \epsilon/2 \forall i$, which is linear and can be solved trivially if sufficient CP observations ($\tau > d$) are provided. However, the solution c_k, which satisfy the linear constraint with equality, cannot improve the correlation beyond ϵ', or the MSE below the constraint ϵ. Therefore, due to these reasons, to maximize the ASP, the separate prediction approaches are proposed.

IV. ONLINE PREDICTION MODELS

In this section, two models are presented. First, a linear model is fitted on the past CP observations and the problem of obtaining regression coefficients is modeled as constrained non-negative least squares (CNNLS) with additional sum constraint. Further, for GPM model, the regression problem is formulated as a regularized CNNLS. With these models, the observed and the expected ASP difference are analyzed.

A. Popularity Prediction Model (PPM)

In this model, we approximate the present content popularity vector at time t to the linear sum of the content popularities

of the past up to time $t - 1$ as

$$\mathbf{p}_t \approx \sum_{k=1}^{d} c_{t-1,k} \mathbf{p}_{t-k}, \tag{21}$$

where $c_{t,k} \in \mathbb{R}$ for all $k = 1, \ldots, d$ are the prediction coefficients such that $\mathbf{p}_t^T \mathbf{1} = \sum_{k=1}^{d} c_{t,k} = 1$, and d is the order of the prediction. Note that the constraint $c_{t,k} \in \mathbb{R}$ is essential for the proper and accurate prediction. If $c_{t,k}$ is set to a non-negative value ($c_{t,k} \geq 0$), the equation (21) will be a convex sum which means that the variations in the fth content popularity at time t beyond $[\min_i p_{f,i}, \max_i p_{f,i}]$ cannot be predicted $\forall f \in \mathcal{F}$. Therefore, with $c_{t,k} \in \mathbb{R}$ and the known τ popularities $p_{f,(t-\tau+1)}, \ldots, p_{f,t}, \forall f$ observations, the future content popularity estimate can be obtained as

$$\hat{\mathbf{p}}_{t+1} = \sum_{k=1}^{d} c_{t,k} \mathbf{p}_{t+1-k} \tag{22}$$

where the coefficients can be given by the least squares problem subject to non-negativity constraint of the future estimate as

$$\{c_{t,k}, \forall k\} = \arg\min_{c_k \in \mathbb{R}} \sum_{i=t-\tau+d+1}^{t} \left\| \mathbf{p}_i - \sum_{k=1}^{d} c_k \mathbf{p}_{i-k} \right\|^2 \tag{23}$$

$$\text{subject to } \sum_{k=1}^{d} c_k = 1, \sum_{k=1}^{d} c_k p_{f,t+1-k} \geq 0, \forall f. \tag{24}$$

The above problem is convex and can be numerically solved using any convex solver such as CVX [22]. However, in practice, a more efficient solution can be obtained by casting the above problem as a constrained NNLS and the solution is given in Appendix-C. NNLS without the constraint has been solved using active set method via fast NNLS (FNNLS) algorithm in [23], [24]. Therefore, with the constraint, we modify FNNLS algorithm using the Karush–Kuhn–Tucker (KKT) conditions. The complexity of the this method is related to the least squares solution and the number of observations (τ) i.e. $O(d^3 N \tau)$.

The above model tries to minimize the MSE of content popularity based on previous popularity data. However, it does not contribute actively in the ASP maximization. Therefore, the observed difference between the ideal ASP and the achievable ASP is expressed as

$$\delta_{t+1}(\hat{\mathbf{p}}_{t+1}) \leq \left(\frac{C\eta^{-1} \mathbf{1}^T \mathbf{p}_{t+1,\mathcal{P}_{t+1}}}{A + C - B} \right) \left[\frac{\mathbf{1}^T \hat{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}}{\min_{f \in \mathcal{P}_{t+1}} \hat{\bar{p}}_{f,t+1}} - |\mathcal{P}_{t+1}| \right] \tag{25}$$

whose proof is given in Appendix-D. In the best case, this difference is minimized when the distribution over subset of library ($\mathcal{P}_{t+1}$) is uniform, i.e., $\frac{\hat{\bar{p}}_{f,t+1}}{\mathbf{1}^T \hat{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}} = \frac{1}{|\mathcal{P}_{t+1}|}$, where $\frac{\hat{\bar{p}}_{f,t+1}}{\mathbf{1}^T \hat{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}}$ represents the distribution function over $f \in \mathcal{P}_{t+1}$. This results $P_s(\mathbf{p}_{t+1}) - P_s(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1}) \leq 0$.

However, $P_s(\mathbf{p}_{t+1}) \geq P_s(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1})$ by definition. Thus, $P_s(\mathbf{p}_{t+1}) = P_s(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1})$, i.e., the estimation of partial uniform CP ($f \in \mathcal{P}_{t+1}$) achieves the best ASP. In the worst case, the observed ASP upper bound is large when $\hat{\bar{p}}_{f,t+1} \to 0$ reaches to a small non-zero value.

Let $\hat{\mathbf{p}}_t = \mathbf{p}_t + \mathbf{e}_t$, where $\mathbf{e}_t$ is a random error vector and $\mathbf{p}_t$ is defined by (21). Thus, we approximate

$$\hat{\bar{p}}_{f,t} = (p_{f,t} + e_{f,t})^{1/2} = \bar{p}_{f,t} \left(1 + \frac{e_{f,t}}{p_{f,t}} \right)^{1/2}$$

$$\approx \bar{p}_{f,t} \left(1 + \frac{e_{f,t}}{2p_{f,t}} \right) = \bar{p}_{f,t} + \tilde{e}_{f,t},$$

where $\tilde{e}_{f,t} = \frac{e_{f,t}}{2\bar{p}_{f,t}}$. Similarly, the expected difference between the ideal ASP and the estimated ASP can be defined from (19) as

$$\Delta_{t+1}(\hat{\mathbf{p}}_{t+1}) = P_s(\mathbf{p}_{t+1}) - P_s(\hat{\mathbf{p}}_{t+1}) = -\mathbf{e}_{t+1}^T \mathbf{a}(\mathcal{R}_{t+1}, \mathcal{P}_{t+1})$$
$$+ \frac{C\eta^{-1}}{A + C - B} \left[\left(\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}^T \mathbf{1} \right)^2 - \left(\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}^T \mathbf{1} \right)^2 \right]$$
$$\approx -\mathbf{e}_{t+1}^T \mathbf{a}(\mathcal{R}_{t+1}, \mathcal{P}_{t+1}) \tag{26}$$
$$+ \frac{C\eta^{-1} \tilde{\mathbf{e}}_{t+1,\mathcal{P}_{t+1}}^T \mathbf{1}}{A + C - B} \mathbf{1}^T \left[\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}^T + \bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}} \right].$$

From the observed and the expected difference above, it can be observed that the ASP difference for PPM is composed of the first order prediction error term and the squared difference term. The first error term which is a random vector with zero mean, can be minimized by using the sufficient observations (τ). The second term corresponds to the difference between squared sum of squared root popularities, which cannot be reduced to zero with the current model, as PPM is not tailored to minimize the prediction error between squared roots. Therefore, we investigate the prediction model which considers the square root of popularities in the following.

B. Grassmannian Prediction Model (GPM)

From the optimization problem in (18), it can be observed that the positive square root of the caching probabilities maximizes ASP. The positive square root of the content popularity vector ($\bar{\mathbf{p}}$) represents a line in the Grassmannian manifold $\mathcal{G}_{N,1}$ [25], [26]. Similar to PPM, here, we model the current SCP vector as a linear sum of the previous d SCP vectors as

$$\bar{\mathbf{p}}_t \approx \sum_{k=1}^{d} z_k \bar{\mathbf{p}}_{t-k}, \tag{27}$$

where $z_k \in \mathbb{R} \forall k$ are the coefficients, used to predict the future estimate of SCP as follows

$$\hat{\bar{\mathbf{p}}}_{t+1} = \sum_{k=1}^{d} z_k \bar{\mathbf{p}}_{n+1-k}. \tag{28}$$

These coefficients can be obtained using the least squares minimization subject to regularization constraint $\|\bar{\mathbf{p}}_t\| \leq 1$

as

$$\{z_k \forall k\} = \arg\min_{z_k \in \mathbb{R}} \sum_{i=t-\tau+d+1}^{t} \left\| \bar{\mathbf{p}}_i - \sum_{k=1}^{d} z_k \bar{\mathbf{p}}_{i-k} \right\|_2^2 \qquad (29)$$

$$\text{s.t.} \left\| \sum_{k=1}^{d} z_k \bar{\mathbf{p}}_{t+1-k} \right\|_2 \leq 1, 0 \leq \sum_{k=1}^{d} z_k \bar{p}_{f,t+1-k} \leq 1, \forall f,$$

which is a convex optimization problem and can be solved using CVX tool. However, for an efficient solution, the above problem can be formulated as constrained NNLS as in the previous subsection and can be solved similarly like in the Appendix-C. To avoid redundancy of the content, we omit the details.

In the above, the SCP prediction is intended to maximize the ASP. It can be seen from the observed ASP difference in (25) that as the estimation error is improved in the SCP vectors, the ASP difference decreases, i.e., the achievable ASP of GPM is better than that of PPM. Towards this, let $\hat{\bar{\mathbf{p}}}_t = \bar{\mathbf{p}}_t + \bar{\mathbf{e}}_t$ where $\bar{\mathbf{e}}_t$ is a random error vector, and approximate

$$\hat{p}_{f,t} = (\bar{p}_{f,t} + \bar{e}_{f,t})^2$$
$$\approx p_{f,t} + 2\bar{e}_{f,t}\bar{p}_{f,t} = p_{f,t} + \underaccent{\sim}{e}_{f,t}$$

with $\underaccent{\sim}{e}_{f,t} = 2p_{f,t}\bar{e}_{f,t}$. For GPM described by (28), the expected difference using (26) can be similarly written as

$$\Delta_{t+1}(\hat{\mathbf{p}}_{t+1}) = P_s(\mathbf{p}_{t+1}) - P_s(\hat{\mathbf{p}}_{t+1}) \approx -\underaccent{\sim}{\mathbf{e}}_{t+1}^T \mathbf{a}\left(\mathcal{R}_{t+1}, \mathcal{P}_{t+1}\right)$$
$$+ \frac{C\eta^{-1}\bar{\mathbf{e}}_{t+1,\mathcal{P}_{t+1}}^T \mathbf{1}}{A+C-B}\mathbf{1}^T \left[\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}^T + \bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}\right] \qquad (30)$$

which consists of two errors terms. Comparing the above equation with (26) for PPM, it can be observed that in the first term, $\underaccent{\sim}{e}_{f,t}$ is much larger than $e_{f,t}$, while for the second term $\bar{e}_{f,t}$ is lower than $\tilde{e}_{f,t}$. Together, it concludes that the GPM improves the ASP over PPM.

Remark: In OP models, each round requires to solve an independent optimization problem given τ previous observations, i.e., there is no-learning. Therefore, in each round, the resultant MSE is approximately similar, i.e., the regret measure is not essential in this case.

V. ONLINE LEARNING MODELS

In the above OP methods, the least squares optimization is required to be solved per online round, which can be computational intensive for large content library. Therefore, to reduce the cost further, we present OL methods using the weighted follow-the-leader (FTL) and weighted follow-the-regularized-leader approaches.

A. Weighted FTL

In the FTL approach, the CP estimate for time $t+1$ is obtained by minimizing the weighted sum of l_2-losses up time t as

$$\hat{\mathbf{p}}_{t+1} = \arg\min_{\mathbf{p}} \sum_{i=1}^{t} w_i \|\mathbf{p} - \mathbf{p}_i\|^2, \qquad (31)$$

where $w_i \geq 0$ for $i = 1, \ldots, t$ are the weights such that $\sum_{i=1}^{t} w_i = 1$, ensuring the sum of the predicted CPs to be one. If all the past CPs are equally important in the learning, a trivial value can be selected $w_i = \frac{1}{t}$, $1 \leq i \leq t$ for t^{th} online round. However, in general, if the recent CPs dominate the prediction, one can choose $w_i = \kappa_t a^{t-i}$, $1 \leq i \leq t$ such that κ_t is set to satisfy the sum constraint, $\kappa_t = \frac{1-a}{1-a^t}$. This yields weights at t^{th} round to be $w_t = \frac{1-a}{1-a^t}$. With $a = 1$, the trivial selection is obtained. The value of $0 < a \leq 1$ can be set according to the preferences for the recent or the past observations. The solution of the above optimization leads to the following prediction

$$\hat{\mathbf{p}}_{t+1} = \sum_{i=1}^{t} w_i \mathbf{p}_i \qquad (32)$$
$$= w_t \mathbf{p}_t + \left(\sum_{i=1}^{t-1} w_i\right) \sum_{i=1}^{t-1} \frac{w_i}{\sum_{i=1}^{t-1} w_i} \mathbf{p}_i \qquad (33)$$
$$= w_t \mathbf{p}_t + (1 - w_t)\hat{\mathbf{p}}_t, \qquad (34)$$

which is a weighted sum of the previously observed CPs. This prediction at time t consists of a balance between the observed CP and the predicted CP at time t, i.e, it forms the convex sum. This is in contrast to OP, where a linear sum is considered. It means that FTL can only predict what has been observed in the past.

The step-wise procedure for the CP prediction is listed in Algorithm 2. In t^{th} round, the prediction is obtained and the CP is observed.

Algorithm 2 Weighted FTL algorithm.

Input: the value of $0 < a \leq 1$
1: **for** for $t = 1, 2, \ldots$ **do**
2:      set $w_t = \frac{1-a}{1-a^t}$
3:      predict $\hat{\mathbf{p}}_{t+1} = w_t \mathbf{p}_t + (1 - w_t)\hat{\mathbf{p}}_t$
4:      observe $\mathbf{p}_{t+1}$
5: **end for**

Lemma 4. *For the weighted FTL approach with $0 < a < 1$, $\mathcal{O}(T)$ regret bound is obtained, and $\mathcal{O}(\log T)$ for $a = 1$.*

Proof: Equation (34) can be rewritten as

$$\hat{\mathbf{p}}_{t+1} - \mathbf{p}_t = (1 - w_t)(\hat{\mathbf{p}}_t - \mathbf{p}_t). \qquad (35)$$

Therefore, the difference with respect to the CP ($\mathbf{p}$) can be given as

$$\|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 - \|\mathbf{p} - \mathbf{p}_t\|^2$$
$$\overset{(a)}{\leq} \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 - \|\hat{\mathbf{p}}_{t+1} - \mathbf{p}_t\|^2 \qquad (36)$$
$$\overset{(b)}{=} \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 \left[1 - (1 - w_t)^2\right] \qquad (37)$$
$$\overset{(c)}{\leq} 2w_t \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 \leq 4w_t,$$

where (a) is from [27, Lem. 2.1]; (b) comes by (35); (c) is obtained by ignoring $-w_t^2$ as w_t is small; and considering $\|\hat{\mathbf{p}}_t\| = \|\mathbf{p}_t\| \leq 1$, the last inequality arises from the triangle

inequality, $\|\hat{\mathbf{p}}_t - \mathbf{p}_t\| \leq 2$. We therefore, obtain the upper bound on the total regret as

$$
\begin{aligned}
R_T(\mathbf{p}) &= \sum_{t=1}^{T} \left[\|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 - \|\mathbf{p} - \mathbf{p}_t\|^2 \right] \\
&\leq 4 \sum_{t=1}^{T} w_t = 4(1-a) \sum_{t=1}^{T} \left(1 - a^t\right)^{-1} \\
&< 4(1-a) \sum_{t=1}^{T} \left(1 + a^t\right) \\
&= 4(1-a) \left(T + a \frac{1 - a^T}{1 - a} \right)
\end{aligned}
$$

where $w_t = \frac{1-a}{1-a^t}$ is used. When $a < 1$, the $\mathcal{O}(T)$ regret bound is obtained. For $a = 1$ i.e. $w_t = t^{-1}$, we obtain the regret within $\mathcal{O}(\log T)$ as $\sum_{t=1}^{T} t^{-1} \leq \log T + 1$. ∎

Similarly, for a given CP $\mathbf{p}$, the regret in terms of ASP can be defined as

$$
R_T^{ASP}(\mathbf{p}) = \sum_{t=1}^{T} \Delta_t(\hat{\mathbf{p}}_t) - \Delta_t(\mathbf{p}),
$$

which is $\mathcal{O}(T)$ and can be seen as follows. The expected ASP difference Δ_t can be approximated with the observed ASP difference in (25), which can be seen to be independent of t, i.e. $\Delta_t(\mathbf{p})$ is $\mathcal{O}(1)$. Therefore, $R_T^{ASP}(\mathbf{p})$ is $\mathcal{O}(T)$.

B. Weighted FoReL

Analogous to GPM in OP methods where the difference between SCPs is minimized, in FoReL approach, we consider the chordal distance as a loss measure of the prediction error, since SCP lies on the Grassmannian manifold. The chordal distance is defined as the principle angle between two unit norm vectors $\bar{\mathbf{p}}_1$ and $\bar{\mathbf{p}}_2$, i.e., $d_c^2(\bar{\mathbf{p}}_1, \bar{\mathbf{p}}_2) = \sin^2 \theta_{12} = 1 - \left| \bar{\mathbf{p}}_1^\dagger \bar{\mathbf{p}}_2 \right|^2$. Minimizing the chordal distance is equivalent to maximizing the correlation, i.e., cross product $\cos \theta_{12} = \bar{\mathbf{p}}_1^\dagger \bar{\mathbf{p}}_2 > 0$. Therefore, the prediction problem with respect to the SCP norm constraint (regularization measure in FoReL terminology) can be expressed as

$$
\hat{\bar{\mathbf{p}}}_{t+1} = \arg \max_{\|\bar{\mathbf{p}}\| \leq 1} \sum_{i=1}^{t} w_i \bar{\mathbf{p}}^\dagger \bar{\mathbf{p}}_i, \tag{38}
$$

$$
= \sum_{i=1}^{t} \frac{w_i \bar{\mathbf{p}}_i}{\left\| \sum_{i=1}^{t} w_i \bar{\mathbf{p}}_i \right\|}, \tag{39}
$$

where we choose $w_i = \kappa_t a^{t-i}$ with $a \in (0,1]$ similar to FTL, and it leads to the following simplification as

$$
\hat{\bar{\mathbf{p}}}_{t+1} = \sum_{i=1}^{t} \bar{w}_{i,t} \bar{\mathbf{p}}_i \tag{40}
$$

$$
= \bar{w}_{t,t} \bar{\mathbf{p}}_t + \frac{\bar{w}_{t,t}}{\bar{w}_{t,t-1}} \sum_{i=1}^{t-1} \bar{w}_{i,t-1} \bar{\mathbf{p}}_i \tag{41}
$$

$$
= \bar{w}_{t,t} \bar{\mathbf{p}}_t + \frac{\bar{w}_{t,t}}{\bar{w}_{t,t-1}} \hat{\bar{\mathbf{p}}}_t, \tag{42}
$$

$$
\geq \bar{w}_{t,t} \bar{\mathbf{p}}_t + (1 - \bar{w}_{t,t}) \hat{\bar{\mathbf{p}}}_t, \tag{43}
$$

with $\bar{w}_{i,t} = \frac{w_i}{\left\| \sum_{i=1}^{t} w_i \bar{\mathbf{p}}_i \right\|}$, $\frac{\bar{w}_{t,t-1}}{\bar{w}_{t,t}} = \frac{\left\| \sum_{i=1}^{t} w_i \bar{\mathbf{p}}_i \right\|}{\left\| \sum_{i=1}^{t-1} w_i \bar{\mathbf{p}}_i \right\|} \leq 1 + \bar{w}_{t,t-1}$ using the triangle inequality, and yielding $\frac{\bar{w}_{t,t}}{\bar{w}_{t,t-1}} \geq 1 - \bar{w}_{t,t}$. To see that the above bound is a convex sum, the weight $\bar{w}_{t,t}$ should be less than 1. Thus, we write $\bar{w}_{t,t} = \left\| \sum_{i=1}^{t} a^{t-i} \bar{\mathbf{p}}_i \right\|^{-1}$, which is the inverse of a norm of the converging sum. In the worst case, the norm has the lowest value, when $\bar{\mathbf{p}}_i$ is uniform[2], i.e., for $\bar{p}_{f,t} = \frac{1}{\sqrt{N}} \forall f$, we have

$$
\begin{aligned}
\lim_{t \to \infty} \bar{w}_{t,t} &\leq \lim_{t \to \infty} \left(\sum_{i=1}^{t} a^{t-i} \left\| \frac{\mathbf{1}}{\sqrt{N}} \right\| \right)^{-1} \\
&= \lim_{t \to \infty} \frac{1-a}{1-a^t} = 1 - a < 1.
\end{aligned}
$$

Thus, $\bar{w}_{t,t} \leq 1$, where the equality is obtained for $t = 1$.

The corresponding online learning procedure is presented in Algorithm 3. In this procedure, two intermediate variables (k_t and $\bar{\mathbf{p}}'_{t+1}$) are introduced to simplify the computations. The rest of the process is same as weighted FTL algorithm. The respective regret is analyzed in the following result.

Algorithm 3 Weighted FoReL algorithm

Input: the value of $0 < a \leq 1$, $\kappa_0 = 0$.
1: **for** $t = 1, 2, \ldots$ **do**
2:      set $w_t = \frac{1-a}{1-a^t}$
3:      predict $\bar{\mathbf{p}}'_{t+1} = w_t \bar{\mathbf{p}}_t + k_{t-1} \hat{\bar{\mathbf{p}}}_t$
4:      set $k_t = \|\bar{\mathbf{p}}'_{t+1}\|$ and obtain $\hat{\bar{\mathbf{p}}}_{t+1} = \frac{\bar{\mathbf{p}}'_{t+1}}{k_t}$
5:      observe $\bar{\mathbf{p}}_{t+1}$
6: **end for**

Lemma 5. *For the weighted FoReL approach with $0 < a < 1$, $\mathcal{O}(T)$ regret bound is obtained, and $\mathcal{O}(\log T)$ for $a = 1$.*

Proof: The CP estimate can be simplified as

$$
\hat{\bar{\mathbf{p}}}_{t+1} - \bar{\mathbf{p}}_t \geq (1 - \bar{w}_{t,t}) \left[\hat{\bar{\mathbf{p}}}_t - \bar{\mathbf{p}}_t \right].
$$

It is used to get the difference with respect to the CP ($\mathbf{p}$) as

$$
\begin{aligned}
&\|\hat{\mathbf{p}}_t - \mathbf{p}_t\|^2 - \|\mathbf{p} - \mathbf{p}_t\|^2 \\
&\overset{(a)}{\approx} \|\hat{\bar{p}}_t - \bar{\mathbf{p}}_t\|^2 - \|\bar{\mathbf{p}} - \bar{\mathbf{p}}_t\|^2 \tag{44} \\
&\overset{(b)}{\leq} \|\hat{\bar{\mathbf{p}}}_t - \bar{\mathbf{p}}_t\|^2 - \|\hat{\bar{\mathbf{p}}}_{t+1} - \bar{\mathbf{p}}_t\|^2 \tag{45} \\
&\overset{(c)}{\leq} 2\bar{w}_{t,t} \|\hat{\bar{\mathbf{p}}}_t - \bar{\mathbf{p}}_t\|^2 \tag{46} \\
&\overset{(d)}{\leq} 4\bar{w}_{t,t},
\end{aligned}
$$

where in (a), we approximate the regret of CP by the regret of SCP; in (b), the result from [27, Lem. 2.1] is used; in (c),

---

[2] The norm minimization problem can be written as

$$
\min_{\bar{\mathbf{p}}_i} \left\| \sum_{i=1}^{t} a^{t-i} \bar{\mathbf{p}}_i \right\|^2 = \min_{\sum_f \bar{p}_{f,t}^2 = 1 \forall t} \sum_f \left| \sum_{i=1}^{t} a^i \bar{p}_{f,i} \right|^2,
$$

which has more unknowns than the known. Relaxing the norm constraint (since the final vector can be normalized), the problem reduces to $\min_{\bar{p}_{f,i} \forall t} \left| \sum_{i=1}^{t} a^i \bar{p}_{f,i} \right|^2 \forall f$, which shows that $\bar{p}_{f,i} \propto a^{-i} \forall f$ minimizes it. This result after normalizing gives uniform distribution $\bar{p}_{f,t} = \frac{1}{\sqrt{N}} \forall f$.

$-\bar{w}_{t,t}^2$ is ignored as $\bar{w}_{t,t} \leq 1$; in (d), triangle inequality has been used with $\|\hat{\bar{\mathbf{p}}}_t\| = \|\bar{\mathbf{p}}_t\| = 1$. The total regret up to time T can be obtained as

$$\sum_{t=1}^{T} \left[\|\hat{\bar{\mathbf{p}}}_t - \bar{\mathbf{p}}_t\|^2 - \|\bar{\mathbf{p}}^* - \bar{\mathbf{p}}_t\|^2 \right] \qquad (47)$$

$$\leq 4 \sum_{t=1}^{T} \bar{w}_{t,t} = 4 \sum_{t=1}^{T} \left\| \sum_{i=1}^{t} a^{t-i} \bar{\mathbf{p}}_i \right\|^{-1} \qquad (48)$$

$$\overset{(a)}{\leq} 4 \sum_{t=1}^{T} \left(\frac{1-a^t}{1-a} \right)^{-1} \qquad (49)$$

$$\overset{(b)}{\approx} 4(1-a) \left(T + a \frac{1-a^T}{1-a} \right), \qquad (50)$$

which (a) is obtained for $\bar{p}_{f,t} = \frac{1}{\sqrt{N}} \forall f$; in (b), $(1-a^t)^{-1} \approx 1 + a^t$. Thus, it yields $\mathcal{O}(T)$ regret. When $a = 1$, $w_t = t^{-1}$ and $\bar{w}_{t,t} = \left\| \sum_{i=1}^{t} \bar{\mathbf{p}}_i \right\|^{-1} \leq \frac{1}{t}$ where the equality of the upper bound on $\bar{w}_{t,t}$ holds for $\bar{p}_{f,t} = \frac{1}{\sqrt{N}} \forall f$. This gives $\mathcal{O}(\log T)$ regret. ∎

Remark (OP vs OL): The OP methods work via linear prediction for given τ observations, while OL methods improves the estimation by the experience. In OP methods, the predictor coefficients are assumed unconstrained, while the same assumptions cannot be set for OL methods due to the objective of weighted MSE minimization. With these observations, the OP yields a linear sum of the recent past CPs, while OL provides the convex sum of all the past CPs. These differentiations suggest that OP can estimate any wide changes in content popularity, while OL can track the changes in the CPs within a convex set of past observations.

VI. SIMULATION RESULTS

To evaluate the performance of the proposed prediction methods with the optimized edge caching policy, we use MovieLens dataset [20]. In this dataset, we choose the user ratings of $N = 100$ movies with IDs 1-100. Using the timestamps provided, the whole duration is divided into time slots to simulate the content request process. A movie rating from users is assumed to be the number of requests of that movie, while the popularity profile for each time slot is obtained by normalizing the ratings across the movies. Moreover, for OP methods, $d = 4$ and $\tau = 10$ are selected, while for OL, $a = 1$ is chosen. PPP parameters for computing ASP are as follows: noise power $\sigma^2 = 0$, BS density $\lambda_{BS} = 200$, bandwidth $W = 24$kHz, path loss exponent $\alpha = 3.5$, rate threshold $R_0 = 1$, BS cache size $L = \frac{N}{2} = 50$. Besides, the performance of OP and OL methods are compared with the request prediction method (OP-AR) [10], and mean guessing [15], [21]. In OP-AR, after modeling the logarithm of the number of requests using auto-regressive model, we use least squares to find the coefficients of prediction. In mean guessing, a mean of previous τ observations is selected as the prediction.

The above methods accurately predict the CPs in the order of 10^{-2}. For comparison, Figure 2 depicts the averaged prediction MSE and the averaged expected ASP difference (dASP) for the methods. It can be observed that for both
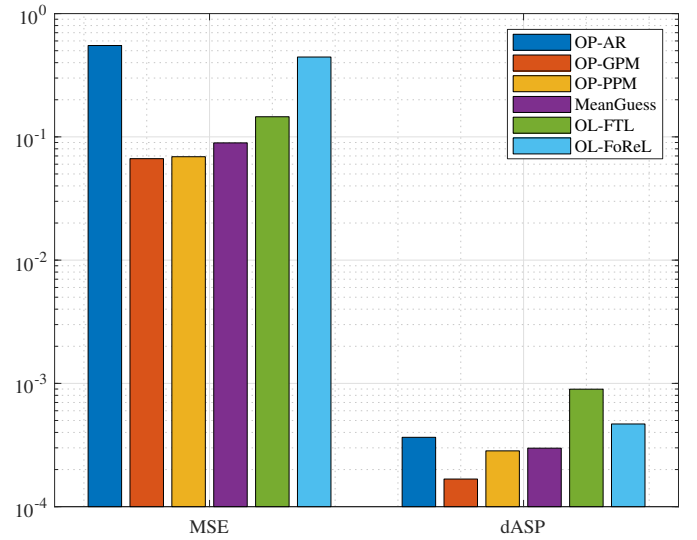

Figure 2. Figure illustrates the average prediction MSE ($\mathbb{E}_t \{\mathcal{E}_t(\hat{\mathbf{p}}_t)\}$) and the expected ASP difference ($\mathbb{E}_t \{\Delta_t(\hat{\mathbf{p}}_t)\}$).

MSE and ASP, OP-GPM and OP-PPM yield better results than OP-AR, mean guessing and OL methods. This is because in each online round of OP-PPM and OP-GPM, the MSE of CP estimate is minimized to get prediction, while in others, it is not. Among the OL methods, OL-FTL provides better MSE while OL-FoReL has better dASP, as OL-FTL considers CP in the formulation, while OL-FoReL is SCP based. Also, the plots of mean guessing lie in between OP and OL methods, which is because mean guess considered here has similarities to both OP and OL, i.e., in other words, mean guessing is equivalent to a trivial OP or a sub-optimal OL-FTL. Note that OP predicts with past τ observations, while OL utilizes the whole past. Therefore, OP performs well when the instantaneous MSE is considered, while OL provides improvements for the cumulative MSE, i.e., regret, which is shown in the Figure 3. OP-AR for MSE and dASP can be seen to provide an approximation to OL-GPM.

Figure 3 (a) and (b) show the MSE and dASP regret respectively. Unlike Figure 2, OL-FTL yields better MSE regret than OP methods and mean guessing, while OL-FoReL provides better dASP regret as per construction. The MSE regret can be seen to be $\mathcal{O}(\log T)$, while dASP regret is $\mathcal{O}(T)$ as presented in the previous sections. Mean guessing provides an approximation between OP and OL methods for both MSE and dASP, as it is a sub-optimal FTL as well as a trivial OP. OP-AR approximates OP-GPM as seen in Figure 2.

Figure 4 shows variations of the expected dASP with the cache size constraint. Here, the MSE remains constant as it is independent of caching scheme. It can be observed that with the increase in the cache size, dASP decreases, i.e., the achievable ASP increases. The trend of different methods is similar to Figure 2, i.e., OP-GPM yields the minimum dASP and so on. For larger cache size, the dASP gap between different methods can be seen to be closing i.e., dASP converges to zero as $L \rightarrow N$. The expected dASP in (25) is inversely proportional to L (in η) i.e., dASP $\propto L^{-1} \implies \log(\text{dASP}) \propto -\log L$, which is a negative proportionality as can be seen in Fig. 4.
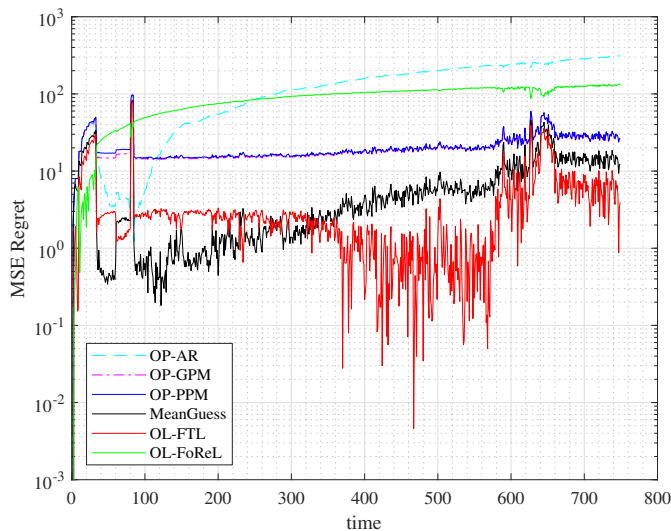
(a)
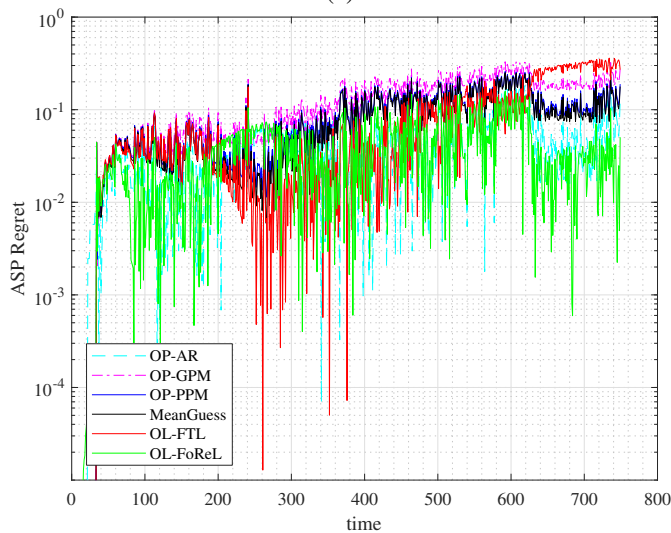
(b)

Figure 3. Figure depicts (a) MSE regret $(R_t(\mathbf{p}_t))$ versus t and (b) expected ASP regret $\left(R_t^{ASP}(\mathbf{p}_t)\right)$ versus t for different methods.

Figure 5 (a) and (b) depicts variations of the MSE and expected dASP as the content library size (N) increases, while the cache size is set proportional to the library size i.e. $L = N/2$. It can be observed that both the MSE and dASP decrease with the size N for all methods, except OP-AR, which increases because the prediction is done without normalization by N. The trend of the performance curves for different methods is similar as in Figure 2, where OP-GPM performs the best in both MSE and ASP measure.

VII. CONCLUSION

In this paper, online prediction (PPM and GPM) and on-line learning (weighted-FTL and weighted-FoReL) methods have been investigated. First, for the given popularity profile, caching probabilities have been optimized to maximize ASP of the PPP based network. In PPM, a linear model is used to predict the popularities to minimize the MSE and ASP difference has been analyzed. In GPM, we predict the future
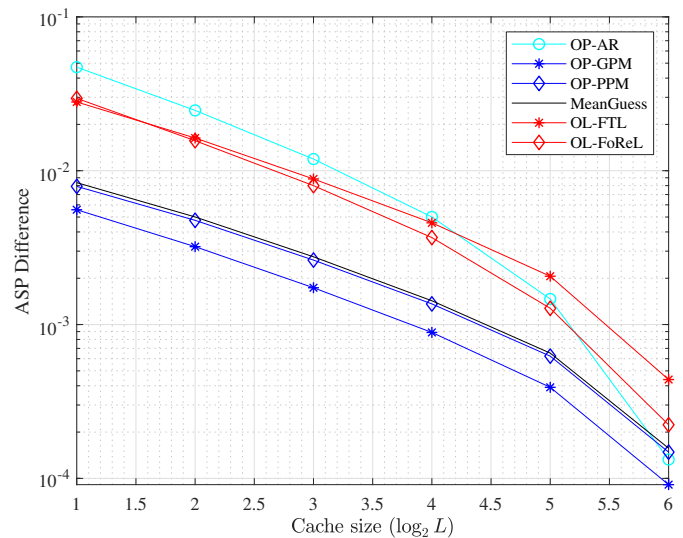

Figure 4. Figure shows the expected ASP difference $\mathbb{E}_t\{\Delta_t(\hat{\mathbf{p}}_t)\}$ with respect to the cache size constraint (L) keeping $N = 100$ fixed.

CP via the linear combination of SCP, which has shown to maximize the ASP. For online learning models, weighted-FTL and weighted-FoReL are presented to minimize the MSE and ASP regrets, which are analyzed to be $\mathcal{O}(\log T)$ and $\mathcal{O}(T)$ respectively. Simulations for MovieLens dataset verify the analysis that PPM and GPM achieves the better MSE and ASP, while FTL and FoReL results into better MSE regret and ASP regret respectively. ASP difference can be seen to be inversely proportional to cache constraint.

ACKNOWLEDGMENT

APPENDIX

A. ASP Derivation

The CCDF of sum rate can be obtained as

$$g(p_l) = \Pr\left(W \log_2(1 + \Gamma_{il}) \geq R_0\right) = \Pr\left(\Gamma_{il} \geq \underbrace{2^{\frac{R_0}{W}} - 1}_{s_0}\right)$$

$$= \mathbb{E}_{h_i, r_i, I_i} \Pr\left(\frac{\frac{|h_i|^2 P}{r_i^\alpha}}{I_i + \sigma^2} \geq s_0\right) \tag{51}$$

$$= \mathbb{E}_{r_i I_i}\left[\mathbb{E}_{h_i} \Pr\left(|h_i|^2 \geq s_0 r_i^\alpha \left(\frac{I_i}{P} + \frac{\sigma^2}{P}\right)\right)\right] \tag{52}$$

$$= \mathbb{E}_{r_i}\left\{\mathbb{E}_{I_i} \exp\left[-s_0 r_i^\alpha \left(\frac{I_i}{P} + \frac{\sigma^2}{P}\right)\right]\right\} \tag{53}$$

$$= \mathbb{E}_{r_i} \exp\left[-s_0 r_i^\alpha \left(\frac{\sigma^2}{P}\right)\right]$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCOMM.2019.2956041, IEEE Transactions on Communications
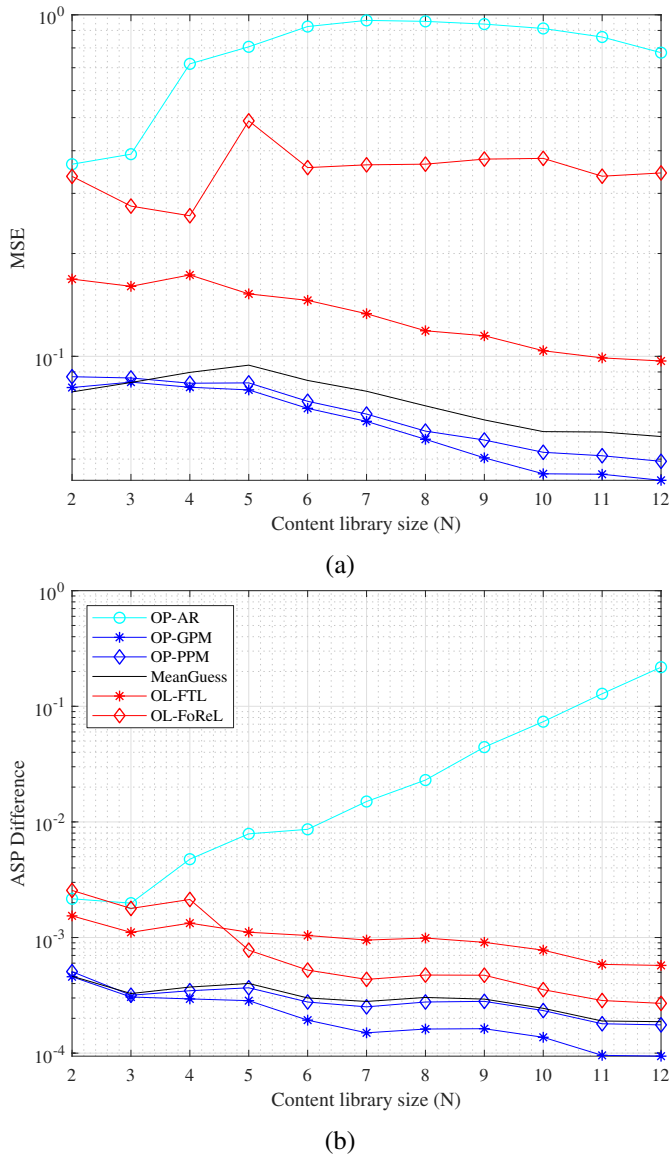
11

(a)

(b)

Figure 5. Figure shows (a) MSE ($\mathbb{E}_t \{\mathcal{E}_t (\hat{\mathbf{p}}_t)\}$) and (b) the expected ASP difference $\mathbb{E}_t \{\Delta_t(\hat{\mathbf{p}}_t)\}$ with respect to the content library size (N) with the cache constraint $L = N/2$.

$$\times \mathbb{E}_{h_j, r_j} \exp \left[-s_0 \sum_{j \in \Phi_{bs} \setminus \{i\}} |h_j|^2 \left(\frac{r_i}{r_j} \right)^\alpha \right] \quad (54)$$

$$= \mathbb{E}_{r_i} \exp \left[-s_0 r_i^\alpha \left(\frac{\sigma^2}{P} \right) \right] \quad (55)$$

$$\times \mathbb{E}_{r_j} \left[\prod_{j \in \Phi_{bs}(l) \setminus \{i\}} \prod_{j \in \Phi_{bs}^c(l) \setminus \{i\}} \frac{1}{1 + s_0 \left(\frac{r_i}{r_j} \right)^\alpha} \right] \quad (56)$$

Now, the expectation on distance will be taken [28]

$$\mathbb{E}_{r_j} \prod_{j \in \Phi_{bs}(l) \setminus \{i\}} \left(1 + s_0 r_i^\alpha r_j^{-\alpha} \right)^{-1}$$

$$= \exp \left(- \int_{r_i}^\infty \left(1 - \frac{1}{1 + s_0 \left(\frac{r_i}{r_j} \right)^\alpha} \right) 2\pi \lambda_{bs} p_l r_j \, (\mathrm{d}r_j) \right)$$

$$= \exp \left(-2\pi \lambda_{bs} p_l \int_{r_i}^\infty \left(\frac{s_0 r_i^\alpha r_j^{-\alpha}}{1 + s_0 r_i^\alpha r_j^{-\alpha}} \right) r_j \, (\mathrm{d}r_j) \right) \quad (57)$$

$$= \exp \left(-2\pi \lambda_{bs} p_l \int_{r_i}^\infty \left(\frac{1}{1 + \frac{r_j^\alpha}{s_0 r_i^\alpha}} \right) r_j \, (\mathrm{d}r_j) \right) \quad (58)$$

$$= \exp \left(-2\pi \lambda_{bs} p_l s_0^{\frac{2}{\alpha}} r_i^2 \frac{1}{\alpha} \underbrace{\int_{\frac{1}{s_0}}^\infty \left(\frac{u^{\frac{2}{\alpha}-1}}{1+u} \right) du}_{=Ap_l r_i^2} \right) \quad (59)$$

$$\mathbb{E}_{r_j} \prod_{j \in \Phi_{bs}^c(l) \setminus \{i\}} \left(1 + s_0 r_i^\alpha r_j^{-\alpha} \right)^{-1} \quad (60)$$

$$= \exp \left(- \int_0^\infty \left(1 - \frac{1}{1 + s_0 \left(\frac{r_i}{r_j} \right)^\alpha} \right) \pi \lambda_{bs} (1 - p_l) \, (\mathrm{d}r_j^2) \right)$$

$$= \exp \left(-2\pi \lambda_{bs} (1 - p_l) s_0^{\frac{2}{\alpha}} r_i^2 \frac{1}{\alpha} \underbrace{\int_0^\infty \left(\frac{u^{\frac{2}{\alpha}-1}}{1+u} \right) du}_{=B(1-p_l)r_i^2} \right) \quad (61)$$

where we let $u = \frac{r_j^\alpha}{s_0 r_i^\alpha}$, $r_j = (us_0)^{1/\alpha} r_i$, $du = \alpha \frac{r_j^{\alpha-1}}{s_0 r_i^\alpha} (\mathrm{d}r_j) = u\alpha \frac{1}{r_j} (\mathrm{d}r_j)$, $r_j (\mathrm{d}r_j) = du r_j^2 \frac{1}{u\alpha} = (us_0)^{2/\alpha} r_i^2 \frac{1}{u\alpha} = u^{\frac{2}{\alpha}-1} s_0^{\frac{2}{\alpha}} r_i^2 \frac{1}{\alpha}$. Therefore, the resultant expression from (55) can be written as

$$g(p_l) = \mathbb{E}_{r_i} \exp \left[-s_0 r_i^\alpha \left(\frac{\sigma^2}{P} \right) - (p_l A + (1 - p_l)B) r_i^2 \right]$$

$$= \int_0^\infty \exp \left[-s_0 r_i^\alpha \left(\frac{\sigma^2}{P} \right) - (p_l A + (1 - p_l)B) r_i^2 \right]$$
$$\times \exp(-\pi p_l \lambda_{bs} r_i^2) 2\pi p_l \lambda_{bs} r_i (\mathrm{d}r_i) \quad (62)$$

$$= p_l C \int_0^\infty \exp \left[-s_0 r_i^\alpha \left(\frac{\sigma^2}{P} \right) \right] \quad (63)$$
$$\times \exp \left[-(p_l A + (1 - p_l)B + p_l C) r_i^2 \right] (\mathrm{d}r_i^2) \quad (64)$$

where $C = \pi \lambda_{bs}$.

B. Proof of ASP maximization

Proof: The ASP optimization problem can be recast as

$$\min_{q_1, \dots q_N} - \sum_f g_0(q_f) p_f \quad (65)$$

$$\text{subject to } \sum_f q_f \leq L, q_f \leq 1, q_f \geq 0, \forall f. \quad (66)$$

The corresponding Karush–Kuhn–Tucker (KKT) conditions can be obtained as

$$0 = -g_o'(q_f) p_f + \lambda + \nu_f - w_f = 0, \forall f \quad (67)$$

$$0 = \lambda (\sum_f q_f - L), \quad (68)$$

$$0 = \nu_f(q_f - 1), \forall f \quad (69)$$

$$0 = w_f q_f, \forall f, \quad (70)$$

where $\lambda \geq 0$, $v_f \geq 0$, $w_f \leq 0, \forall f$ are the dual variables and $g'(q) = \frac{BC}{[B+q(A+C-B)]^2}$. Simplifying above yields $q_f = \left(\frac{B}{A+C-B}\right)\left[\sqrt{\frac{C}{B}\frac{p_f}{\lambda'_f}} - 1\right]$, where $\lambda'_f = \lambda + \nu_f - w_f$. Let $\mathcal{F} = \mathcal{Z} \cup \mathcal{P} \cup \mathcal{R}$ with $\mathcal{Z} = \{f | v_f = 0, w_f < 0, q_f = 0\}$ $\mathcal{P} = \{f | v_f = w_f = 0, 0 < q_f < 1\}$, and $\mathcal{R} = \{f | v_f > 0, w_f = 0, q_f = 1\}$ being the sets of indices of q_f with zero, positive and one values respectively, i.e., for each set, we have

$$\lambda - w_f = g'(0)p_f = \frac{C}{B}p_f, \forall f \in \mathcal{Z} \tag{71}$$

$$\lambda = g'(q_f)p_f = \frac{C}{B}\frac{\left(\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}\right)^2}{\eta^2}, \forall f \in \mathcal{P} \tag{72}$$

$$\lambda + v_f = g'(1)p_f = \frac{C}{B}\left(\frac{B}{A+C}\right)^2 p_f, \forall f \in \mathcal{R}. \tag{73}$$

Further, the objective function is simplified as

$$\sum_f g_0(q_f)p_f = \sum_f \lambda'_f q_f \frac{B + q_f(A+C-B)}{B} \tag{74}$$

$$= \sum_{f \in \mathcal{P}} \lambda q_f \sqrt{\frac{C}{B}\frac{p_f}{\lambda}} + \sum_{f \in \mathcal{R}} \lambda'_f \frac{A+C}{B} \tag{75}$$

$$= \left(\frac{B\lambda}{A+C-B}\right)\sum_{f \in \mathcal{P}}\left[\frac{C}{B}\frac{p_f}{\lambda} - \sqrt{\frac{C}{B}\frac{p_f}{\lambda}}\right] + \frac{C}{A+C}\sum_{f \in \mathcal{R}} p_f$$

$$= \left(\frac{C}{A+C-B}\right)\left[\sum_{f \in \mathcal{P}} p_f - \frac{\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{1}}{\eta}\sqrt{p_f}\right] + \frac{C}{A+C}\bar{\mathbf{p}}_{\mathcal{R}}^T\bar{\mathbf{p}}_{\mathcal{R}}$$

$$= \left(\frac{C}{A+C-B}\right)\bar{\mathbf{p}}_{\mathcal{P}}^T \mathbf{Z}_{\mathcal{P}} \bar{\mathbf{p}}_{\mathcal{P}} + \frac{C}{A+C}\bar{\mathbf{p}}_{\mathcal{R}}^T\bar{\mathbf{p}}_{\mathcal{R}} \tag{76}$$

$$= \mathbf{p}^T \mathbf{a} - \left(\frac{C}{A+C-B}\right)\frac{\left(\bar{\mathbf{p}}_{\mathcal{P}}^T\mathbf{1}\right)^2}{\eta}. \tag{77}$$

$\blacksquare$

C. Solution of constrained NNLS problem

The equation (24) can be written as constrained NNLS problem as

$$\hat{\mathbf{p}}_{t+1} = \arg\min_{\mathbf{p} \geq \mathbf{0}} \left\| \overrightarrow{\mathbf{p}}_t - \overrightarrow{\mathbf{P}}_t\mathbf{p} \right\|_2^2 \tag{78}$$

$$\text{subject to } \mathbf{1}^T\mathbf{p} = 1, \tag{79}$$

where $\overrightarrow{\mathbf{p}}_t = [\mathbf{p}_{t-\tau+d+1}^T, \dots, \mathbf{p}_t^T]^T$, $\mathbf{p} = \sum_{k=1}^d c_k\mathbf{p}_{t+1-k}$, $\mathbf{P}'_t = [\overrightarrow{\mathbf{p}}_{t-d}, \dots \overrightarrow{\mathbf{p}}_{t-1}]$, $\overrightarrow{\mathbf{P}}_t = \mathbf{P}'_t[\mathbf{p}_t, \dots, \mathbf{p}_{t+1-d}]^+$; $\mathbf{A}^+$ stands for the pseudo inverse of $\mathbf{A}$. Further, we modify FNNLS algorithm to handle constraints using the following Karush–Kuhn–Tucker (KKT) conditions as

$$\mathbf{0} = \overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\mathbf{p} - \overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{p}}_t + \lambda\mathbf{1} + \mathbf{v}, \tag{80}$$

$$0 = \lambda(\mathbf{1}^T\mathbf{p} - 1), \tag{81}$$

$$0 = v_f p_f, \forall f, \tag{82}$$

where $\lambda \neq 0$ and $\mathbf{v} = [v_1, \dots, v_N] \leq \mathbf{0}$ are the dual variables corresponding to sum and the non-negativity constraints

---

Algorithm 4 Constrained NNLS algorithm

Output: $\mathbf{x}^* = \arg\min_{\mathbf{x}\geq\mathbf{0}} \|\mathbf{y} - \mathbf{Hx}\|_2^2$ such that $\mathbf{1}^T\mathbf{x} = 1$
Input: Initialize $\mathcal{P} = \emptyset$, $\mathcal{R} = \{1, \dots, n\}$, $\mathbf{p} = \mathbf{0}$, $\mathbf{v} = \mathbf{H}^T(\mathbf{y} - \mathbf{Hx})$ and tolerance ϵ
1: **while** $\mathcal{R} \neq \emptyset$ and $\max_i v_i > \epsilon$ **do**
2:     set $j = \arg\max_i v_i$, add j in $\mathcal{P}$ and remove from $\mathcal{R}$
3:     set $\mathbf{s}_{\mathcal{R}} = \mathbf{0}$ and $\mathbf{s}_{\mathcal{P}} = \mathbf{x}_{\mathcal{P}}(\mathbf{v})$,
4:     **if** $\min \mathbf{s}_{\mathcal{P}} \leq 0$ **then**
5:         $\alpha = -\min\left[\frac{\mathbf{x}_{\mathcal{P}}}{\mathbf{x}_{\mathcal{P}} - \mathbf{s}_{\mathcal{P}}}\right]$
6:         **if** $\alpha = 0$ **then**
7:             $\mathcal{I} = \{i | x_i \neq 0\}$ and $\alpha = -\min\left[\frac{\mathbf{x}_{\mathcal{I}}}{\mathbf{x}_{\mathcal{I}} - \mathbf{s}_{\mathcal{I}}}\right]$
8:         **end if**
9:         $\mathbf{x} = \mathbf{x} + \alpha(\mathbf{s} - \mathbf{x})$
10:       update $\mathcal{P} = \{i | x_i > 0\}$ and $\mathcal{R} = \{i | x_i \leq 0\}$
11:       set $\mathbf{s}_{\mathcal{R}} = \mathbf{0}$ and $\mathbf{s}_{\mathcal{P}} = \mathbf{x}_{\mathcal{P}}(\mathbf{v})$
12:     **end if**
13:     $\mathbf{x} = \mathbf{s}$ and $\mathbf{v} = \mathbf{H}^T(\mathbf{y} - \mathbf{Hx}) - \lambda_{\mathcal{P}}(\mathbf{v})\mathbf{1}$
14: **end while**

---

respectively. Solving these equations gives

$$\mathbf{p}(\overrightarrow{\mathbf{P}}_t, \mathbf{v}) = \left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)^{-1}\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{p}}_t - \lambda(\overrightarrow{\mathbf{P}}_t, \mathbf{v})\mathbf{1} - \mathbf{v}\right), \tag{83}$$

$$\lambda(\overrightarrow{\mathbf{P}}_t, \mathbf{v}) = \frac{\mathbf{1}^T\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)^{-1}\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{p}}_t - \mathbf{v}\right) - 1}{\mathbf{1}^T\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)^{-1}\mathbf{1}}. \tag{84}$$

However, for faster updation in an online round especially for a large content library, the following modified equations are used (inspired from FNNLS scheme [23], [24])

$$\mathbf{p}_{\mathcal{P}}(\mathbf{v}) = \left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)_{\mathcal{P}}^{-1}\left[\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{p}}_t\right)_{\mathcal{P}} - \lambda_{\mathcal{P}}(\overrightarrow{\mathbf{P}}_t, \mathbf{v})\mathbf{1} - \mathbf{v}\right],$$

$$\lambda_{\mathcal{P}}(\mathbf{v}) = \frac{\mathbf{1}^T\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)_{\mathcal{P}}^{-1}\left[\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{p}}_t\right)_{\mathcal{P}} - \mathbf{v}\right] - 1}{\mathbf{1}^T\left(\overrightarrow{\mathbf{P}}_t^T\overrightarrow{\mathbf{P}}_t\right)_{\mathcal{P}}^{-1}\mathbf{1}}, \tag{85}$$

where $\mathbf{A}_{\mathcal{P}}$ denotes the sub-matrix of $\mathbf{A}$ with the rows-columns indices defined by $\mathcal{P}$. These equations depends on the dual variable $\mathbf{v}$, which is acquired by active set method as presented in the modified NNLS procedure in Algorithm 4. After initializing $\mathbf{v}$, this method works by computing the positive set of entries and updating the corresponding $\mathbf{v}$ iteratively. The number of "while" iterations is equal to the number of non-zero entries in the solution. The difference between Algorithm 4 from FNNLS algorithm [23], [24] is the presence of dual variable λ which handles the additional constraint other than non-negativity.

D. Simplification of ASP Difference

The simplification is given in the equation on the top of the page.

$$\delta_{t+1}(\hat{\mathbf{p}}_{t+1}) = P_s(\mathbf{p}_{t+1}) - P_s(\mathbf{p}_{t+1}, \hat{\mathbf{q}}_{t+1}) = \sum_f p_{f,t+1}\left[g_0(q_f(p_{f,t+1})) - g_0(q_f(\hat{p}_{f,t+1}))\right] \tag{86}$$

$$= \frac{C}{A+C-B}\sum_{f\in\mathcal{P}_{t+1}}\frac{p_{f,t+1}}{\eta}\left[\frac{\mathbf{1}^T\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}}{\hat{\bar{p}}_{f,t+1}} - \frac{\mathbf{1}^T\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}}{\bar{p}_{f,t+1}}\right] \tag{87}$$

$$= \left(\frac{C\eta^{-1}}{A+C-B}\right)\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}^T\left[\mathbf{1}^T\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\mathcal{D}^{-1}\left(\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\right) - \mathbf{1}\mathbf{1}^T\right]\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}} \tag{88}$$

$$\leq \left(\frac{C\eta^{-1}}{A+C-B}\right)\lambda_{\max}\left[\mathbf{1}^T\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\mathcal{D}^{-1}\left(\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\right) - \mathbf{1}\mathbf{1}^T\right]\|\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}\|^2 \tag{89}$$

$$= \left(\frac{C\eta^{-1}\|\bar{\mathbf{p}}_{t+1,\mathcal{P}_{t+1}}\|^2}{A+C-B}\right)\lambda_{\max}\left[\mathbf{1}^T\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\mathcal{D}^{-1}\left(\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}\right) - |\mathcal{P}_{t+1}|\right] \tag{90}$$

$$= \left(\frac{C\eta^{-1}\mathbf{1}^T\mathbf{p}_{t+1,\mathcal{P}_{t+1}}}{A+C-B}\right)\left[\frac{\mathbf{1}^T\hat{\bar{\mathbf{p}}}_{t+1,\mathcal{P}_{t+1}}}{\min_{f\in\mathcal{P}_{t+1}}\hat{\bar{p}}_{f,t+1}} - |\mathcal{P}_{t+1}|\right] \tag{91}$$

REFERENCES

[1] H. S. Goian, O. Y. Al-Jarrah, S. Muhaidat, Y. Al-Hammadi, P. Yoo, and M. Dianati, "Popularity-based video caching techniques for cache-enabled networks: A survey," *IEEE Access*, vol. 7, pp. 27 699–27 719, 2019.

[2] L. Li, G. Zhao, and R. S. Blum, "A survey of caching techniques in cellular networks: Research issues and challenges in content placement and delivery strategies," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1710–1732, 2018.

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

[4] K. Poularakis and L. Tassiulas, "On the complexity of optimal content placement in hierarchical caching networks," *IEEE Transactions on Communications*, vol. 64, no. 5, pp. 2092–2103, 2016.

[5] B. Blaszczyszyn and A. Giovanidis, "Optimal geographic caching in cellular networks," in *IEEE International Conference on Communications (ICC)*, 2015, pp. 3358–3363.

[6] B. Serbetci and J. Goseling, "Optimal geographical caching in heterogeneous cellular networks with nonhomogeneous helpers," *arXiv preprint arXiv:1710.09626*, 2017.

[7] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, "Optimal and scalable caching for 5G using reinforcement learning of space-time popularities," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 180–190, 2018.

[8] J. Yin, L. Li, H. Zhang, X. Li, A. Gao, and Z. Han, "A prediction-based coordination caching scheme for content centric networking," in *WOCC*, 2018, pp. 1–5.

[9] W.-X. Liu, J. Zhang, Z.-W. Liang, L.-X. Peng, and J. Cai, "Content popularity prediction and caching for ICN: A deep learning approach with SDN," *IEEE Access*, vol. 6, pp. 5075–5089, 2018.

[10] H. Nakayama, S. Ata, and I. Oka, "Caching algorithm for content-oriented networks using prediction of popularity of contents," in *IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015, pp. 1171–1176.

[11] Y. Zhang, X. Tan, and W. Li, "Ppc: Popularity prediction caching in icn," *IEEE Communications Letters*, vol. 22, no. 1, pp. 5–8, Jan 2018.

[12] B. Bharath, K. G. Nagananda, and H. V. Poor, "A learning-based approach to caching in heterogenous small cell networks," *IEEE Transactions on Communications*, vol. 64, no. 4, pp. 1674–1686, 2016.

[13] R. Haw, S. M. A. Kazmi, K. Thar, M. G. R. Alam, and C. S. Hong, "Cache aware user association for wireless heterogeneous networks," *IEEE Access*, vol. 7, pp. 3472–3485, 2019.

[14] J. Wu, Y. Zhou, D. M. Chiu, and Z. Zhu, "Modeling dynamics of online video popularity," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1882–1895, Sep. 2016.

[15] Y. Jiang, M. Ma, M. Bennis, F. Zheng, and X. You, "User preference learning-based edge caching for fog radio access network," *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1268–1283, Feb 2019.

[16] K. Avrachenkov, J. Goseling, and B. Serbetci, "A low-complexity approach to distributed cooperative caching with geographic constraints,"

Proceedings of the ACM on Measurement and Analysis of Computing Systems, vol. 1, no. 1, p. 27, 2017.

[17] D. Liu and C. Yang, "Caching policy toward maximal success probability and area spectral efficiency of cache-enabled HetNets," *IEEE Transactions on Communications*, vol. 65, no. 6, pp. 2699–2714, 2017.

[18] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, and G. Caire, "The role of caching in future communication systems and networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 6, pp. 1111–1125, 2018.

[19] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, "Dynamic mobile edge caching with location differentiation," in *IEEE Global Communications Conference*, Dec 2017, pp. 1–6.

[20] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1–19:19, Dec. 2015. [Online]. Available: http://doi.acm.org/10.1145/2827872

[21] S. Müller, O. Atan, M. van der Schaar, and A. Klein, "Context-aware proactive content caching with service differentiation in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 2, pp. 1024–1036, 2017.

[22] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[23] R. Bro and S. Jong, "A fast non-negativity- constrained least squares algorithm," *Journal of Chemometrics*, vol. 11, pp. 393–401, 1997.

[24] C. L. Lawson and R. J. Hanson, *Solving least squares problems*. Siam, 1995, vol. 15.

[25] O. El Ayach and R. W. Heath, "Grassmannian differential limited feedback for interference alignment," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6481–6494, 2012.

[26] Y. Zhang and M. Lei, "Robust Grassmannian prediction for limited feedback multiuser MIMO systems," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2012, pp. 863–867.

[27] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Foundations and Trends® in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2012.

[28] U. Schilcher, S. Toumpis, M. Haenggi, A. Crismani, G. Brandner, and C. Bettstetter, "Interference functionals in poisson networks," *IEEE Transactions on Information Theory*, vol. 62, no. 1, pp. 370–383, 2016.